

macro@ufmg

Universidade Federal de Minas Gerais

Programa de Pós-Graduação em Engenharia Elétrica

Research group MACRO - Mechatronics, Control and Robotics

**SET-BASED STATE ESTIMATION AND FAULT
DIAGNOSIS USING CONSTRAINED
ZONOTOPES AND APPLICATIONS**

Brenner Santana Rego

Belo Horizonte, Brazil

2022

Brenner Santana Rego

**SET-BASED STATE ESTIMATION AND FAULT
DIAGNOSIS USING CONSTRAINED
ZONOTOPES AND APPLICATIONS**

Thesis submitted to the Graduate Program in Electrical Engineering of Escola de Engenharia at the Universidade Federal de Minas Gerais, in partial fulfillment of the requirements for the degree of Doctor in Electrical Engineering.

Advisors: Guilherme Vianna Raffo
Davide Martino Raimondo

Belo Horizonte, Brazil
2022

R343s

Rego, Brenner Santana.

Set-based State Estimation and Fault Diagnosis Using Constrained Zonotopes and Applications [recurso eletrônico] / Brenner Santana Rego. - 2021.

1 recurso online (xxxiv, 240 f. : il., color.) : pdf.

Orientador: Guilherme Vianna Raffo.

Coorientador: Davide Martino Raimondo.

Tese (doutorado) - Universidade Federal de Minas Gerais, Escola de Engenharia.

Apêndices: f.209-222.

Bibliografia: f.223-240.

Exigências do sistema: Adobe Acrobat Reader.

1. Engenharia Elétrica - Teses. 2. Teoria da Estimativa – Teses. 3. Zonotopos – Teses. I. Raffo, Guilherme Vianna. II. Raimondo, Davide Martino. III. Universidade Federal de Minas Gerais. Escola de Engenharia. IV. Título.

CDU: 621.3(043)



UNIVERSIDADE FEDERAL DE MINAS GERAIS
ESCOLA DE ENGENHARIA
Programa de Pós-Graduação em Engenharia Elétrica

**ATA DA 367ª DEFESA DE TESE DE DOUTORADO
DO PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA**

ATA DE DEFESA DE TESE DE DOUTORADO do aluno **Brenner Santana Rego** - registro de matrícula de número 2016752747. Às 09:00 horas do dia 28 do mês de junho de 2021, reuniu-se na Escola de Engenharia da UFMG a Comissão Examinadora da TESE DE DOUTORADO para julgar, em exame final, o trabalho intitulado "**Set-based State Estimation And Fault Diagnosis Using Constrained Zonotopes And Applications**" da Área de Concentração em Sinais e Sistemas. O Prof. Guilherme Vianna Raffo, orientador do aluno, abriu a sessão apresentando os membros da Comissão e, dando continuidade aos trabalhos, informou aos presentes que, de acordo com o Regulamento do Programa no seu Art. 8.16, será considerado APROVADO na defesa da Tese de Doutorado o candidato que obtiver a aprovação unânime dos membros da Comissão Examinadora. Em seguida deu início à apresentação do trabalho pelo Candidato. Ao final da apresentação seguiu-se a arguição do candidato pelos examinadores. Logo após o término da arguição a Comissão Examinadora se reuniu, sem a presença do Candidato e do público, e elegeu o Prof. Guilherme Vianna Raffo para presidir a fase de avaliação do trabalho, constituída de deliberação individual de APROVAÇÃO ou de REPROVAÇÃO e expedição do resultado final. As deliberações individuais de cada membro da Comissão Examinadora foram as seguintes:

Membro da Comissão Examinadora	Instituição de Origem	Deliberação	Assinatura
Prof. Dr. Guilherme Vianna Raffo - Orientador	DELT (UFMG)	APROVADO	
Prof. Dr. Davide Martino Raimondo - Co-Orientador	DIII (University of Pavia)	APROVADO	
Prof. Dr. Joseph Kirk Scott	ChBE (Georgia Institute of Technology)	APROVADO	
Prof. Dr. Bruno Vilhena Adorno	DEEE (University of Manchester)	APROVADO	
Prof. Dr. Roberto Kawakami Harrop Galvão	DELE (ITA)	APROVADO	
Prof. Dr. Bruno Otávio Soares Teixeira	DELT (UFMG)	APROVADO	 BRUNO OTAVIO SOARES TEIXEIRA:01358300607 Assinado de forma digital por BRUNO OTAVIO SOARES TEIXEIRA:01358300607 Dados: 2021.07.01 09:56:02 -03'00'

Tendo como base as deliberações dos membros da Comissão Examinadora a Tese de Doutorado foi APROVADA. O resultado final de APROVAÇÃO foi comunicado publicamente ao Candidato pelo Presidente da Comissão, ressaltando que a obtenção do Grau de Doutor em ENGENHARIA ELÉTRICA fica condicionada à entrega do TEXTO FINAL da Tese de Doutorado. O Candidato terá um prazo máximo de 30 (trinta) dias, a partir desta data, para fazer as CORREÇÕES DE FORMA e entregar o texto final da Tese de Doutorado na secretaria do PPGE/UFMG. As correções de forma exigidas pelos membros da Comissão Examinadora deverão ser registradas em um exemplar do texto da Tese de Doutorado, cuja verificação ficará sob a responsabilidade do Presidente da Banca Examinadora. Nada mais havendo a tratar o Presidente encerrou a reunião e lavrou a presente ATA, que será assinada pelo Presidente da Comissão Examinadora. Belo Horizonte, 28 de junho de 2021.

ASSINATURA DO PRESIDENTE DA COMISSÃO EXAMINADORA

*To Maria José Rego
and Osvaldo Santana Ferreira*

Acknowledgements

To my parents Maria José Rego and Osvaldo Santana Ferreira, and also to my closest brother Felipe Santana Rego, for supporting me and cheering me up throughout this path.

To my advisor Prof. Guilherme Vianna Raffo, for accepting me as his student, guiding me and also supporting me. Also to my advisor Prof. Davide Martino Raimondo for the essential support and collaboration up to this moment. In addition, to Prof. Joseph K. Scott, who played a very important role in most of the contributions presented in this thesis.

To my colleagues and ex-colleagues from MACRO, ProVANT, MACSIN, and other laboratories at the Federal University of Minas Gerais, for their friendship and also for supporting me when going through hard times. A special thanks to my friends from the Identification and Control of Dynamic Systems Laboratory at the University of Pavia, for their cordiality and support during my stay in Italy for the doctoral sandwich program. I prefer to not list any names here, so that I can thank everyone in an equal manner.

To the funding agencies CAPES, CNPq, FAPEMIG, and the InSAC, for the financial support, which was essential for the elaboration of this thesis.

To the Graduate Program in Electrical Engineering for providing the infrastructure required for the elaboration of this thesis.

Resumo

Esta tese de doutorado desenvolve novos métodos para estimação de estados baseada em conjuntos e diagnóstico ativo de falhas de (i) sistemas não lineares em tempo discreto com incertezas limitadas, (ii) sistemas não lineares em tempo discreto cujas trajetórias satisfazem restrições de igualdade não linear (chamadas de invariantes), (iii) estimação de estados baseada em conjuntos e diagnóstico de falha ativa de sistemas descritores lineares, e (iv) estimação conjunta de estados e parâmetros de sistemas descritores não lineares. Estimação baseada em conjuntos tem como objetivo obter envoltórios precisos para os possíveis estados do sistema em cada instante de tempo, sujeito a incertezas desconhecidas-porém-limitadas. A maioria dos métodos existentes emprega uma estrutura predição-atualização padrão, com etapas de predição e atualização baseadas em diferentes técnicas e representações de conjunto. No entanto, obter envoltórios precisos para sistemas não lineares continua sendo um desafio significativo. Quando esses envoltórios são representados por conjuntos simples, como intervalos, elipsóides, paralelotos e zonotos, certas operações de conjunto podem ser muito conservadoras. Ainda, a utilização de politopos convexos generalizados é muito mais exigente do ponto de vista computacional. Para resolver este problema, esta tese de doutorado propõe novos métodos para propagar, com eficiência, zonotos restritos (CZs) por meio de mapeamentos não lineares. Estes estendem métodos existentes baseados em zonotos de maneira consistente. Além disso, esta tese de doutorado melhora a estrutura de predição-atualização padrão para sistemas com invariantes, adicionando uma etapa de consistência. Esta nova etapa utiliza invariantes para reduzir o conservadorismo do envoltório e é habilitada por novos algoritmos para refinar zonotos restritos com base em restrições não lineares. Novos algoritmos de atualização são também desenvolvidos permitindo o uso de equações de medição não linear pela primeira vez em estimação de estados baseada em CZ. Além disso, esta tese de doutorado introduz uma nova abordagem para o diagnóstico ativo de falhas baseado em conjuntos para uma classe de sistemas não lineares em tempo discreto. Os novos envoltórios baseados em zonotos restritos são utilizados para detecção passiva de falhas de uma classe de sistemas não lineares em tempo discreto. Ainda, uma parametrização afim dos conjuntos alcançáveis também é obtida, na direção do projeto de uma entrada ótima para a realização de diagnóstico ativo de falhas baseado em conjuntos para sistemas não lineares. Além disso, esta tese apresenta

novos métodos baseados em zonotopos restritos para estimação de estados e diagnóstico ativo de falhas de sistemas descritores lineares. Em contraste com outras representações de conjuntos, restrições estáticas lineares nas variáveis de estado, típicas de sistemas descritores, podem ser incorporadas diretamente na descrição matemática de zonotopos restritos. Graças a esse recurso, os métodos baseados em CZs são capazes de fornecer envoltórios menos conservadores para as trajetórias de sistemas descritores lineares. Além disso, esta tese propõe uma nova representação para conjuntos ilimitados baseada em zonotopos, que permite desenvolver métodos de estimação de estados e diagnóstico ativo de falhas de sistemas descritores lineares sem assumir a existência de um conjunto admissível que envolva todas as trajetórias possíveis do sistema, e permitindo lidar com sistemas instáveis. Esta tese também desenvolve um novo método para a estimação conjunta de estados e parâmetros para sistemas descritores não lineares, que estende os métodos de estimação de estados não linear usando zonotopos restritos para incluir a estimação de parâmetros em uma estrutura unificada e, portanto, mantendo as dependências existentes entre os estados, variáveis algébricas e parâmetros desconhecidos, o que resulta em tanto a estimação de estados, quanto de parâmetros, sendo significativamente melhoradas. Por último, este texto apresenta a aplicação dos métodos propostos de estimação de estados e diagnóstico de falhas baseados em zonotopos restritos para veículos aéreos não tripulados, redes de distribuição de água e uma célula de íons de lítio.

Palavras-chave: Estimação de estados baseada em conjuntos, Diagnóstico de falhas, Zonotopos restritos.

Abstract

This doctoral thesis develops new methods for set-based state estimation and active fault diagnosis of (i) nonlinear discrete-time systems with bounded uncertainties, (ii) discrete-time nonlinear systems whose trajectories are known to satisfy nonlinear equality constraints (called invariants), (iii) set-based state estimation and active fault diagnosis of linear descriptor systems, and (iv) joint state and parameter estimation of nonlinear descriptor systems. Set-based estimation aims to compute tight enclosures of the possible system states in each time step subject to unknown-but-bounded uncertainties. Most existing methods employ a standard prediction-update framework with set-based prediction and update steps based on various set representations and techniques. However, achieving accurate enclosures for nonlinear systems remains a significant challenge. When these enclosures are represented by simple sets such as intervals, ellipsoids, parallelotopes, and zonotopes, certain set operations can be very conservative. Yet, using general convex polytopes is much more computationally demanding. To address this issue, the present doctoral thesis proposes new methods for efficiently propagating constrained zonotopes (CZs) through nonlinear mappings. These extend existing methods for zonotopes in a consistent way. Besides, this thesis improves the standard prediction-update framework for systems with invariants by adding a consistency step. This new step uses invariants to reduce conservatism and is enabled by new algorithms for refining constrained zonotopes based on nonlinear constraints. New update algorithms are also developed allowing nonlinear measurement equations in CZ-based state estimation for the first time. In addition, this doctoral thesis introduces a new approach for set-based active fault diagnosis of a class of nonlinear discrete-time systems. The new enclosures based on constrained zonotopes are employed for passive fault detection of a class of nonlinear discrete-time systems. Also, an affine parametrization of the reachable sets is obtained, in the direction of the design of an optimal input for set-based active fault diagnosis of nonlinear systems. In addition, this thesis presents new methods based on constrained zonotopes for set-valued state estimation and active fault diagnosis of linear descriptor systems. In contrast to other sets representations, linear static constraints on the state variables, typical of descriptor systems, can be directly incorporated in the mathematical description of constrained zonotopes. Thanks to this feature, set-based methods based on CZs can provide less conservative

enclosures for the trajectories of linear descriptor systems. Moreover, this thesis proposes a new representation for unbounded sets based on zonotopes, which allows to develop methods for state estimation and active fault diagnosis of linear descriptor systems without assuming the knowledge of an admissible set that encloses all the possible trajectories of the system, and allowing to deal with unstable systems. This thesis also develops a new method for set-based joint state and parameter estimation of nonlinear discrete-time systems, which extends the nonlinear state estimation methods using constrained zonotopes to include parameter estimation in a unified framework, and therefore maintaining existing dependencies between states, algebraic variables, and unknown parameters, which results in the accuracy of both state and parameter estimation being significantly improved. Lastly, this manuscript presents the application of the proposed set-based state estimation and fault diagnosis methods using constrained zonotopes to unmanned aerial vehicles, water distribution networks, and a lithium-ion cell.

Keywords: Set-based state estimation, Fault diagnosis, Constrained zonotopes.

List of Figures

1.1	Flow chart of this doctoral thesis.	34
4.1	Solid lines (blue) depicts a constrained zonotope $Z = \{\mathbf{G}, \mathbf{c}, \mathbf{A}, \mathbf{b}\}$ defined by $\mathbf{G} = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 2 & -1 \end{bmatrix}$, $\mathbf{c} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$, $\mathbf{A} = [-2 \ 1 \ -1]$, and $\mathbf{b} = 2$. Dashed lines denote the interval hull of Z (left), and the tight interval bundle containing Z considering a 10-partition (right).	62
4.2	Solid lines depicts an interval bundle \mathcal{B} . Left: the interval hull (dashed) and the endpoint intervals (red) of \mathcal{B} . Right: the auxiliary convex polytope, formed by the midpoints of the endpoint intervals (red, dashed).	63
4.3	Time evolution of the estimated bounds and the real states.	80
4.4	Top: The constrained zonotope X_0 with ‘ \times ’ denoting its center. Left: Enclosures obtained by applying Theorem 4.2 to (4.19). Right: Enclosures obtained by applying Theorem 4.3 to (4.19). The real vector γ_x is determined by $C1$ (green), $C2$ (red), $C3$ (yellow), and $C4$ (blue). For the mean value extension (left), $C4$ is overlapped with $C1$. Black dots denote uniform samples from X_0 propagated through (4.19).	81
4.5	The initial set X_0 (gray), the initial bounded uncertain measurements (dashed lines), the intersection computed as in Bravo et al. (2006) (yellow), and the constrained zonotope (green) computed by (4.18).	82
4.6	Results from the first four time steps of set-valued state estimation (after update) using the constrained zonotopic method CZMV (green) and the zonotopic method ZMV (yellow). Black dots denote uniform samples from X_0 propagated through (4.19) that are consistent with the current measurement.	83
4.7	Radii of the update sets obtained by applying ZMV (\circ), ZFO (\times), CZMV (\triangle), and CZFO ($+$) to (4.19) with process disturbances.	84
5.1	The sets X (blue), $\square X$ (dashed lines), the center of X (\times), the enclosures obtained using Proposition 5.2 with γ_x as the center of X (green), and using Theorem 4.3 with γ_x as the closest point in X to its center (red).	92

5.2	The enclosures \tilde{X}_k from the first four time steps of set-valued state estimation in the example in Section 5.4.1 using ZMV (yellow), ZFO (blue), CZMV (green), and CZFO (orange).	101
5.3	The radii of the estimated enclosures \tilde{X}_k in the example in Section 5.4.1 obtained using ZMV, ZFO, CZMV, and CZFO.	101
5.4	The zonotope X_0 (yellow), the enclosure \tilde{X}_0 obtained in the consistency step (5.5) using Corollary 5.1 with $\hat{X}_0 = X_0$ (red), the set $X_0 \cap \{\mathbf{I}_2, \mathbf{0}\}$ (blue), and the enclosure \tilde{X}_0 obtained in the consistency step (5.5) using Corollary 5.1 with $\hat{X}_0 = X_0 \cap \{\mathbf{I}_2, \mathbf{0}\}$ (green). The resulting enclosures contain each other according to the sequence above. The dashed line denotes the box $\{\mathbf{I}_2, \mathbf{0}\}$. The circle that describes the feasible state set of $\ \mathbf{x}_0\ _2^2 = 1$ is also depicted.	104
5.5	The radii of the estimated enclosures \tilde{X}_k for (5.16) obtained using ZMV-like and CZMV-like methods.	106
5.6	The radii of the estimated enclosures \tilde{X}_k for (5.16) obtained using ZFO-like and CZFO-like methods.	106
6.1	The planar robotic manipulator (D’Amato et al., 2011).	118
6.2	The reachable output sets for models 1 (colorless sets) and 2 (blue sets) along with the output $\vec{\mathbf{y}}$ observed by the injection of $\vec{\mathbf{u}}$ into models 1 (o) and 2 (x) for $k = 1, 2, \dots, 5$. For clarity, numbers the note the corresponding time step k for the output sequence $\vec{\mathbf{y}}^{[1]}$ observed from model 1 and its output reachable sets.	120
7.1	The feasible sets for the system (7.3): the initial set X_0 (red), the set S_0 (yellow), and S_k (cyan) for $k = 1, 2$	124
7.2	Examples of mixed zonotopes: the Euclidean space $\mathbb{R}^n = \{\mathbf{M}_R, _, \mathbf{c}_R, _, _, _ \}$ (left), where \mathbf{M}_R and \mathbf{c}_R can be chosen as \mathbf{I}_n and $\mathbf{0}_{n \times 1}$, respectively, and the strip $S = \{\mathbf{x} \in \mathbb{R}^2 : [-1 \ 1]\mathbf{x} - 1 \leq 0.5\} = \{\mathbf{I}_2, \mathbf{0}_{2 \times 1}, \mathbf{0}_{2 \times 1}, [-1 \ 1], -0.5, 1\}$ (right).	133
7.3	The zonotope Z (gray), the strip S (blue), the zonotope obtained using the intersection method (Bravo et al., 2006) (yellow), and the mixed zonotope $Z \cap S$ computed in MG-rep (red)	136
7.4	The strips S_1 (blue) and S_2 (red), and the intersection $S_1 \cap S_2$ computed computed in MG-rep (magenta).	137
7.5	The radii of the enclosures \hat{X}_k obtained using zonotope method proposed in Wang et al. (2018a), the constrained zonotope method proposed in Section 7.2, and the mixed zonotope method proposed in Section 7.5 (top), as well as the projections of \hat{X}_k onto x_3 (bottom).	148
7.6	The sets $\mathbf{C}^{[i]}X_0 \oplus \mathbf{D}^{[i]}\mathbf{u}_0 \oplus \mathbf{D}_v^{[i]}V$, the output reachable sets $Y_0^{[i]}$, $Y_4^{[i]}$, and 2000 samples for $\mathbf{y}_4^{[i]}$, with $i = 1$ (yellow), $i = 2$ (red), $i = 3$ (blue), and $i = 4$ (magenta), obtained by the injection of the input sequence $\vec{\mathbf{u}}$	149

7.7	The sets $\mathbf{C}^{[i]}X_0 \oplus \mathbf{D}^{[i]}\mathbf{u}_0 \oplus \mathbf{D}_v^{[i]}V$, the output reachable sets $Y_0^{[i]}$, $Y_5^{[i]}$, and 1000 samples for $\mathbf{y}_5^{[i]}$, with $i = 1$ (yellow), $i = 2$ (red), $i = 3$ (blue), and $i = 4$ (magenta), obtained by the injection of the input sequence $\vec{\mathbf{u}}$	150
8.1	The initial state \mathbf{x}_0 , the set \bar{X}_0 , and the projections onto \mathbf{x} of the enclosures obtained by CZMV, CZMV-N, and CZMV-J, in the update step using \mathbf{y}_0 for the first example (which are exactly the same in this case).	159
8.2	The variable (\mathbf{x}_0, p) , the initial set $\bar{X}_0 \times P$, and the enclosures obtained by CZMV, CZMV-N, and CZMV-J, in the first update step for the first example.	159
8.3	The states \mathbf{x}_k , and the projections onto the state space of the enclosures provided by CZMV, CZMV-N, and CZMV-J, for the first example and $k \in [0, 200]$	160
8.4	The radii of the projections onto the state space of the enclosures provided by CZMV, CZMV-N, and CZMV-J, for the first example and $k \in [0, 200]$	160
8.5	The volumes of the projections onto the state space of the enclosures provided by CZMV, CZMV-N, and CZMV-J, for the first example and $k \in [0, 200]$	161
8.6	The enclosures of the unknown parameter p provided by CZMV, CZMV-N, and CZMV-J, for the first example and $k \in [0, 200]$	162
8.7	The states \mathbf{x}_k , and the projections onto the state space of the enclosures provided by CZMV, CZMV-N, and CZMV-J, for the second example and $k \in [0, 500]$	163
8.8	The volumes of the projections onto the state space of the enclosures provided by CZMV, CZMV-N, and CZMV-J, for the second example and $k \in [0, 500]$	163
8.9	The enclosures of the algebraic variable λ_k provided by CZMV, CZMV-N, and CZMV-J, for the second example and $k \in [0, 500]$	164
8.10	The enclosures of the unknown parameter p provided by CZMV, CZMV-N, and CZMV-J, for the second example and $k \in [0, 200]$	164
9.1	The tilt-rotor UAV with suspended load, kinematic definitions, input forces and torques.	169
9.2	Trajectories performed by the UAV and the load.	178
9.3	Tracking error of the regulated variables.	179
9.4	Estimation error of the regulated variables obtained using the CZs (blue solid line) and the zonotope-based algorithm from Rego & Raffo (2019) (black dotted line), along with confidence limits (red solid lines for CZs, and green dotted lines for zonotopes).	179
9.5	Control signals applied to the UAV.	179

9.6	The trajectory performed by the quadrotor UAV (solid line), and the interval hulls of the constrained zonotopes (red boxes) and zonotopes (green boxes) estimated by CZMV and ZMV, respectively, projected onto (x, y, z)	182
9.7	Radii of the update sets computed by CZMV (solid black line) and ZMV (dashed blue line) for the quadrotor UAV experiment.	183
9.8	Radii of the update sets computed by CZFO (solid black line) and ZFO (dashed blue line) for the quadrotor UAV experiment.	183
10.1	The two-pipe network example.	199
10.2	The volumes of the sets \hat{X}_k obtained using IHISE, IHISE-CZ, ISAR, ISAR-CZ, and also by the interval hull and convex hull of the trajectories resulting from 1000 Monte-Carlo simulations (MCS) using EPANET for the two-pipe network.	200
10.3	The projections of \hat{X}_0 onto (q_1, q_2, h_1) , obtained using ISAR-CZ, the intervals computed component-wise by maximum and minimum values obtained from 1000 MCS, as well as the convex hull and samples of the corresponding trajectories.	201
10.4	Leak detection rate of the IHISE, ISAR, IHISE-CZ, and ISAR-CZ for the two-pipe network (leak at node 1), considering different leak magnitudes, averaged over 100 random simulations and the time interval $k \in [0, 24]$. . .	202
10.5	Leak detection rate of the IHISE, ISAR, IHISE-CZ, and ISAR-CZ for the two-pipe network (leak at node 1) over time, averaged over 100 random simulations and the leak magnitudes in the interval $[1, 15]\%$	202
10.6	The projection onto $\mathbf{q}_k \in \mathbb{R}^2$ of the sets \hat{X}_k obtained using IHISE and ISAR-CZ, together with the measured flow interval for \mathbf{q}_k at time $k = 2$ in one of the 100 random simulations considering leakage at node 1 of 4% of the average nominal inflow in the two-pipe example.	203
10.7	The benchmark network “Net1” used here as an example of a transport network.	203
10.8	The volumes of the sets \hat{X}_k obtained using IHISE, IHISE-CZ, ISAR, and ISAR-CZ for the transport network ‘Net1’.	204
10.9	Leak detection rate of the IHISE, ISAR, IHISE-CZ, and ISAR-CZ for the transport network ‘Net1’ (leak at node 8), considering different leak magnitudes, averaged over 100 random simulations and the time interval $k \in [0, 24]$	205
10.10	Leak detection rate of the IHISE, ISAR, IHISE-CZ, and ISAR-CZ for the transport network ‘Net1’ (leak at node 8) over time, averaged over 100 random simulations and the leak magnitudes in the interval $[2, 50]\%$	206

10.11	Leak detection rate of the IHISE, ISAR, IHISE-CZ, and ISAR-CZ for the transport network ‘Net1’ (leak at node 4), considering different leak magnitudes, averaged over 100 random simulations and the time interval $k \in [0, 24]$	206
10.12	Leak detection rate of the IHISE, ISAR, IHISE-CZ, and ISAR-CZ for the transport network ‘Net1’ (leak at node 4) over time, averaged over 100 random simulations and the leak magnitudes in the interval $[2, 50]\%$	207
10.13	The ‘CY-DMA’ network, a realistic example of a district metered area network.	207
10.14	The volumes of the sets \hat{X}_k obtained using IHISE, IHISE-CZ, ISAR, and ISAR-CZ for the CY-DMA network.	208
10.15	Leak detection rate of the IHISE, ISAR, IHISE-CZ, and ISAR-CZ for the CY-DMA network (leak at node 19), considering 7 pressure sensors, different leak magnitudes, averaged over 100 random simulations and the time interval $k \in [0, 24]$	209
10.16	Leak detection rate of the IHISE, ISAR, IHISE-CZ, and ISAR-CZ for the CY-DMA network (leak at node 19) over time, considering 7 pressure sensors, averaged over 100 random simulations and the leak magnitudes in the interval $[1, 25]\%$	209
11.1	Thévenin equivalent circuit model of the Lithium-ion cell.	216
11.2	Polynomial approximation of OCP-SOC dependency.	217
11.3	Training and validation current input profiles.	218
11.4	Validation profile of the terminal voltage.	220
11.5	Parametric correlation matrix.	221
11.6	Current input profile $I(t)$ applied to the ECM.	225
11.7	Enclosures obtained using interval state estimation.	225
11.8	Area of the estimated enclosures over time.	226

List of Tables

4.1	Computational complexity $O(\cdot)$ of basic operations.	76
4.2	Computational complexity $O(\cdot)$ of the state estimators.	78
4.3	Average radius ratio of the estimators with varying numbers of constraints. Each average is taken over 3 separate simulations using $n_g \in \{8, 12, 20\}$. . .	84
4.4	Average total times per iteration of the estimators with varying numbers of constraints. Each average is taken over 30 separate simulations using $n_g \in \{8, 12, 20\}$. Times spent only on complexity reduction are shown in parenthesis.	84
5.1	Computational complexity $O(\cdot)$ of the prediction, update, and consistency steps using constrained zonotopes.	99
5.2	Total and complexity reduction average execution times per time step of the state estimators for the first example.	102
5.3	Average radius ratio of the enclosures obtained by the state estimators (column per row) for the system (5.16).	107
5.4	Total and complexity reduction average execution times per time step of the state estimators for the system (5.16).	107
9.1	Model parameters of the system.	177
9.2	Parameters of the sensors.	177
9.3	Model parameters of the quadrotor UAV considered in the state estimation experiment.	180
9.4	Measured variables with error bounds.	181
9.5	Average radius ratio of the estimators with varying numbers of constraints. Each average is taken over 3 separate simulations with $n_g \in \{25, 30, 40\}$. . .	182
9.6	Average total times per iteration of the estimators with varying numbers of constraints. Each average is taken over 15 separate simulations using $n_g \in \{25, 30, 40\}$. Times spent only on complexity reduction are shown in parenthesis.	183

9.7	Model parameters of the quadrotor UAV considered in the fault diagnosis experiment.	185
10.1	Average execution times and number of iterations per time step in the two-pipe example.	200
10.2	Average execution times and number of iterations per time step for the ‘Net1’ transport network.	204
10.3	Average execution times and number of iterations per time step for the CY-DMA network.	207
11.1	SPMe cell section varying parameters.	216
11.2	Estimated ECM parameters.	219

Acronyms

AFD	Active Fault Diagnosis
ARR	Average Radius Ratio
BMS	Battery Management System
CAD	Computer Aided Design
CG-rep	Constrained Generator representation
CZ	Constrained Zonotope
CZFO	First-Order extension using Constrained Zonotopes
CZFO+C	First-Order extension using Constrained Zonotopes with Consistency step
CZFO+F	First-Order extension using Constrained Zonotopes considering Feasible set
CZFO+FC	First-Order extension using Constrained Zonotopes considering Feasible set and with Consistency step
CZFO-J	First-Order extension using Constrained Zonotopes, Joint approach
CZFO-N	First-Order extension using Constrained Zonotopes, Naive approach
CZIB	Constrained Zonotope and Interval Bundle
CZMV	Mean Value extension using Constrained Zonotopes
CZMV+C	Mean Value extension using Constrained Zonotopes with Consistency step
CZMV+F	Mean Value extension using Constrained Zonotopes considering Feasible set
CZMV+FC	Mean Value extension using Constrained Zonotopes considering Feasible set and with Consistency step
CZMV-J	Mean Value extension using Constrained Zonotopes, Joint approach

CZMV-N	Mean Value extension using Constrained Zonotopes, Naive approach
DC	Difference of Convex functions
DFN	Doyle-Fuller-Newman
DMA	District Metered Area
ECM	Electrical Circuit Model
EM	Electrochemical Model
FBCP	Forward-Backward Constraint Propagation
FD	Fault Diagnosis
FIM	Fisher Information Matrix
FTC	Fault-Tolerant Control
FVM	Finite Volume Method
G-rep	Generator representation
GPS	Global Positioning System
GPU	Graphics Processing Unit
H-rep	Half-space representation
IHISE	Iterative Hydraulic Interval State Estimation
IHISE-CZ	Iterative Hydraulic Interval State Estimation with Constrained Zonotope step
IMU	Inertial Measurement Unit
ISAR	Iterative Slope Approximation and Rescaling
ISAR-CZ	Iterative Slope Approximation and Rescaling with Constrained Zonotope step
LDS	Linear Descriptor Systems
LMI	Linear Matrix Inequality
LP	Linear Program
LPV	Linear Parameter Varying
MCS	Monte-Carlo Simulation

MIQP	Mixed-Integer Quadratic Programming
MG-rep	Mixed Generator representation
MPC	Model Predictive Control
MPT	Multi-Parametric Toolbox
MZ	Mixed Zonotope
OCP	Open Circuit Potential
ODE	Ordinary Differential Equation
PDE	Partial Differential Equation
SOC	State Of Charge
SPMe	Single Particle Model with electrolyte dynamics
SVD	Singular Value Decomposition
UAV	Unmanned Aerial Vehicle
UKF	Unscented Kalman Filter
V-rep	Vertex Representation
WDN	Water Distribution Network
ZFO	First-Order extension using Zonotopes
ZFO+F	First-Order extension using Zonotopes with Feasible set
ZMV	Mean Value extension using Zonotopes
ZMV+F	Mean Value extension using Zonotopes with Feasible set

Notation

General notation

a	Italic lower case letters denote scalars
\mathbf{a}	Boldface lower case letters denote vectors
\mathbf{A}	Boldface upper case letters denote matrices

General symbols and operators

\mathbb{N}	Set of natural numbers
\mathbb{N}^+	Set of natural numbers with zero excluded
\mathbb{R}	Set of real numbers
$\mathbb{I}\mathbb{R}$	Set of real compact intervals
$\mathbf{0}_{n \times m}$	Matrix of zeros with n rows and m columns
$\mathbf{1}_{n \times m}$	Matrix of ones with n rows and m columns
\mathbf{I}_n	Identity matrix with dimension n
$\dot{\mathbf{A}}$	Time derivative of \mathbf{A}
\mathbf{A}^T	Transpose of \mathbf{A}
\mathbf{A}^{-1}	Inverse of \mathbf{A}
\mathbf{A}^+	Pseudo-inverse of \mathbf{A}
\mathbf{a}_i	i -th component of \mathbf{a}
$\mathbf{A}_{i,:}$	i -th row of \mathbf{A}
$\mathbf{A}_{:,j}$	j -th column of \mathbf{A}
trace (\mathbf{A})	Trace of \mathbf{A}

$\text{diag}(\mathbf{a})$	Diagonal matrix whose diagonal is defined by \mathbf{a}
$\text{blkdiag}(\mathbf{A}, \dots)$	Block diagonal matrix with blocks \mathbf{A}, \dots
$\vec{\mathbf{a}}$	Sequence of \mathbf{a} defined over a given time interval
Z^+	Lifted zonotope for Z
$\text{mid}(\mathbf{A})$	Midpoint of \mathbf{A}
$\text{diam}(\mathbf{A})$	Diameter of \mathbf{A}
$\text{rad}(\mathbf{A})$	Radius of \mathbf{A}
$\nabla_x \mathbf{f}$	Gradient of \mathbf{f} with respect to \mathbf{x}
$\mathbf{H}_x \mathbf{f}$	Upper triangular matrix denoting half the Hessian of \mathbf{f} with respect to \mathbf{x}
$\square(\mathbf{f}(\cdot))$	Inclusion function of \mathbf{f}
$\square X$	Interval hull of X
\oplus	Minkowski sum
$\cap_{\mathbf{A}}$	Generalized intersection with linear mapping \mathbf{A}
$\cap_{\mathbf{A}_1, \mathbf{A}_2}$	Two-side generalized intersection with linear mappings \mathbf{A}_1 and \mathbf{A}_2
$\triangleleft(\cdot, \cdot)$	CZ-inclusion operator
$\{\mathbf{G}, \mathbf{c}\}$	Zonotope with center \mathbf{c} and generators \mathbf{G}
$\{\mathbf{G}, \mathbf{c}, \mathbf{A}, \mathbf{b}\}$	Constrained zonotope with center \mathbf{c} , generators \mathbf{G} , and constraints (\mathbf{A}, \mathbf{b})
$\{\mathbf{M}, \mathbf{G}, \mathbf{c}, \mathbf{S}, \mathbf{A}, \mathbf{b}\}$	Mixed zonotope with center \mathbf{c} , bounded generators \mathbf{G} , unbounded generators \mathbf{M} , and constraints $(\mathbf{S}, \mathbf{A}, \mathbf{b})$

Contents

List of Figures	11
List of Tables	16
Acronyms	18
Notation	21
1 Introduction	29
1.1 Motivation	29
1.1.1 State estimation	29
1.1.2 Fault diagnosis	31
1.1.3 Descriptor systems	32
1.2 Justification	32
1.3 Objectives	33
1.3.1 Specific Objectives	33
1.4 Structure of the text	34
1.5 List of publications	36
2 Literature review	39
2.1 Reachability analysis and set-based state estimation of nonlinear systems	39
2.2 Set-based fault diagnosis of nonlinear systems	42
2.3 Set-based state estimation and fault diagnosis of descriptor systems	44
2.4 Set-based joint state and parameter estimation	45
2.5 Final remarks	46
3 Mathematical background	47
3.1 Basic set operations and notation	47
3.2 Zonotopes and constrained zonotopes	48
3.3 Complexity reduction and rescaling	50
3.4 Interval analysis	54

3.4.1	Partitioning	55
3.5	Linear state estimation	56
3.6	Final remarks	57
4	State estimation of systems with nonlinear dynamics	58
4.1	Problem formulation	58
4.2	Prediction step	59
4.2.1	CZIB approach	60
4.2.1.1	Outline and assumptions	60
4.2.1.2	Intermediate steps	61
4.2.2	Mean value extension	65
4.2.2.1	Selection of approximation point	67
4.2.3	First-order Taylor extension	70
4.2.3.1	Selection of approximation point	73
4.3	Update step	75
4.4	Complexity analysis	76
4.5	Numerical examples	78
4.5.1	Example 1	78
4.5.2	Example 2	79
4.6	Final remarks	84
5	State estimation of systems with nonlinear measurement and invariants	86
5.1	Problem formulation	87
5.2	Recursive algorithm	88
5.2.1	Prediction step	90
5.2.2	Update step	92
5.2.3	Consistency step	94
5.2.4	Selection of approximation point	96
5.2.5	Linear case	97
5.3	Computational complexity	98
5.4	Numerical examples	99
5.4.1	A system with nonlinear measurement equations	100
5.4.2	A system with nonlinear measurements and invariants	102
5.5	Final remarks	105
6	Fault diagnosis of nonlinear systems	108
6.1	Problem formulation	108
6.2	Passive fault detection	109
6.3	Reachable sets	109
6.3.1	Guaranteed enclosure	110

6.3.2	Affine parameterization of the reachable set	111
6.4	Input design for active fault diagnosis	113
6.4.1	Zonotope method	113
6.4.2	Constrained zonotope method	116
6.5	Numerical example	118
6.6	Final remarks	120
7	Descriptor systems	122
7.1	Problem formulation	123
7.1.1	State estimation	123
7.1.2	Fault diagnosis	123
7.1.3	Motivational example	123
7.2	Set-based state estimation of LDS using constrained zonotopes	125
7.3	Active fault diagnosis of LDS using constrained zonotopes	127
7.3.1	State and output reachable sets	128
7.3.2	Input design	129
7.4	Mixed zonotopes	131
7.4.1	Complexity reduction of mixed zonotopes	134
7.4.1.1	Elimination of unbounded generators	134
7.4.1.2	Constraint elimination	134
7.4.1.3	Elimination of bounded generators	135
7.4.2	Concept examples	135
7.4.2.1	Exact intersection of a zonotope and a strip	135
7.4.2.2	Unbounded intersection between strips	136
7.4.2.3	Two-side generalized intersection	136
7.5	Set-based state estimation of LDS using mixed zonotopes	137
7.6	Active fault diagnosis of LDS using mixed zonotopes	140
7.6.1	State reachable tube	141
7.6.2	Output reachable tube	144
7.6.3	Separation of output reachable tubes	145
7.7	Numerical examples	147
7.7.1	State estimation	147
7.7.2	Fault diagnosis using constrained zonotopes	149
7.7.3	Fault diagnosis using mixed zonotopes	150
7.8	Final remarks	151
8	Joint state and parameter estimation	152
8.1	Problem formulation	152
8.2	Proposed scheme based on constrained zonotopes	154
8.3	Numerical example	158

8.3.1	Example 1	158
8.3.2	Example 2	161
8.4	Final remarks	164
9	Application: Unmanned aerial vehicles	166
9.1	Linear state estimation applied to aerial load transportation	166
9.1.1	Tilt-rotor UAV with suspended load modeling	168
9.1.1.1	Equations of motion	168
9.1.1.2	Measurement equation	170
9.1.1.3	Linearized model for state estimation	172
9.1.2	Problem formulation	172
9.1.3	Minimum-variance feedback	173
9.1.4	Path tracking control	174
9.1.4.1	Linearized parameter-varying error dynamics	174
9.1.4.2	Discrete-time mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control with pole placement constraints	175
9.1.5	Numerical experiments	176
9.2	Nonlinear state estimation of a quadrotor UAV	179
9.3	Fault diagnosis of a quadrotor UAV	183
9.4	Final remarks	185
10	Application: Water distribution networks	187
10.1	Introduction	188
10.2	Problem formulation	191
10.3	Set-based state estimation of WDNs	192
10.3.1	Iterative Hydraulic Interval State Estimation	192
10.3.1.1	Initial bounds on the state vector	192
10.3.1.2	Bounding linearization of the interval nonlinear terms	192
10.3.1.3	Solution of the linear interval system of equations using linear programming	193
10.3.1.4	Iterative solution of the nonlinear system of equations	193
10.3.2	Slope approximation of the nonlinear equalities	194
10.3.3	Iterative Slope Approximation and Rescaling	195
10.3.4	Final enclosure with constrained zonotopes	197
10.4	Passive leak detection in WDNs	198
10.5	Numerical experiments	199
10.5.1	Two-pipe example	199
10.5.2	Transport network	203
10.5.3	CY-DMA network	205
10.6	Final remarks	209

11 Application: Li-Ion batteries	211
11.1 Introduction	211
11.2 Lithium-ion cell modeling	213
11.2.1 Single Particle Model With Electrolyte dynamics	213
11.2.2 Equivalent Circuit Model	215
11.3 Identification of the ECM parameters	218
11.4 Set-based state estimation method	221
11.4.1 Problem formulation	222
11.4.2 State estimation of the Lithium-ion cell using FBCP	223
11.5 Numerical results	224
11.6 Final remarks	226
12 Conclusions	227
12.1 Overview and contributions	227
12.2 Future work	231
Bibliography	234
A Derivation of computational complexities	252
A.1 Zonotopes	252
A.1.1 Basic operations	252
A.1.1.1 Linear image	252
A.1.1.2 Minkowski sum	252
A.1.1.3 Generator reduction	253
A.1.1.4 Interval hull	253
A.1.1.5 Zonotope inclusion	253
A.1.2 ZMV	253
A.1.2.1 Prediction step	253
A.1.2.2 Update step	254
A.1.2.3 Order reduction	255
A.1.3 ZFO	256
A.1.3.1 Prediction step	256
A.1.3.2 Update step	256
A.1.3.3 Order reduction	256
A.2 Constrained zonotopes	257
A.2.1 Basic operations	257
A.2.1.1 Linear image	257
A.2.1.2 Minkowski sum	257
A.2.1.3 Generalized intersection	257
A.2.1.4 Generator reduction	258

A.2.1.5	Constraint elimination	258
A.2.1.6	Interval hull (Property 1)	259
A.2.1.7	CZ-inclusion (Theorem 1)	259
A.2.1.8	Closest point (Proposition 1)	259
A.2.1.9	Change center (Proposition 2)	259
A.2.2	CZMV	260
A.2.2.1	Prediction step (Theorem 2)	260
A.2.2.2	Update step	261
A.2.2.3	Order reduction	261
A.2.3	CZFO	262
A.2.3.1	Prediction step	262
A.2.3.2	Update step	263
A.2.3.3	Order reduction	263

1

Introduction

1.1 Motivation

In view of the required reliability in many applications, the development of set-based methods has gained attention in the last decades in a wide range of applications ([Jaulin et al., 2001b](#); [Blanchini & Miani, 2015](#)). These are usually characterized by their robustness and reliability, since the obtained results are often guaranteed in a formal manner. These methods are based on computation with sets, either under set-based concepts such as positive invariance, set-membership methodology, or through numerical tools such as interval arithmetic ([Moore et al., 2009](#)). Recent works address the problems of robust control ([Le et al., 2011](#); [Polyakov et al., 2013](#); [Ping & Sun, 2012](#)), robust state estimation ([Bars et al., 2012](#); [Jaulin, 2012, 2016](#); [Mazenc et al., 2013](#); [Chabane et al., 2014b,c,d](#)), parameter identification ([Herrero et al., 2016](#)), robot navigation ([Jaulin & Bars, 2013](#)), robot localization ([Neuland et al., 2014](#); [Yu et al., 2016](#)), and collision avoidance ([Zhou & Baras, 2015](#)), for instance.

1.1.1 State estimation

In recent decades, the importance of state estimation has gained attention in many fields of research ([Simon, 2006](#)). This includes a wide range of applications such as state-feedback control ([Jaulin, 2009b](#); [Goodarzi & Lee, 2017](#); [Rego & Raffo, 2019](#)), fault diagnosis ([Zhang & Jiang, 2008](#); [Combastel, 2015a](#); [Raimondo et al., 2016](#)), and robot localization

(Saeedi et al., 2016). In contrast to Bayesian strategies such as Kalman filtering (Teixeira et al., 2009; Simon, 2010), set-valued state estimation methods aim to provide guaranteed enclosures of the system trajectories in applications affected by unknown-but-bounded uncertainties, without assuming knowledge of their stochastic properties (Schweppe, 1968; Chisci et al., 1996).

The set-based state estimation problem has been extensively studied for linear discrete-time systems. The pioneering methods make use of ellipsoids to bound the trajectories of the system (Schweppe, 1968). Other classical methods propose recursive state estimation algorithms based on parallelotopes (Chisci et al., 1996) and also zonotopes (Combastel, 2003). More recent works combine different classes of sets for set-valued estimation, such as ellipsoids and zonotopes (Chabane et al., 2014a), and even combinations of zonotopic methods with Kalman filtering can be found (Combastel, 2015a). However, since these sets are not closed in every set operation that arises in the state estimation problem, recently, Scott et al. (2016) use a generalization of zonotopes, called *constrained zonotopes*, to overcome most of the difficulties encountered by the former algorithms. Whereas the state estimation problem is already consolidated for linear systems, nonlinear state estimation is still an open field, both for stochastic or set-based approaches. The exact characterization of sets containing the evolution of the system states is very difficult in the nonlinear case, if not intractable (Kieffer et al., 1998; Kühn, 1998; Platzer & Clarke, 2007). Therefore, in the set-membership framework the objective is to enclose such sets as tightly as possible by guaranteed outer bounds on the possible trajectories of the system states. Such outer bounds must be consistent with the previous estimate, known inputs, the current measurement, the bounds on the disturbances and uncertainties, and also bounds on the initial states.

However, achieving accurate enclosures for nonlinear systems remains a significant challenge. When these enclosures are represented by simple sets such as intervals, ellipsoids, parallelotopes, and zonotopes, certain set operations can be very conservative. Yet, using general convex polytopes is much more computationally demanding. A common tool for computing these enclosures for nonlinear systems is *interval analysis*. With a wide range of applications, such as global optimization, parameter estimation and robust control (Jaulin et al., 2001b), interval analysis can be used to generate guaranteed bounds on the range of real valued functions, through interval extensions (Moore et al., 2009). However, severe overestimation often occurs due to interval dependency and the wrapping effect (Kühn, 1998). The latter is associated to the inability of intervals to capture dependencies between variables resulting from multivariate mappings, while the former arises from multiple occurrences of an interval variable in the same algebraic expression. Such problems are mitigated by computing unions of interval extensions over a partitioned domain (*refinements*). Nonetheless, set inversion algorithms often are required to take measurements into account in interval-based observers, which can be very expensive (Jaulin,

2009b).

1.1.2 Fault diagnosis

The requirements on reliability and safety of control systems under actuator and sensor malfunctions motivated the development of Fault-Tolerant Control (FTC) strategies since the 70's (Montgomery & Caglayan, 1976; Montgomery & Price, 1976; Chizeck & Willsky, 1978). Several techniques were developed to cope with faults in a great variety of systems and fault scenarios since then (Zhang & Jiang, 2008). The purpose of such strategies is to accommodate component malfunctions by maintaining the overall system stable with acceptable closed-loop performance in fault scenarios, changing control objectives if necessary to cope with physical limitations imposed by the faults, while performing the original control task for the nominal plant. In particular, these control strategies have been of utmost importance in safety critical systems, such as nuclear power plants and aerospace systems, in which minor faults in components may lead to catastrophic consequences if not dealt with appropriately (Hatami et al., 2016; Castaldi et al., 2014).

Existing FTC strategies can be classified into two main categories: passive fault-tolerant control and active fault-tolerant control (Zhang & Jiang, 2008; Fekih, 2014). Passive strategies are based on fixed robust controllers, designed to ensure that the closed-loop system remains insensitive to a predetermined repertory of faults. These strategies are characterized by their reduced complexity and are independent of Fault Diagnosis (FD) techniques. However, these methods are very conservative and capable of accommodating only a greatly limited number of fault scenarios. Examples of passive FTC strategies are based on linear quadratic control (Hsieh, 2002), sliding mode control (Alwi et al., 2008; Hu, 2010; Shen et al., 2015), linear \mathcal{H}_∞ control (Tao et al., 2015), and also fuzzy \mathcal{H}_∞ control (Li et al., 2012).

In view of the required reliability in FTC systems, the development of set-based fault diagnosis (FD) methods has gained attention in the last decades (Xu et al., 2014; Raimondo et al., 2016). Fault diagnosis aims to determine exactly which fault a process is subject to. Set-based FD methods are usually characterized by robustness and reliability, as diagnosis is often guaranteed. These methods are based on computation with sets, either under concepts such as positive invariance, consistency tests and set separation. Passive fault detection methods are quite common in the literature (Guerra et al., 2008; Blesa et al., 2012; Xu et al., 2014; Vento et al., 2015); however, passive fault detection alone cannot provide information on which fault occurred, in order to mitigate any harmful effects resulting from the fault. On the other hand, active fault diagnosis (AFD) methods allow to determine which fault has occurred, usually by the injection of an optimal input sequence in order to identify the fault (Scott et al., 2014). Set-based fault diagnosis often relies on the separation of reachable sets for models describing all the possible faulty

scenarios. Therefore, high conservatism in the computation of these enclosures results in poor performance of the fault diagnosis procedure. Reduction of computational burden is another investigated issue in the literature, for instance by using offline computations based on multi-parametric programming (Marseglia & Raimondo, 2017).

1.1.3 Descriptor systems

Another object of application of set-based state estimation and fault diagnosis involves many physical processes such as battery packs, robotic systems with holonomic and nonholonomic constraints, and socioeconomic systems (Janschek, 2011; Yang et al., 2019), which exhibit static relations between their internal variables. These processes are known as descriptor systems (or implicit systems), which have generalized dynamic and static behaviors described through differential and algebraic equations, respectively (Wang et al., 2018a). Descriptor systems appear in several contexts, such as linear control (Bender & Laub, 1987), fault-tolerant control (Shi & Patton, 2014), and fault diagnosis (Wang et al., 2019). However, few strategies can deal effectively with state estimation and fault diagnosis of descriptor systems when uncertainties with unknown probability distribution are present (Hamdi et al., 2012).

Set-based strategies for state estimation of discrete-time descriptor systems with unknown-but-bounded uncertainties is a recent subject, often addressed using intervals, zonotopes and ellipsoids (Efimov et al., 2015; Wang et al., 2018a,b; Merhy et al., 2019). Interval methods are used in Efimov et al. (2015) to design a state estimator for time-delay descriptor systems based on linear matrix inequalities and the Luenberger structure. However, interval arithmetics can lead to conservative enclosures due to the wrapping effect. A few methods are proposed in Wang et al. (2018a) for set-based state estimation of linear descriptor systems by enclosing the intersection of two consistent sets with a zonotope bound. Nevertheless, since zonotopes cannot effectively capture the algebraic constraints (which are typical of descriptor systems), and also the intersection cannot be computed exactly, the resulting bound is conservative.

1.2 Justification

As commented in the previous section, set-based methods have gained attention in the last decades in a wide range of applications due to their robustness and reliability, since the obtained results are often guaranteed. Nevertheless, while set-based state estimation and fault diagnosis are relatively consolidated for linear systems, set-based state estimation and fault diagnosis of nonlinear discrete-time systems is still an open field.

Set-based estimation aims to compute tight enclosures of the possible system states in each time step subject to unknown-but-bounded uncertainties. Most existing methods

employ a standard prediction-update framework with set-based prediction and update steps based on various set representations and techniques. However, achieving accurate enclosures for nonlinear systems remains a significant challenge. When these enclosures are represented by simple sets such as intervals, ellipsoids, parallelotopes, and zonotopes, certain set operations can be very conservative. Yet, using general convex polytopes is much more computationally demanding. In addition, the set-based fault diagnosis considered in this doctoral thesis relies on the separation of reachable sets for all the possible models describing faulty scenarios. Therefore, high conservatism in the computation of these enclosures result in poor performance of the fault diagnosis procedure.

In view of the few available methods in the literature, and considering that both set-based state estimation and fault diagnosis are still open fields for several classes of discrete-time dynamical systems, this doctoral thesis focuses on the investigation of new methods for set-based state estimation and fault diagnosis. These explore the benefits of using constrained zonotopes as main set representation, which already presented advantages in set-based linear state estimation in previous works in comparison to other simple set representations.

1.3 Objectives

The main objective of this doctoral thesis is to investigate the development of new methods for set-based state estimation and fault diagnosis for a few classes of discrete-time dynamical systems, exploring the benefits of using constrained zonotopes as main set representation.

1.3.1 Specific Objectives

Specifically, it is intended to investigate the following issues:

1. Development of set-based state estimation methods for a class of discrete-time systems with nonlinear dynamics using constrained zonotopes.
2. Development of set-based state estimation methods for a class of discrete-time systems with nonlinear dynamics, nonlinear measurement, and nonlinear invariants, using constrained zonotopes.
3. Development of set-based fault diagnosis methods for a class of discrete-time systems with nonlinear dynamics using constrained zonotopes.
4. Development of set-based state estimation and fault diagnosis methods for linear descriptor systems using constrained zonotopes.

5. Development of a new set representation to describe unbounded sets, aiming the design of new methods for set-based state estimation and fault diagnosis for linear descriptor systems.
6. Development of set-based joint state and parameter estimation methods using constrained zonotopes for a class of nonlinear systems with nonlinear dynamics, nonlinear measurement, and nonlinear algebraic constraints.
7. Application of the developed strategies in practical scenarios. In this thesis, unmanned aerial vehicles, water distribution networks, and a Lithium-ion cell, were considered.

1.4 Structure of the text

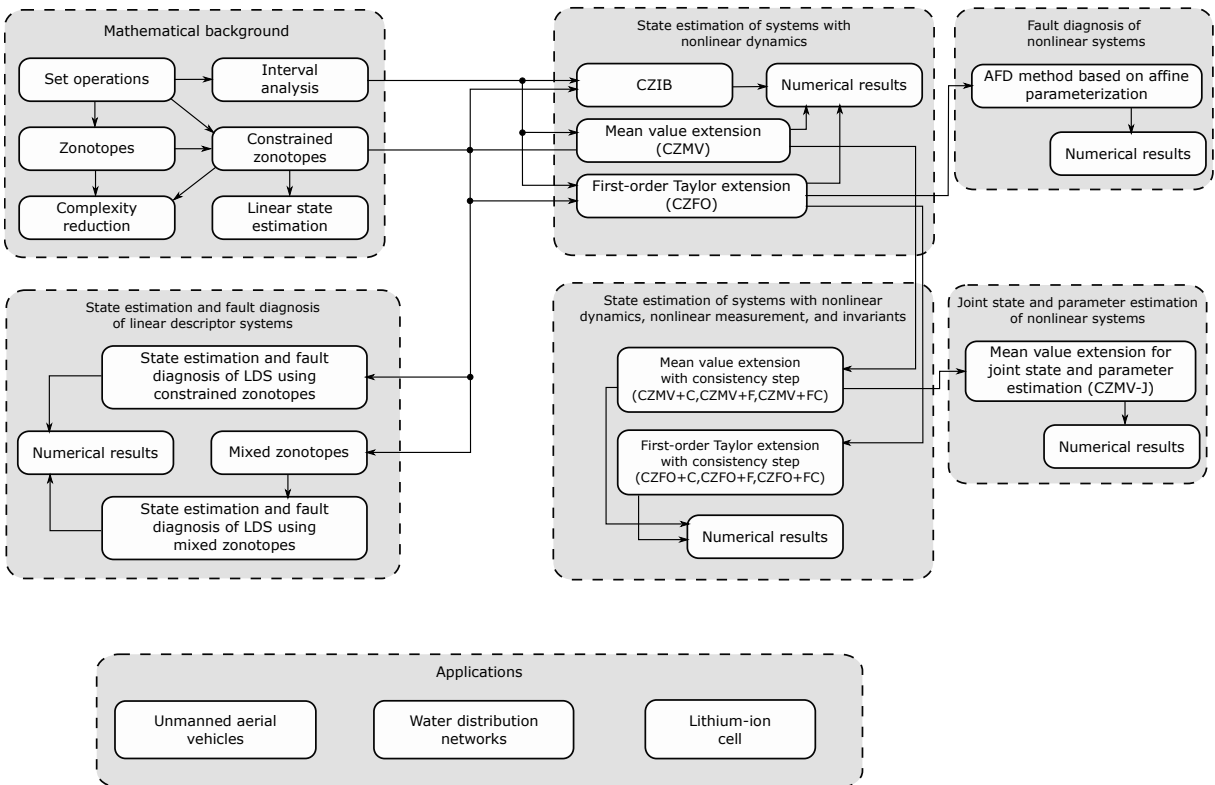


Figure 1.1: Flow chart of this doctoral thesis.

Figure 1.1 shows a flow chart used in the development of this doctoral thesis. This doctoral thesis is organized as follows:

- **Chapter 2** presents a literature review highlighting the state of the art in set-based state estimation and fault diagnosis of nonlinear discrete-time systems and descriptor systems.

- **Chapter 3** presents the necessary mathematical background, concepts, and definitions used in this doctoral thesis. The following topics are introduced: (i) basic set operations; (ii) zonotopes; (iii) constrained zonotopes; (iv) complexity reduction methods; (v) interval analysis; and (iv) linear state estimation using constrained zonotopes.
- **Chapter 4** introduces new state estimation methods based on constrained zonotopes for discrete-time systems with nonlinear dynamics and linear measurement: (i) the constrained zonotope and interval bundle method; (ii) the mean value extension and (iii) the first-order Taylor extension using constrained zonotopes, which are generalizations of two existing zonotopic methods.
- **Chapter 5** introduces new state estimation methods based on constrained zonotopes for discrete-time systems with nonlinear dynamics, nonlinear measurements, and nonlinear invariants. The proposed methods improve the standard prediction-update framework for systems with invariants by adding a consistency step (which uses invariants to reduce conservatism), besides presenting significant improvements to prediction and update steps proposed in Chapter 4, and allowing nonlinear measurement equations for the first time.
- **Chapter 6** develops a new approach for set-based active fault diagnosis for discrete-time systems with nonlinear dynamics using zonotopes and constrained zonotopes. Passive fault detection is proposed based on the new enclosures obtained in Chapter 4. An adaptation of the first-order Taylor extension is used for the reachable set parameterization, which leads to the development of an optimal method for input design for guaranteed set-based active fault diagnosis.
- **Chapter 7** presents new methods based on constrained zonotopes and unbounded sets for set-valued state estimation and active fault diagnosis of linear descriptor systems: (i) linear static constraints on the state variables are directly incorporated in the mathematical description of constrained zonotopes, leading to less conservative enclosures; and (ii) new methods for state estimation and active fault diagnosis of linear descriptor systems are developed based on a new representation for unbounded sets, without assuming the knowledge of an enclosure of all the possible trajectories of the system or the initial states.
- **Chapter 8** develops a new method for set-based joint state and parameter estimation of discrete-time systems with nonlinear dynamics, nonlinear measurement, and nonlinear algebraic constraints. The state estimation method proposed in Chapter 5 is extended to include parameter estimation in a unified framework, and therefore maintaining existing dependencies between states, algebraic variables, and unknown parameters.

- **Chapter 9** addresses the application to unmanned aerial vehicles (UAVs): (i) a set-based linear state estimation using constrained zonotopes considering different sampling times, applied in state-feedback control for trajectory tracking of a suspended load using a tilt-rotor UAV; application of the (ii) state estimation and (iii) fault diagnosis methods developed in Chapters 4 and 6 to a quadrotor UAV, respectively.
- **Chapter 10** proposes two new methods for set-based state estimation and fault detection in water distribution networks (WDNs): (i) a new interval method is proposed based on iterative computation of tight enclosures of the nonlinear head-loss functions, and bounding the solution of the algebraic equations using rescaling; (ii) since intervals are not capable of capturing the dependencies between state variables, constrained zonotopes are used as an additional step, leading to two new algorithms capable of capturing the dependencies between hydraulic states.
- **Chapter 11** develops a discrete-time interval state estimator for a Lithium-ion cell, based on the forward-backward constraint propagation. The contributions include also the identification of parameter uncertainties of the electrical chemical model based on the Fisher Information Matrix. In practice, only cell current and voltage are usually measurable, and therefore, the model states need to be estimated. The proposed interval estimator provides a tight enclosure of the cell states, which is essential for fault detection and model-based control of a Lithium-ion cell.
- **Chapter 12** summarizes the main contributions developed in this doctoral thesis, and presents proposals for future work.
- **Appendix A** details the computational complexities of some of the state estimation methods developed in this thesis.

1.5 List of publications

The following articles were published, accepted for publication, or submitted during the elaboration of this doctoral thesis:

Journal papers:

1. ([Santos et al., 2017b](#)) Santos, M. A., Rego, B. S., Raffo, G. V., & Ferramosca, A. (2017b). Suspended load path tracking control strategy using a tilt-rotor UAV. *Journal of Advanced Transportation*, 2017, 1–22.

2. (Rego & Raffo, 2019) Rego, B. S. & Raffo, G. V. (2019). Suspended load path tracking control using a tilt-rotor UAV based on zonotopic state estimation. *Journal of the Franklin Institute*, 356(4), 1695–1729.
3. (Rego et al., 2020a) Rego, B. S., Raffo, G. V., Scott, J. K., & Raimondo, D. M. (2020a). Guaranteed methods based on constrained zonotopes for set-valued state estimation of nonlinear discrete-time systems. *Automatica*, 111, 108614.
4. (Rego et al., 2021d) Rego, B. S., Scott, J. K., Raimondo, D. M., & Raffo, G. V. (2021a). Set-valued state estimation of nonlinear discrete-time systems with nonlinear invariants based on constrained zonotopes. *Automatica*, 129, 109638.
5. (Rego et al., 2021e) Rego, B. S., Vrachimis, S. F., Polycarpou, M. M., Raffo, G. V., & Raimondo, D. M. (2021b). State estimation and leakage detection in water distribution networks using constrained zonotopes. *IEEE Transactions on Control Systems Technology*. Minor revision after first round.
6. (Rego et al., 2021c) Rego, B. S., Raimondo, D. M., & Raffo, G. V. (2021c). Set-based state estimation and active fault diagnosis of linear descriptor systems using mixed zonotopes. To be submitted.
7. (Rego et al., 2021a) Rego, B. S., Locatelli, D., Raimondo, D. M., & Raffo, G. V. (2021a). Set-based joint state and parameter estimation of discrete-time systems using constrained zonotopes. To be submitted.
8. (Rego et al., 2021b) Rego, B. S., Raimondo, D. M., & Raffo, G. V. (2021b). Set-based joint state and parameter estimation of nonlinear systems with nonlinear algebraic constraints using constrained zonotopes. To be submitted.

Conference papers:

1. (Lara et al., 2017) Lara, A. V., Rego, B. S., Raffo, G. V., & Arias-Garcia, J. (2017). Desenvolvimento de um ambiente de simulação de VANTs tilt-rotor para testes de estratégias de controle. In *Anais do XIII Simpósio Brasileiro de Automação Inteligente* (pp. 2135–2141).
2. (Santos et al., 2017a) Santos, M. A., Cardoso, D. N., Rego, B. S., Raffo, G. V., & Esteban, S. (2017a). A discrete robust adaptive control of a tilt-rotor UAV for an enlarged flight envelope. In *Proc. of the 56th IEEE Conference on Decision and Control* (pp. 5208–5214).
3. (Rego et al., 2018a) Rego, B. S., Raimondo, D. M., & Raffo, G. V. (2018a). Path tracking control with state estimation based on constrained zonotopes for aerial load

- transportation. In *Proc. of the 57th IEEE Conference on Decision and Control* (pp. 1979–1984).
4. (Rego et al., 2018b) Rego, B. S., Raimondo, D. M., & Raffo, G. V. (2018b). Set-based state estimation of nonlinear systems using constrained zonotopes and interval arithmetic. In *Proc. of the 2018 European Control Conference* (pp. 1584–1589).
 5. (Rego et al., 2020b) Rego, B. S., Raimondo, D. M., & Raffo, G. V. (2020b). Set-based state estimation and fault diagnosis of linear discrete-time descriptor systems using constrained zonotopes. In *Proc. of the 1st IFAC-V World Congress* (pp. 4291-4296).
 6. (Locatelli et al., 2021) Locatelli, D., Rego, B. S., Raffo, G. V., & Raimondo, D. M. (2021). Interval state estimation based on constraint propagation for a lithium-ion cell using an equivalent circuit model. In *Proc. of the 11th IFAC International Symposium on Advanced Control of Chemical Processes*. Accepted.

2

Literature review

This chapter presents a detailed literature review on the main topics of interest in this doctoral thesis, namely set-based state estimation and set-based fault diagnosis of nonlinear systems. Brief literature reviews on set-based state estimation and fault diagnosis of descriptor systems, and joint state and parameter estimation, are also depicted.

2.1 Reachability analysis and set-based state estimation of nonlinear systems

Reachability analysis aim to construct compact sets that are guaranteed to enclose all possible trajectories of the system subject to unknown-but-bounded uncertainties. Interval analysis, proposed in [Moore \(1966\)](#), is a classic tool for computing these enclosures for nonlinear systems, with a wide range of other applications, such as global optimization, parameter estimation, and robust control ([Jaulin et al., 2001b](#)). Nevertheless, severe conservatism may can occur in the computation of enclosures using intervals due to interval dependency and the wrapping effect.

[Kühn \(1998\)](#) proposes methods to enclose the trajectories of discrete-time nonlinear systems using zonotopes, which are shown to be capable of notably mitigating the wrapping effect. [Girard et al. \(2006\)](#) use zonotopes to compute the exact reachable sets of linear discrete-time systems efficiently. Moreover, zonotopes are further used in [Althoff et al. \(2008\)](#) to bound the trajectories of nonlinear continuous-time systems by conservative

linearization; however, in contrast to the linear discrete-time case, the method still results in substantial overestimation for systems with highly nonlinear behavior.

Seeking the reduction of conservatism in the nonlinear case, non-convex sets denoted by polynomial zonotopes are developed in [Althoff \(2013\)](#) to bound the trajectories of nonlinear systems through conservative polynomialization. Nevertheless, the complexity of the representation described by polynomial zonotopes grows polynomially each time step, which is mitigated by the development of a sparse description in [Kochdumper & Althoff \(2020c\)](#) (called sparse polynomial zonotopes). Later, [Kochdumper & Althoff \(2020a\)](#) extend the latter to constrained polynomial zonotopes, which allow the efficient computation of several set operations exactly for the first time (e.g., set union) using a single set.

Alternative techniques for reachability analysis include ellipsoidal methods ([Kurzhanski & Varaiya, 2000](#)); however, despite the efficiency of such methods, due to the limited complexity of ellipsoids, the resulting enclosure is often conservative even in the linear case. In addition, other techniques for reachability analysis consist of using Taylor models ([Chen et al., 2012](#)), which are based on approximating the nonlinear function by high-order Taylor expansions, allowing to combine different set representations such as intervals and zonotopes. Support functions are used in [Girard & Guernic \(2008\)](#) for reachability analysis of linear discrete-time systems, and in [Frehse et al. \(2015\)](#) for linear continuous-time systems; however, in practice, these methods are limited to approximation of the true set by half-spaces. Other frameworks include the use of Bernstein expansions ([Dang & Testylier, 2012](#)) to outer-approximate the reachable sets of polynomial systems by convex sets, while Hamilton-Jacobi methods have been proposed in ([Mitchell et al., 2005](#)) to enclose the trajectories of nonlinear continuous-time systems through differential game formulation.

Set-based state estimation, using the standard recursive approach, involves first bounding the image of the current enclosure under the dynamics (prediction), and then enclosing the intersection of this set with the set of states consistent with a bounded-error measurement (update). For linear discrete-time systems, exact enclosures can be computed as convex polytopes. However, even for linear dynamics, polytope propagation requires demanding computations (e.g., polytope projection, Minkowski addition, or conversion between vertex and half-space representations) ([Walter & Piet-Lahanier, 1989](#); [Shamma & Tu, 1999](#)). Thus, enclosures are often described by simpler sets including ellipsoids ([Schweppe, 1968](#); [Durieu et al., 2001](#); [Polyak et al., 2004](#)), parallelotopes ([Chisci et al., 1996](#); [Vicino & Zappa, 1996](#)), zonotopes ([Le et al., 2013](#); [Combastel, 2003](#); [Wang et al., 2018b](#)), or combinations of these ([Chabane et al., 2014a](#)). However, the mathematical limitations of these sets require certain operations to be over-approximated, sometimes quite significantly. Notably, this includes set intersection, which is critical for the update step in set-based state estimation ([Chisci et al., 1996](#); [Le et al., 2013](#); [Durieu et al., 2001](#)).

Althoff & Krogh (2011) propose the use of zonotope bundles to describe intersections of zonotopes without explicit computation. However, the Minkowski sum and linear image are over-approximated. Efficient algorithms for linear set-based state estimation are proposed in Scott et al. (2016), in which constrained zonotopes are introduced to overcome many of the limitations of zonotopes. These sets are closed under intersection, Minkowski sum, and linear image, and are capable of describing arbitrary convex polytopes if the complexity of the set description is not limited.

In contrast to the linear case, effective set-based state estimation for nonlinear systems is still an open challenge (Jaulin, 2009b; Alamo et al., 2005; Combastel, 2005; Wan et al., 2018). Early approaches in this field used inclusion functions based on interval analysis (Moore et al., 2009) to propagate bounds through the nonlinear dynamics, and used interval-based set inversion techniques to enclose the set of states consistent with the current measurement (Jaulin, 2009a,b; Kieffer et al., 1998). Improved accuracy is achieved using refinements (i.e., unions of intervals) as well as more advanced interval methods such as contractor and separator algebras (Jaulin, 2009a, 2016), and constraint propagation using tube contractors (Rohou et al., 2018). Unfortunately, even these methods often provide conservative bounds without extensive refinement, which is only tractable for systems with relatively few states (Jaulin, 2009b).

A few alternatives for nonlinear set-based state estimation can be found in the literature, mostly based on zonotopes. Alamo et al. (2005) and Combastel (2005) both propose approximation procedures for propagating a zonotope through a nonlinear function. The first is based on the Mean Value Theorem and is referred to as the *mean value extension*, while the second uses a first-order Taylor expansion with a rigorous remainder bound, and is referred to as the *first-order Taylor extension*. Using these extensions for the prediction step, updates are then achieved by methods for over-approximating the intersection of a zonotope with a strip (i.e., a linear measurement with bounded error). An alternative zonotope-based prediction step using Difference of Convex functions (DC) programming is proposed in Alamo et al. (2008), but with the same update as in Alamo et al. (2005). Even for linear measurements, the symmetry of zonotopes is known to cause significant errors in the update step (Scott et al., 2016). General convex polytopes in half-space representation are used in Wan et al. (2018) to enable an exact update. Prediction is then done by representing the polytope as an intersection of zonotopes and applying the mean value extension. Unfortunately, conversion between these representations is computationally demanding, and the increasing complexity of the zonotope bundle with time is not addressed.

Still in the context of state estimation, several methods address the case of nonlinear systems whose solutions satisfy a set of potentially nonlinear equality constraints, referred to as *invariants*. The trajectories of such systems evolve on a lower-dimensional manifold embedded in the state space. This is true for many systems of practical interest, including

models of (bio)chemical reaction networks (Shen & Scott, 2017), attitude estimation in aircraft systems (Goodarzi & Lee, 2017), and the pose of the body frame in humanoids (Rotella et al., 2014). In the stochastic state estimation framework, invariants have previously been used to force the estimated states to lie on the embedded manifold (Julier & LaViola, 2010; Yang & Blash, 2009; Teixeira et al., 2009; Simon, 2010; Eras-Herrera et al., 2019). In the set-based estimation framework, the aim is to use invariants to reduce the conservatism of the enclosure computed in each time step by eliminating enclosed regions that can be proven to violate the invariants, and hence cannot contain real trajectories. Such refinement is known to be very effective at reducing conservatism in interval-based nonlinear reachability calculations (Scott et al., 2013b; Shen & Scott, 2017; Yang & Scott, 2020). To the best of the author’s knowledge, Yang & Li (2009) and Yang & Scott (2018) are the only prior studies that have used invariants in set-based state estimation. Yang & Li (2009) propose a set-valued state estimator using ellipsoids. A linear matrix inequality approach is used to design the estimator taking into account the nonlinear state equality constraints. However, the method only applies to linear dynamics, and the nonlinear state constraints must be conservatively linearized. Moreover, an effective procedure for computing rigorous and accurate linearization error bounds is not provided. Yang & Scott (2018) propose an effective method for using invariants to reduce the conservatism of a set-based state estimation method based on differential inequalities and interval analysis. However, the method is limited to systems that have been discretized by Euler approximation with a sufficiently small step size, in which the step size can also be difficult to compute and/or restrictive in some cases. Moreover, although the theory is general, the provided algorithm only applies to linear invariants and linear measurement equations.

2.2 Set-based fault diagnosis of nonlinear systems

A passive fault detection method is proposed in Guerra et al. (2008) for uncertain LPV systems based on state estimation by zonotopes. This method makes use of the zonotope methods proposed in Alamo et al. (2005), while the fault detection is performed by verifying if the intersection of the predicted zonotope with the measurement strips is empty. Another passive fault detection method is proposed in Blesa et al. (2012) for uncertain LPV systems based on consistency tests using polytopes and strips. The fault detection test is performed by verifying the intersection of a convex polytope containing all the feasible parameters and strips containing all parameters consistent with the current measurement. If this set is empty, then a fault has occurred. Later in Vento et al. (2015), a set-based FD strategy is proposed for uncertain LPV hybrid systems. FD is done using the parity space and residual evaluation with computation by zonotopes, resulting in a hybrid diagnosis automaton. Lastly, a passive fault detection method is proposed in Xu et al. (2016) combining the set-theoretic methods based on zonotope computations and

robust unknown input observers for linear discrete-time systems.

For better detectability of faults, a robust active FD strategy is proposed in [Scott et al. \(2013a\)](#) for linear discrete-time systems based on input design and later improved in [Scott et al. \(2014\)](#). The occurrence of faults is modeled as abrupt model switching. This method is based on the computation of an optimal input sequence such that the output reachable sets (described by zonotopes) associated to each model are not overlapping by the injection of the sequence into the system. The design of such an input sequence is made by solving a mixed-integer quadratic program. Therefore, fault diagnosis is guaranteed along the respective time horizon by verifying which output reachable set encloses the observed output sequence. A receding horizon approach is developed in [Raimondo et al. \(2013\)](#) as an extension of the method in [Scott et al. \(2013a\)](#) for closed-loop active FD of linear discrete-time systems. After the introduction of the constrained zonotopes in [Scott et al. \(2016\)](#), in which an effective passive fault detection method is also developed for linear systems, the open-loop active FD method in [Scott et al. \(2014\)](#) and the receding horizon approach in [Raimondo et al. \(2013\)](#) were both improved in [Raimondo et al. \(2016\)](#). These new linear active FD methods are based on constrained zonotopes, which allowed reduced conservativeness in comparison with zonotopes in the computation of the optimal input sequence to be injected into the system, with fault isolation guaranteed in a reduced time interval. Finally, to reduce the computational burden of the active FD methods proposed in [Raimondo et al. \(2016\)](#), several operations are moved to prior offline computation in the active FD approach developed in [Marseglia & Raimondo \(2017\)](#) based on multi-parametric programming. Another receding horizon approach for active FD is proposed in [Tabatabaeipour \(2015\)](#) based on optimal input design for linear discrete-time systems using convex polytopes. Nevertheless, the complexity of this active FD method scales exponentially with the system dimension due to the use of set-based state estimators based on convex polytopes.

Considering the design of reference signals instead of input sequences, extensions of the open-loop and receding horizon active FD approaches in [Raimondo et al. \(2016\)](#) are developed in [Marseglia et al. \(2017\)](#) for linear discrete-time systems considering state-feedback control. Using constrained zonotopes, the sequences of optimal reference inputs provided to the controller are obtained through the solution of a mixed-integer quadratic program. Since the injection of additional input sequences into the system is not required for fault isolation, this active FD method is more suitable for active FD of closed-loop systems. Fault isolation is performed similarly to [Raimondo et al. \(2016\)](#). On the other hand, a nonlinear active FD method is proposed in [Paulson et al. \(2014\)](#) based on input design. This method is formulated for uncertain polynomial or rational discrete-time systems, with faults modeled by abrupt model switching. Robust fault isolation is achieved based on feasibility checking of inputs and measurement for the nonlinear dynamics of each model. After convex relaxations of the problem, the design of the optimal input

sequence is performed through the solution of semidefinite or linear programs.

Lastly, [Blanchini et al. \(2018\)](#) proposes an active FD method based on separation of invariant sets for linear systems. This method is formulated for open-loop stable linear systems and is based on input design. Set computations are obtained using implicit representation of convex polytopes with later explicit characterization.

2.3 Set-based state estimation and fault diagnosis of descriptor systems

Set-based strategies for state estimation of discrete-time descriptor systems with unknown-but-bounded uncertainties is a recent subject, often addressed using intervals, zonotopes and ellipsoids ([Efimov et al., 2015](#); [Wang et al., 2018a,b](#); [Merhy et al., 2019](#)). [Efimov et al. \(2015\)](#) use interval methods to design a state estimator for time-delay descriptor systems based on linear matrix inequalities and the Luenberger structure. However, interval arithmetics can lead to conservative enclosures due to the wrapping effect. A few methods are proposed in [Wang et al. \(2018a\)](#) for set-based state estimation of linear descriptor systems by enclosing the intersection of two consistent sets with a zonotope bound. Nevertheless, since the intersection cannot be computed exactly, the resulting bound can be conservative. These strategies are extended in [Wang et al. \(2018b\)](#) for linear parameter-varying descriptor systems, but conservative enclosures are still present since the intersection method is maintained. [Merhy et al. \(2019\)](#) use ellipsoids for state estimation of linear descriptor systems based on Luenberger type observers. Despite being able to provide stable bounds, the ellipsoidal estimation can be conservative since the complexity of the set is fixed and static relations are not directly incorporated. In addition, as in [Wang et al. \(2018a\)](#), restrictive assumptions on the rank of the system matrices are required to be able to design the proposed estimator. Moreover, due to the static constraints, the reachable sets of models of a descriptor system may be asymmetric even if the initial set is symmetric, and therefore methods based on the sets above can provide conservative enclosures.

Fault diagnosis aims to determine exactly which fault a process is subject to. For the case of descriptor systems, this problem has been considered in recent works using zonotopes ([Yang et al., 2019](#); [Wang et al., 2019](#)). While [Wang et al. \(2019\)](#) explore the use of unknown input observers for robust passive fault diagnosis limited to additive faults, [Yang et al. \(2019\)](#) propose a zonotope-based method for active fault diagnosis (AFD) of descriptor systems. The latter is based on the design of an input sequence for separation of the reachable sets. Unfortunately, the generator representation of zonotopes cannot incorporate exactly static relations between the state variables in general linear descriptor systems. Consequently, in practice, this may lead to a more difficult diagnosis.

2.4 Set-based joint state and parameter estimation

Set-based estimation has been widely used also in the parameter identification field, as an alternative to stochastic methods such as least squares or maximum likelihood, since it is able to provide guaranteed enclosures of the model parameters, when the uncertain model parameters have unknown stochastic properties but known bounds.

Zonotopes have been used to approximate the parametric set (i.e., the set of parameters consistent with the model and noisy measurements) for discrete-time systems with additive uncertainties in [Bravo et al. \(2006\)](#), which was later extended to allow multiplicative uncertainties in [Wang et al. \(2017\)](#). However, both methods are applied only to systems described by regression models, and rely on conservative intersections with strips to refine the parametric set. Intervals have been used in the context of optimal design of experiments in [Denis-Vidal et al. \(2019\)](#) and [Mukkula & Paulen \(2017\)](#), to minimize the conservatism of the parametric enclosure. Moreover, a bisection-based interval algorithm has been used in [Rumschinski et al. \(2010\)](#) to deal with non-convex parameter sets using collections of intervals. Nevertheless, intervals are not able to capture dependencies between parameters, which may result in conservative enclosures due to wrapping effect.

Few state estimation strategies in the literature refine online the model parameter uncertainties in order to improve the accuracy of state estimation. Such methodology is referred to as *joint state and parameter estimation*. Stochastic approaches have been used in order to take into account state and parameters dependencies (covariances for Gaussian filters) in an augmented state space. A Kalman filtering strategy, based on multi-innovation recursive extended least squares algorithm has been proposed in [Cui et al. \(2020\)](#) to enhance the parameters estimation, while an iterative ensemble Kalman smoother is designed in [Bocquet & Sakov \(2013\)](#). However, as in state estimation, bias issues introduced by Kalman filtering make such approaches unreliable in case the assumptions on the stochastic properties of the uncertainties are violated. Deterministic approaches include Luenberger-based observers ([Zhang et al., 2020b](#)) and set-based interval estimation ([Raissi et al., 2004](#)). [Raissi et al. \(2004\)](#) propose a prediction-update state and parameter estimator suitable for nonlinear continuous-time systems. However, besides not being able to capture the dependencies between states and parameters, the method can lead to high computational complexity. Moreover, [Chen & Brdys \(1995\)](#) propose an estimation method based on mixed-integer programming to obtain the smallest interval enclosing the states and parameters consistent with measurements. However, this requires the solution to an optimization problem, leading to high computational burden.

2.5 Final remarks

This chapter presented a literature review on the main topics of interest in this doctoral thesis, namely set-based state estimation and set-based fault diagnosis of nonlinear systems, set-based state estimation and fault diagnosis of descriptor systems, and joint state and parameter estimation.

As commented in Chapter 1, and based on the literature review presented in this chapter, achieving accurate enclosures for the states of nonlinear systems remains a significant challenge. When these enclosures are represented by simple sets such as intervals, ellipsoids, parallelotopes, and zonotopes, certain set operations can be very conservative. Yet, using general convex polytopes is much more computationally demanding. Also, several set-based state estimation methods in the literature assume linear measurement, which is not true in most practical scenarios. Besides, in the presence of invariants, the existing methods are not able to incorporate this information efficiently due to the simple set representations used.

In addition, in set-based fault diagnosis, most methods in the literature are limited to linear discrete-time systems. It remains a challenge to develop fault diagnosis methods for nonlinear discrete-time systems, which make use of conservative reachable sets if the available methods for reachability analysis are employed.

Moreover, the techniques available for set-based state estimation and fault diagnosis of linear descriptor systems make use of the simple sets mentioned above. However, those set representations are not able to effectively incorporate equality constraints, typical of descriptor systems, in the set description, resulting in significant conservatism.

Lastly, the methods available for set-based joint state and parameter estimation in the literature are not able to effectively take into account the dependencies between states and parameters. This disadvantage is a result from the limitations of the employed set representations (mostly intervals), which suffer from severe wrapping effect.

3

Mathematical background

This chapter presents basic concepts and definitions required in the rest of this doctoral thesis. The following topics are introduced: (i) basic set operations; (ii) zonotopes; (iii) constrained zonotopes; (iv) complexity reduction methods; (v) interval analysis; and (iv) linear state estimation using constrained zonotopes.

3.1 Basic set operations and notation

This section describes some common set operations which are useful for computing enclosures required by set-based state estimation (Le et al., 2013; Scott et al., 2016). Consider sets $Z, W \subset \mathbb{R}^n$, and $Y \subset \mathbb{R}^m$, and a real matrix $\mathbf{R} \in \mathbb{R}^{m \times n}$. Define the linear mapping, Minkowski sum, and generalized intersection, as

$$\mathbf{R}Z \triangleq \{\mathbf{R}\mathbf{z} : \mathbf{z} \in Z\}, \quad (3.1)$$

$$Z \oplus W \triangleq \{\mathbf{z} + \mathbf{w} : \mathbf{z} \in Z, \mathbf{w} \in W\}, \quad (3.2)$$

$$Z \cap_{\mathbf{R}} Y \triangleq \{\mathbf{z} \in Z : \mathbf{R}\mathbf{z} \in Y\}, \quad (3.3)$$

respectively. Using ellipsoids or parallelotopes, the linear mapping (3.1) can be computed efficiently and accurately, but the Minkowski sum (3.2) and intersection (3.3) must be conservatively outer-approximated (Chisci et al., 1996; Schweppe, 1968). On the other hand, for intervals, the Minkowski sum (3.2) is efficient and exact, but (3.1) and (3.3) are

conservative due to the presence of the wrapping effect¹. This well-known effect in interval analysis is associated to the inability of an interval variable to capture existing dependencies between its components (Moore et al., 2009). In contrast, convex polytopes are closed under (3.1)–(3.3). The operations (3.1) and (3.2) can be computed efficiently in vertex representation (V-rep), while (3.3) can be computed efficiently in half-space representation (H-rep). However, if multiple computations of (3.1)–(3.3) are done using convex polytopes, this requires conversions between H-rep and V-rep which can be extremely expensive and numerically unstable in high dimensions (Hagemann, 2015).

In this doctoral thesis, functions with set-valued arguments will be consistently used to denote exact image of the set under the function; e.g., $\boldsymbol{\mu}(X, W) \triangleq \{\boldsymbol{\mu}(\mathbf{x}, \mathbf{w}) : \mathbf{x} \in X, \mathbf{w} \in W\}$. In addition, the following notations are defined to be used in the proofs. Let $\boldsymbol{\kappa}$ be a function of class \mathcal{C}^2 , and let \mathbf{z} denote its argument. Then, κ_q denotes the q -th component of $\boldsymbol{\kappa}$, $\nabla \boldsymbol{\kappa}$ denotes the gradient of $\boldsymbol{\kappa}$, and $\mathbf{H}\kappa_q$ is an upper triangular matrix describing half of the Hessian of κ_q . Specifically, $H_{ii}\kappa_q = (1/2)\partial^2\kappa_q/\partial z_i^2$, $H_{ij}\kappa_q = \partial^2\kappa_q/\partial z_i\partial z_j$ for $i < j$, and $H_{ij}\kappa_q = 0$ for $i > j$.

3.2 Zonotopes and constrained zonotopes

Zonotopes are centrally symmetric sets (Kühn, 1998) which allow both (3.1) and (3.2) to be computed exactly and with low computational burden. However, the intersection (3.3) does not result in a zonotope in general, can be difficult to compute, and must be outer-approximated (Le et al., 2013; Bravo et al., 2006).

Definition 3.1. A set $Z \subset \mathbb{R}^n$ is a *zonotope* if there exists $(\mathbf{G}_z, \mathbf{c}_z) \in \mathbb{R}^{n \times n_g} \times \mathbb{R}^n$ such that

$$Z = \{\mathbf{c}_z + \mathbf{G}_z \boldsymbol{\xi} : \|\boldsymbol{\xi}\|_\infty \leq 1\}. \quad (3.4)$$

Equation (3.4) is called *generator representation* (G-rep). Each column of \mathbf{G}_z is a *generator*, and \mathbf{c}_z is the *center*. In this doctoral thesis, we refer to $\boldsymbol{\xi}$ as the *generator variables*. By defining the n_g -dimensional unitary hypercube $B_\infty^{n_g} \triangleq \{\boldsymbol{\xi} \in \mathbb{R}^{n_g} : \|\boldsymbol{\xi}\|_\infty \leq 1\}$ (with $B_\infty \triangleq B_\infty^1$), a zonotope Z can be alternatively interpreted as an affine transformation of $B_\infty^{n_g}$, given by $Z = \mathbf{c}_z \oplus \mathbf{G}_z B_\infty^{n_g}$. Since $B_\infty^{n_g}$ is symmetric, then the G-rep (3.4) is limited to centrally symmetric convex polytopes. Another interpretation of a zonotope Z is the Minkowski sum of line segments, given by $Z = \bigoplus_{j=1}^{n_g} \mathbf{g}_{z,j} B_\infty \oplus \mathbf{c}_z$, where $\mathbf{g}_{z,j}$ is the j -th column of \mathbf{G}_z . In this doctoral thesis, we use the shorthand notation $Z = \{\mathbf{G}_z, \mathbf{c}_z\}$ for zonotopes.

Zonotopes have a few computational benefits in comparison to other set representations. Let $Z = \{\mathbf{G}_z, \mathbf{c}_z\} \subset \mathbb{R}^n$, $W = \{\mathbf{G}_w, \mathbf{c}_w\} \subset \mathbb{R}^n$, and $\mathbf{R} \in \mathbb{R}^{m \times n}$. The set operations (3.1)–(3.2)

¹The generalized intersection in (3.3) is not conservative when $\mathbf{R} = \mathbf{I}$, which corresponds to the standard intersection \cap .

are computed efficiently in G-rep as

$$\mathbf{R}Z = \{\mathbf{R}\mathbf{G}_z, \mathbf{R}\mathbf{c}_z\}, \quad (3.5)$$

$$Z \oplus W = \left\{ \begin{bmatrix} \mathbf{G}_z & \mathbf{G}_w \end{bmatrix}, \mathbf{c}_z + \mathbf{c}_w \right\}. \quad (3.6)$$

The operations (3.5)–(3.6) cause only a mild increase in the complexity of the G-rep (3.4). Unlike polytopes, this increased complexity can be addressed efficiently using order reduction algorithms (Section 3.3), to enclose a zonotope with another one with fewer number of generators, allowing one to balance accuracy and computational efficiency.

The term *constrained zonotope* was first used in Ponsini et al. (2012) to refer to a new class of sets introduced in Ghorbal et al. (2010). The latter is an extension of zonotopes defined by the logical product of a zonotope and polyhedral sets. In this doctoral thesis, we are interested in the constrained zonotopes (also an extension of zonotopes) defined as in Scott et al. (2016). These are capable of describing also asymmetric convex polytopes, while maintaining many of the well-known computational benefits of zonotopes.

Definition 3.2. A set $Z \subset \mathbb{R}^n$ is a *constrained zonotope* if there exists $(\mathbf{G}_z, \mathbf{c}_z, \mathbf{A}_z, \mathbf{b}_z) \in \mathbb{R}^{n \times n_g} \times \mathbb{R}^n \times \mathbb{R}^{n_c \times n_g} \times \mathbb{R}^{n_c}$ such that

$$Z = \{\mathbf{c}_z + \mathbf{G}_z \boldsymbol{\xi} : \|\boldsymbol{\xi}\|_\infty \leq 1, \mathbf{A}_z \boldsymbol{\xi} = \mathbf{b}_z\}. \quad (3.7)$$

We refer to (3.7) as the *constrained generator representation* (CG-rep). Each column of \mathbf{G}_z is a *generator*, \mathbf{c}_z is the *center*, $\mathbf{A}_z \boldsymbol{\xi} = \mathbf{b}_z$ are the *constraints*, and $\boldsymbol{\xi}$ are the *generator variables*. By defining the constrained unitary hypercube² $B_\infty(\mathbf{A}_z, \mathbf{b}_z) \triangleq \{\boldsymbol{\xi} \in \mathbb{R}^{n_g} : \|\boldsymbol{\xi}\|_\infty \leq 1, \mathbf{A}_z \boldsymbol{\xi} = \mathbf{b}_z\}$, a constrained zonotope Z can be alternatively interpreted as an affine transformation of $B_\infty(\mathbf{A}_z, \mathbf{b}_z)$, given by $Z = \mathbf{c} \oplus \mathbf{G}_z B_\infty(\mathbf{A}_z, \mathbf{b}_z)$. Differently from zonotopes, the linear equality constraints in (3.7) allow constrained zonotopes to represent any convex polytope provided that the complexity of the CG-rep (3.7) (i.e., the number of generators and constraints) is not limited. In fact, Z is a constrained zonotope iff it is a convex polytope (Property 3.2). We use the compact notation $Z = \{\mathbf{G}_z, \mathbf{c}_z, \mathbf{A}_z, \mathbf{b}_z\}$ for constrained zonotopes.

In addition to (3.1) and (3.2), the intersection (3.3) can also be computed exactly with constrained zonotopes. Let $Z = \{\mathbf{G}_z, \mathbf{c}_z, \mathbf{A}_z, \mathbf{b}_z\} \subset \mathbb{R}^n$, $W = \{\mathbf{G}_w, \mathbf{c}_w, \mathbf{A}_w, \mathbf{b}_w\} \subset \mathbb{R}^n$, $Y = \{\mathbf{G}_y, \mathbf{c}_y, \mathbf{A}_y, \mathbf{b}_y\} \subset \mathbb{R}^m$, and $\mathbf{R} \in \mathbb{R}^{m \times n}$. The set operations (3.1)–(3.3) are computed trivially in CG-rep (see Scott et al. (2016) for a proof) as

$$\mathbf{R}Z = \{\mathbf{R}\mathbf{G}_z, \mathbf{R}\mathbf{c}_z, \mathbf{A}_z, \mathbf{b}_z\}, \quad (3.8)$$

²We drop the use of the superscript n_g for $B_\infty(\mathbf{A}_z, \mathbf{b}_z)$ since this dimension can be inferred from the number of columns of \mathbf{A}_z .

$$Z \oplus W = \left\{ \left[\mathbf{G}_z \ \mathbf{G}_w \right], \mathbf{c}_z + \mathbf{c}_w, \begin{bmatrix} \mathbf{A}_z & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_w \end{bmatrix}, \begin{bmatrix} \mathbf{b}_z \\ \mathbf{b}_w \end{bmatrix} \right\}, \quad (3.9)$$

$$Z \cap_{\mathbb{R}} Y = \left\{ \left[\mathbf{G}_z \ \mathbf{0} \right], \mathbf{c}_z, \begin{bmatrix} \mathbf{A}_z & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_y \\ \mathbf{R}\mathbf{G}_z & -\mathbf{G}_y \end{bmatrix}, \begin{bmatrix} \mathbf{b}_z \\ \mathbf{b}_y \\ \mathbf{c}_y - \mathbf{R}\mathbf{c}_z \end{bmatrix} \right\}. \quad (3.10)$$

These operations can be performed efficiently and cause only a moderate increase in the complexity of the CG-rep. As with zonotopes, efficient methods for complexity reduction of constrained zonotopes, i.e. to enclose a constrained zonotope by another one with a fewer number of generators and constraints, are available (Section 3.3). Other useful operations with constrained zonotopes are presented in the following. Property 3.1 provides simple methods to verify if a constrained zonotope is empty and if a given point belongs to it (Scott et al., 2016). Property 3.2 is a method to compute the corresponding CG-rep of a given convex polytope. Property 3.3 provides a simple method for computing the interval hull of a constrained zonotope by solving $2n$ linear programs (LPs). For simplicity, the subscripts of the variables in (3.7) will be suppressed henceforth when not necessary.

Property 3.1. (Scott et al., 2016) For every $Z = \{\mathbf{G}, \mathbf{c}, \mathbf{A}, \mathbf{b}\} \subset \mathbb{R}^n$ and $\mathbf{z} \in \mathbb{R}^n$,

$$\begin{aligned} Z \neq \emptyset &\iff \min_{\boldsymbol{\xi}} \{ \|\boldsymbol{\xi}\|_{\infty} : \mathbf{A}\boldsymbol{\xi} = \mathbf{b} \} \leq 1, \\ \mathbf{z} \in Z &\iff \min_{\boldsymbol{\xi}} \left\{ \|\boldsymbol{\xi}\|_{\infty} : \begin{bmatrix} \mathbf{G} \\ \mathbf{A} \end{bmatrix} \boldsymbol{\xi} = \begin{bmatrix} \mathbf{z} - \mathbf{c} \\ \mathbf{b} \end{bmatrix} \right\} \leq 1. \end{aligned}$$

Property 3.2. (From Theorem 1 in Scott et al. (2016)) Let $P = \{\mathbf{z} : \mathbf{H}\mathbf{z} \leq \mathbf{k}\} \subset \mathbb{R}^n$ be a convex polytope in H-rep, and choose $Z = \{\mathbf{G}, \mathbf{c}\} \subset \mathbb{R}^n$ and $\boldsymbol{\sigma} \in \mathbb{R}^n$ such that $P \subseteq Z$ and $\mathbf{H}\mathbf{z} \in [\boldsymbol{\sigma}, \mathbf{k}], \forall \mathbf{z} \in P$. Then, the convex polytope P can be written in CG-rep as

$$P = \left\{ \left[\mathbf{G} \ \mathbf{0} \right], \mathbf{c}, \left[\mathbf{H}\mathbf{G} \ \text{diag}\left(\frac{\boldsymbol{\sigma} - \mathbf{k}}{2}\right) \right], \frac{\mathbf{k} + \boldsymbol{\sigma}}{2} - \mathbf{H}\mathbf{c} \right\}. \quad (3.11)$$

Property 3.3. (Scott et al., 2016; Rego et al., 2018b) Let $Z = \{\mathbf{G}, \mathbf{c}, \mathbf{A}, \mathbf{b}\} \subset \mathbb{R}^n$, and let $\mathbf{G}_{j,:}$ denote the j -th row of \mathbf{G} . The *interval hull* $[\zeta^L, \zeta^U] \supseteq Z$ is obtained by solving the following linear programs for each $j = 1, 2, \dots, n$:

$$\begin{aligned} \zeta_j^L &= \min_{\boldsymbol{\xi}} \{ c_j + \mathbf{G}_{j,:}\boldsymbol{\xi} : \|\boldsymbol{\xi}\|_{\infty} \leq 1, \mathbf{A}\boldsymbol{\xi} = \mathbf{b} \}, \\ \zeta_j^U &= \max_{\boldsymbol{\xi}} \{ c_j + \mathbf{G}_{j,:}\boldsymbol{\xi} : \|\boldsymbol{\xi}\|_{\infty} \leq 1, \mathbf{A}\boldsymbol{\xi} = \mathbf{b} \}. \end{aligned}$$

3.3 Complexity reduction and rescaling

This section describes the methods used in this doctoral thesis for complexity reduction of zonotopes and constrained zonotopes. Other useful algorithms required by the techniques

proposed in this thesis are also presented.

Method 3.1 is the pioneering general-purpose method for order reduction of zonotopes. It allows the efficient computation of a zonotope with a desired number of generators enclosing a more complex zonotope. Nevertheless, the resulting enclosure can be very conservative (Yang & Scott, 2018).

Method 3.1. (Generator reduction A) (Combastel, 2003; Alamo et al., 2005) Let $Z = \{\mathbf{G}, \mathbf{c}\} \subset \mathbb{R}^n$ be a zonotope, with $\mathbf{G} \in \mathbb{R}^{n \times n_g}$, $\mathbf{c} \in \mathbb{R}^n$, and consider $s \in \mathbb{N}$ such that $n \leq s < n_g$. Denote by $\hat{\mathbf{G}}$ the matrix resulting from reordering the columns of \mathbf{G} in decreasing 2-norm. Then,

$$Z \subseteq \bar{Z} = \{[\mathbf{H} \ \mathbf{Q}], \mathbf{c}\},$$

where \bar{Z} has s generators, \mathbf{H} is a matrix composed of the first $s - n$ columns of matrix $\hat{\mathbf{G}}$, and \mathbf{Q} is a diagonal matrix given by

$$Q_{ii} = \sum_{j=s-n+1}^{n_g} |\hat{G}_{ij}|,$$

for $i = 1, 2, \dots, n$.

A weighted version of Method 3.1 is proposed in Combastel (2015b). However, the available criteria for choosing the weighting matrix depends on the structure of the state estimation algorithm, and are proposed only for few specific linear cases. A more accurate order reduction method has been proposed in Althoff et al. (2010). Nevertheless, this exhibits a very high computational cost in higher dimensions. On the other hand, Method 3.2 was proposed in Scott et al. (2016), with improved effectiveness and accuracy with respect to other available methods, as highlighted in Yang & Scott (2018).

Method 3.2. (Generator reduction B) (Scott et al., 2016; Yang & Scott, 2018) Consider the zonotope $Z = \{\mathbf{G}, \mathbf{c}\} \subset \mathbb{R}^n$, with $\mathbf{G} \in \mathbb{R}^{n \times n_g}$, $\mathbf{c} \in \mathbb{R}^n$. A less conservative method for generator reduction of zonotopes is given by:

- 1) Reorder the columns of \mathbf{G} as $[\mathbf{T} \ \mathbf{V}]$, where $\mathbf{T} \in \mathbb{R}^{n \times n}$ is an invertible matrix.
- 2) Choose a column \mathbf{v} of $\mathbf{V} \in \mathbb{R}^{n \times (n_g - n)}$ and rewrite Z as $Z = X \oplus Y \triangleq \{[\mathbf{T} \ \mathbf{v}], \mathbf{c}\} \oplus \{\mathbf{V}_-, \mathbf{0}\}$, where $\mathbf{V}_- \in \mathbb{R}^{n \times (n_g - n - 1)}$ is the matrix obtained by removing \mathbf{v} from \mathbf{V} .
- 3) Reduce X to an optimal parallelotope \tilde{X} satisfying $X \subseteq \tilde{X}$.
- 4) Assign $\tilde{Z} = \tilde{X} + Y$, where \tilde{Z} is a zonotope with $n_g - 1$ generators.

Step 1 in Method 3.2 is performed using Gauss-Jordan elimination, in which the sequence of column pivots yield the desired ordering $[\mathbf{T} \ \mathbf{V}]$. Moreover, Step 3 is done using the optimal reduction method proposed in Chisci et al. (1996), according to which the

minimum volume parallelotope containing X is $\tilde{X} \triangleq \{\mathbf{T}(\mathbf{I} + \text{diag}(|\mathbf{r}|)), \mathbf{c}\}$, where $\mathbf{r} = \mathbf{T}^{-1}\mathbf{v}$. It can be shown that $\|\mathbf{r}\|_\infty \leq 1$. Lastly, in Step 2, the column \mathbf{v} is chosen by minimizing the volume error between \tilde{X} and X . This error is denoted by $v(\tilde{X}) - v(X)$, which is computed using volume equation presented in Alamo et al. (2005), resulting in

$$v(\tilde{X}) - v(X) = 2^n |\det(\mathbf{T})| \left(\prod_{i=1}^n (1 + |r_i|) - \left(1 + \sum_{i=1}^n |r_i| \right) \right).$$

Implementation details of Method 3.2 are found in Scott et al. (2016).

Method 3.3 corresponds to the *rescaling* procedure of constrained zonotopes proposed in the work of Scott et al. (2016), which transfers part of the information present in the equality constraints to its generators.

Method 3.3. (Rescaling) (Scott et al., 2016) Consider a constrained zonotope $Z = \{\mathbf{G}, \mathbf{c}, \mathbf{A}, \mathbf{b}\}$, with $\mathbf{G} \in \mathbb{R}^{n \times n_g}$, $\mathbf{c} \in \mathbb{R}^n$, $\mathbf{A} \in \mathbb{R}^{n_c \times n_g}$, and $\mathbf{b} \in \mathbb{R}^{n_c}$. If $\boldsymbol{\xi}^L, \boldsymbol{\xi}^U \in \mathbb{R}^{n_g}$ satisfy $B_\infty(\mathbf{A}, \mathbf{b}) \subset [\boldsymbol{\xi}^L, \boldsymbol{\xi}^U] \subset [-\mathbf{1}, \mathbf{1}]$, then an equivalent CG-rep is

$$Z = \{\mathbf{G}\text{diag}(\boldsymbol{\xi}_r), \mathbf{c} + \mathbf{G}\boldsymbol{\xi}_m, \mathbf{A}\text{diag}(\boldsymbol{\xi}_r), \mathbf{b} - \mathbf{A}\boldsymbol{\xi}_m\}, \quad (3.12)$$

where $\boldsymbol{\xi}_m = \frac{1}{2}(\boldsymbol{\xi}^L + \boldsymbol{\xi}^U)$ and $\boldsymbol{\xi}_r = \frac{1}{2}(\boldsymbol{\xi}^U - \boldsymbol{\xi}^L)$. The *rescaling* procedure corresponds to the process of computing the interval $[\boldsymbol{\xi}^L, \boldsymbol{\xi}^U]$ and replacing $Z = \{\mathbf{G}, \mathbf{c}, \mathbf{A}, \mathbf{b}\}$ by (3.12).

An efficient procedure based on interval arithmetic to obtain a tight interval $[\boldsymbol{\xi}^L, \boldsymbol{\xi}^U]$, satisfying $B_\infty(\mathbf{A}, \mathbf{b}) \subset [\boldsymbol{\xi}^L, \boldsymbol{\xi}^U] \subset [-\mathbf{1}, \mathbf{1}]$ in Method 3.3, is described in Algorithm 3.1.

Algorithm 3.1 Rescaling using interval arithmetic (Scott et al., 2016)

- 1: Assign $E \leftarrow [-\mathbf{1}, \mathbf{1}] \subset \mathbb{R}^{n_g}$, $R \leftarrow (-\infty, +\infty) = \mathbb{R}^{n_g}$, $i = j = 1$.
- 2: If $A_{ij} \neq 0$, assign

$$R_j \leftarrow R_j \cap \left(A_{ij}^{-1} b_i - \sum_{k \neq j} A_{ij}^{-1} A_{ik} E_k \right),$$

$$E_j \leftarrow E_j \cap R_j.$$

- 3: If $j < n_g$, assign $j \leftarrow j + 1$ and go to Step 2. Otherwise, if $j = n_g$ and $i < n_c$, assign $(i, j) \leftarrow (i + 1, 1)$ and go to Step 2. If $(i, j) = (n_c, n_g)$, assign $[\boldsymbol{\xi}^L, \boldsymbol{\xi}^U] \leftarrow E$ and terminate.
-

Method 3.4 was proposed in Scott et al. (2016) to compute a constrained zonotope enclosing another one having one less constraint. This is done by eliminating one constraint-generator pair from the latter with reduced conservativeness. Algorithm 3.2 summarizes the method.

Method 3.4. (Constraint elimination) (Scott et al., 2016) Consider the constrained zonotope $Z = \{\mathbf{G}, \mathbf{c}, \mathbf{A}, \mathbf{b}\}$, with $\mathbf{G} \in \mathbb{R}^{n \times n_g}$, $\mathbf{c} \in \mathbb{R}^n$, $\mathbf{A} \in \mathbb{R}^{n_c \times n_g}$, and $\mathbf{b} \in \mathbb{R}^{n_c}$. Let $\boldsymbol{\Lambda}_G \in \mathbb{R}^{n \times n_c}$ and $\boldsymbol{\Lambda}_A \in \mathbb{R}^{n_c \times n_c}$. Then, the following holds (Proposition 5 in Scott et al.

(2016)):

$$Z \subseteq \tilde{Z} = \{\mathbf{G} - \mathbf{\Lambda}_G \mathbf{A}, \mathbf{c} + \mathbf{\Lambda}_G \mathbf{b}, \mathbf{A} - \mathbf{\Lambda}_A \mathbf{A}, \mathbf{b} - \mathbf{\Lambda}_A \mathbf{b}\}. \quad (3.13)$$

Therefore, one pair of constraint-generator can be eliminated from Z as follows. Let the i -th row of $\mathbf{\Lambda}_A$ be the i -th unit vector, then \tilde{Z} has the trivial i -th constraint $\mathbf{0}^T \boldsymbol{\xi} = \mathbf{0}$, which can be removed. Choose any $i \in \{1, 2, \dots, n_c\}$ and $j \in \{1, 2, \dots, n_g\}$ such that $A_{ij} \neq 0$. From the equality constraints $\mathbf{A}\boldsymbol{\xi} = \mathbf{b}$, it holds that

$$\xi_j = A_{ij}^{-1} b_j - A_{ij}^{-1} \sum_{k \neq j} A_{ik} \xi_k. \quad (3.14)$$

Thus, the elimination of the i -th constraint in Z is performed by computing (3.13) with $\mathbf{\Lambda}_G = \mathbf{G} \mathbf{E}_{ji} A_{ij}^{-1}$ and $\mathbf{\Lambda}_A = \mathbf{A} \mathbf{E}_{ji} A_{ij}^{-1}$, where \mathbf{E}_{ji} is a matrix of zeros with exception of an element one in the j -th row and i -th column. The resulting constrained zonotope has $n_g - 1$ generators and $n_c - 1$ constraints.

To choose the j -th generator to eliminate with the least conservativeness, the constraints $\mathbf{A}\boldsymbol{\xi} = \mathbf{b}$ are first preconditioned using Gauss-Jordan elimination with full pivoting. Then, consider the variables $\boldsymbol{\xi}_m, \boldsymbol{\xi}_r$ from Method 3.3, and let $[\boldsymbol{\rho}^L, \boldsymbol{\rho}^U]$ denote the interval obtained in Algorithm 3.1, rescaled as

$$[\rho_j^L, \rho_j^U] \leftarrow \left[\frac{\rho_j^L - \xi_{m,j}}{\xi_{r,j}}, \frac{\rho_j^U - \xi_{m,j}}{\xi_{r,j}} \right], \quad \forall j \in \{1, 2, \dots, n_g\}.$$

The choice of $j \in \{1, 2, \dots, n_g\}$ is made such that the Hausdorff error between Z and \tilde{Z} in (3.13) is minimized (Scott et al., 2016). Let $\mathbf{r} \in \mathbb{R}^{n_g}$ such that $r_j = \max\{0, \max\{|\rho_j^L|, |\rho_j^U|\} - 1\}$. For computational reasons, for each j , the Hausdorff error is then approximated by

$$\hat{H}_j = \min_{\mathbf{d}} \{\|\mathbf{G}\mathbf{d}\|_2^2 + \|\mathbf{d}\|_2^2 : \mathbf{A}\mathbf{d} = \mathbf{0}, d_j = r_j\}.$$

The optimal \mathbf{d}^* is obtained by solving (Scott et al., 2016)

$$\begin{bmatrix} \mathbf{I}_{n_g+n_c} & \mathbf{Q}^{-1} \mathbf{e}_j \\ \mathbf{e}_j^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{d}^* \\ \boldsymbol{\lambda}^* \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{(n_g+n_c) \times 1} \\ r_j \end{bmatrix}, \quad (3.15)$$

with

$$\mathbf{Q} = \begin{bmatrix} \mathbf{G}^T \mathbf{G} + \mathbf{I}_{n_g} & \mathbf{A}^T \\ \mathbf{A} & \mathbf{0}_{n_c \times n_c} \end{bmatrix},$$

where $\mathbf{e}_j \in \mathbb{R}^{n_g+n_c}$ is the j -th standard unit vector augmented with n_c zeros. Then, \hat{H}_j is computed for each j , from which $j^* = \arg \min_j \hat{H}_j$. Implementation details can be found in Scott et al. (2016).

Method 3.5 has been proposed in Scott et al. (2016) as a required intermediate step to apply generator reduction methods to constrained zonotopes. It is used after constraint

Algorithm 3.2 Constraint elimination (Scott et al., 2016)

- 1: Assign to (\mathbf{A}, \mathbf{b}) the result obtained from Gauss-Jordan elimination applied to $\mathbf{A}\boldsymbol{\xi} = \mathbf{b}$ and reorder the columns of \mathbf{G} accordingly.
- 2: Rescale Z using Method 3.3 and Algorithm 3.1.
- 3: Compute every \hat{H}_j by solving (3.15), assigning $\hat{H}_j = \|\mathbf{G}\mathbf{d}^*\|_2^2 + \|\mathbf{d}^*\|_2^2$.
- 4: Choose which generator to eliminate based on $j^* = \arg \min_j \hat{H}_j$.
- 5: Eliminate one constraint from Z using (3.13), with $\boldsymbol{\Lambda}_G = \mathbf{G}\mathbf{E}_{j^*}A_{ij^*}^{-1}$, and $\boldsymbol{\Lambda}_A = \mathbf{A}\mathbf{E}_{j^*}A_{ij^*}^{-1}$.

elimination (Method 3.4), and allows zonotope-based order reduction methods, such as Methods 3.1 and 3.2, to be extended to constrained zonotopes as well.

Method 3.5. (Lift-then-reduce) (Scott et al., 2016) Consider a constrained zonotope $Z = \{\mathbf{G}, \mathbf{c}, \mathbf{A}, \mathbf{b}\}$, with $\mathbf{G} \in \mathbb{R}^{n \times n_g}$, $\mathbf{c} \in \mathbb{R}^n$, $\mathbf{A} \in \mathbb{R}^{n_c \times n_g}$, and $\mathbf{b} \in \mathbb{R}^{n_c}$. Let Z^+ denote its *lifted zonotope*, defined by

$$Z^+ \triangleq \left\{ \begin{bmatrix} \mathbf{G} \\ \mathbf{A} \end{bmatrix}, \begin{bmatrix} \mathbf{c} \\ -\mathbf{b} \end{bmatrix} \right\} \subset \mathbb{R}^{n+n_c}.$$

Then, it holds that $\mathbf{z} \in Z \iff (\mathbf{z}, \mathbf{0}) \in Z^+$ (Proposition 3 in Scott et al. (2016)). Therefore, generator reduction of constrained zonotopes is done as follows. Let $s \in \mathbb{N}$ satisfy $(n+n_c) \leq s < n_g$, and let \tilde{Z}^+ be the zonotope with s generators obtained by applying a zonotope order reduction method to Z^+ . Then, $Z^+ \subseteq \tilde{Z}^+$, and $Z \subseteq \tilde{Z}$, where \tilde{Z} is the constrained zonotope associated to \tilde{Z}^+ , having $s < n_g$ generators and n_c constraints.

Remark 3.1. Method 3.5 is used as an intermediate step for generator reduction of constrained zonotopes in all the experiments in this thesis. Therefore, explicit references to this method will be omitted.

3.4 Interval analysis

The methods developed in this doctoral thesis require some concepts from interval analysis, which are briefly introduced in this section.

Interval analysis is a traditional tool for set-based state estimation of nonlinear systems (Jaulin et al., 2001b). Let the set of compact intervals in \mathbb{R} be denoted by \mathbb{IR} . An *interval* $X \in \mathbb{IR}$ is a real compact set defined by $X \triangleq \{x \in \mathbb{R} : x^L \leq a \leq x^U\}$, with shorthand notation $X = [x^L, x^U]$, where $x^L, x^U \in \mathbb{R}$ are called *endpoints*. The midpoint and radius are defined by $\text{mid}(X) = \frac{1}{2}(x^U + x^L)$ and $\text{rad}(X) = \frac{1}{2}(x^U - x^L)$. The diameter is defined by $\text{diam}(X) = 2\text{rad}(X)$. The four basic interval arithmetic operations are defined by $[x^L, x^U] \odot [y^L, y^U] = \{x \odot y : x \in [x^L, x^U], y \in [y^L, y^U]\}$, where ‘ \odot ’ denotes interval addition, subtraction, multiplication or division (the latter works provided that $0 \notin [y^L, y^U]$). Moreover, for any bounded set $X \subset \mathbb{R}^n$, let $\square X$ refer to the interval hull of X .

Definition 3.3. Given $(\mathbf{x}^L, \mathbf{x}^U) \in \mathbb{R}^n \times \mathbb{R}^n$ satisfying $x_j^L \leq x_j^U$ for all $j = 1, 2, \dots, n$, an *interval vector* $X \in \mathbb{IR}^n$ is defined by $X \triangleq \{\mathbf{x} \in \mathbb{R}^n : x_j^L \leq x_j \leq x_j^U, j = 1, 2, \dots, n\}$.

An interval vector $X \in \mathbb{IR}^n$ can be written in G-rep as $\text{mid}(X) \oplus \text{diag}(\text{rad}(X))B_\infty^n = \{\text{diag}(\text{rad}(X)), \text{mid}(X)\}$. Similarly to Definition 3.3, interval matrices are defined by $\{\mathbf{A} \in \mathbb{R}^{n \times m} : A_{ij}^L \leq A_{ij} \leq A_{ij}^U, i = 1, 2, \dots, n, j = 1, 2, \dots, m\} \in \mathbb{IR}^{n \times m}$. Interval matrix operations are performed according to the rules of real matrix algebra. The midpoint and radius are defined componentwise for interval vectors and matrices (Moore et al., 2009). Moreover, an interval scalar $X = [x^L, x^U] \in \mathbb{IR}$ is said to be *degenerate* iff $x^L = x^U$.

Definition 3.4. An interval vector $[x^L, x^U] \in \mathbb{IR}^n$ is *degenerate* iff $x_j^L = x_j^U$ for at least one $j \in \{1, 2, \dots, n\}$. Otherwise, it is said to be *non-degenerate*.

Inclusion functions perform a central role in interval analysis. Let $\square(\mathbf{f}(X))$ denote an interval enclosure of a real valued function \mathbf{f} over an interval $X \subset \mathbb{R}^n$, where X is an interval enclosure of the domain of interest³. Then, $\mathbf{f}(X) \subseteq \square(\mathbf{f}(X))$. Inclusion functions of elementary operations, such as $\{\sin, \cos, \ln, \exp\}$, are defined by their images over the respective interval arguments. For instance, the natural inclusion function of a real function is obtained by recursively computing inclusion functions for all arithmetic and elementary operations that compose it (Moore et al., 2009).

Remark 3.2. Well-known sources of conservatism in inclusion functions are the *interval dependency* and the *wrapping effect*. The latter results from the inability of an interval vector (or matrix) to capture any existing dependency between its components. On the other side, the interval dependency results from the multiple occurrences of an interval variable in the same algebraic expression (Moore et al., 2009).

3.4.1 Partitioning

A usual method to reduce the conservatism of inclusion functions is the use of refinements (Moore et al., 2009). In this doctoral thesis, we recall the definition of *interval bundles*, and define *n_d-partitions*.

Definition 3.5. Given n_b interval vectors $B_{(j)} \subset \mathbb{R}^n$, with $j = 1, 2, \dots, n_b$, an *interval bundle* is defined by $\mathcal{B} \triangleq \bigcup_{j=1}^{n_b} B_{(j)}$.

Clearly an interval bundle is not a interval vector in the general case. However, it can be regarded as a list of interval vectors without computing the union explicitly. Moreover, the concept of an interval bundle is closely related to subpavings.

³The notation $\square(\mathbf{f}(X))$ is used through the doctoral thesis even when X is not an interval. In this case, it is assumed that an interval hull of X is employed in the operation.

Remark 3.3. A subpaving is defined as the union of non-overlapping intervals, and it is often associated with a tree structure. Subpavings are usually employed in interval branch-and-bound procedures and set inversion methods (Jaulin et al., 2001b).

Definition 3.6. Let $B \in \mathbb{IR}^n$ be a non-degenerate interval vector. The n_d -partition of B is an interval bundle $\mathcal{B} = \cup_{j=1}^{n_b} B_{(j)}$ such that $\mathcal{B} = B$, $\text{diam}(B_{(j)}) = (1/n_d)\text{diam}(B)$, and the intersection of two intervals $B_{(j)}$ and $B_{(i)}$ produces a degenerate interval vector in \mathbb{R}^n for every $i \neq j$, with $i, j = 1, 2, \dots, n_b$, and $n_d \in \mathbb{N}^+$.

The n_d -partition of an interval vector can be interpreted as a “multi-dimensional grid” with n_d divisions per dimension. By definition, the diameter of each interval vector composing the partition is inversely proportional to the number of divisions, and proportional to the diameter of the original interval vector. Note that $n_b = n_d$ for the unidimensional case.

For a real valued function $f(\cdot)$ and a n_d -partition $\mathcal{B} = \cup_{j=1}^{n_b} B_{(j)}$ of an interval vector B , a *refinement* of an inclusion function $\square(f(\cdot))$ over \mathcal{B} is defined by $\cup_{j=1}^{n_b} \square(f(B_{(j)}))$. The conservatism of a refinement is substantially reduced if compared with single inclusion functions over B (see Chapter 6 in Moore et al. (2009) for details), and since each $\square(f(B_{(j)}))$ results in an interval vector, a refinement is also an interval bundle.

3.5 Linear state estimation

To better introduce and contextualize the advantages of using constrained zonotopes, this section describes the existing method based on a prediction-update structure, proposed in Scott et al. (2016), for set-based state estimation of linear discrete-time systems.

Consider a linear discrete-time system with time k , state $\mathbf{x}_k \in \mathbb{R}^n$, input $\mathbf{u}_k \in \mathbb{R}^{n_u}$, process uncertainty $\mathbf{w}_k \in \mathbb{R}^{n_w}$, measured output $\mathbf{y}_k \in \mathbb{R}^{n_y}$, and measurement uncertainty $\mathbf{v}_k \in \mathbb{R}^{n_v}$. In each time interval $[k-1, k]$, $k = 1, 2, \dots$, the system evolves according to the model

$$\begin{aligned}\mathbf{x}_k &= \mathbf{A}\mathbf{x}_{k-1} + \mathbf{B}\mathbf{u}_{k-1} + \mathbf{B}_w\mathbf{w}_{k-1}, \\ \mathbf{y}_k &= \mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k + \mathbf{D}_v\mathbf{v}_k,\end{aligned}\tag{3.16}$$

with $\mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{B} \in \mathbb{R}^{n \times n_u}$, $\mathbf{B}_w \in \mathbb{R}^{n \times n_w}$, $\mathbf{C} \in \mathbb{R}^{n_y \times n}$, $\mathbf{D} \in \mathbb{R}^{n_y \times n_u}$, and $\mathbf{D}_v \in \mathbb{R}^{n_y \times n_v}$. The disturbances and uncertainties are assumed to be bounded, i.e., $\mathbf{w}_k \in W$ and $\mathbf{v}_k \in V$, where W and V are known convex polytopical sets.

Given an initial condition $\mathbf{x}_0 \in \hat{X}_0$, the prediction-update algorithm consists in computing sets \bar{X}_k and \hat{X}_k such that

$$\bar{X}_k \supseteq \{\mathbf{A}\mathbf{x}_{k-1} + \mathbf{B}\mathbf{u}_{k-1} + \mathbf{B}_w\mathbf{w}_{k-1} : \mathbf{x}_{k-1} \in \hat{X}_{k-1}, \mathbf{w}_{k-1} \in W\},\tag{3.17}$$

$$\hat{X}_k \supseteq \{\mathbf{x}_k \in \bar{X}_k : \mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k + \mathbf{D}_v\mathbf{v}_k = \mathbf{y}_k, \mathbf{v}_k \in V\},\tag{3.18}$$

in which (3.17) is referred to as the *prediction step*, and (3.18) as the *update step*. In the method proposed in Scott et al. (2016), both the prediction and update steps are computed using constrained zonotopes, according to

$$\bar{X}_k = \mathbf{A}\hat{X}_{k-1} \oplus \mathbf{B}_u \mathbf{u}_{k-1} \oplus \mathbf{B}_w W_{k-1}, \quad (3.19)$$

$$\hat{X}_k = \bar{X}_k \cap_{\mathbf{C}} ((\mathbf{y}_k - \mathbf{D}_u \mathbf{u}_k) \oplus (-\mathbf{D}_v V)). \quad (3.20)$$

Unlike all the other sets mentioned in this chapter, the set operations (3.1)–(3.3) present in (3.19) and (3.20) are all computed trivially (they are identities) and exactly using constrained zonotopes by (3.8)–(3.10), and hence the equalities hold. Therefore, the enclosures in (3.17) and (3.18) are obtained accurately and efficiently. In addition, due to the linear increase in the number of generators and constraints, the only source of conservatism comes from the necessity of using complexity reduction methods (Section 3.3). Nevertheless, the results obtained using constrained zonotopes are compared in Scott et al. (2016) with methods based on polytopes (Shamma & Tu, 1999), parallellotopes (Chisci et al., 1996), and zonotopes (Bravo et al., 2006), in which constrained zonotopes provided more accurate enclosures for (3.17)–(3.18) while having a comparable computational burden.

3.6 Final remarks

This chapter introduced most of the required concepts, notation, and definitions used in this doctoral thesis. The following topics were presented: (i) basic set operations; (ii) zonotopes and their properties; (iii) constrained zonotopes and their properties; (iv) methods for reducing the complexity of zonotopes and constrained zonotopes, (v) interval analysis; and (vi) linear state estimation using constrained zonotopes.

The next chapter develops state estimation methods based on constrained zonotopes for discrete-time systems with nonlinear dynamics for the first time, considering linear measurement: (i) the constrained zonotope and interval bundle approach; and the generalization of two existing nonlinear zonotopic methods, namely the (ii) mean value extension and the (iii) first-order Taylor extension using constrained zonotopes.

4

State estimation of systems with nonlinear
dynamics

This chapter introduces new nonlinear state estimation methods based on constrained zonotopes. Three novel nonlinear set-based state estimation methods are developed considering linear measurement, namely the constrained zonotope and interval bundle approach, and the consistent generalization of two existing zonotopic methods for nonlinear systems. The latter are referred to as the mean value extension and the first-order Taylor extension using constrained zonotopes.

4.1 Problem formulation

Consider a class of discrete-time systems with nonlinear dynamics and linear measurements, described by

$$\begin{aligned}\mathbf{x}_k &= \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}), \\ \mathbf{y}_k &= \mathbf{C}\mathbf{x}_k + \mathbf{D}_u\mathbf{u}_k + \mathbf{D}_v\mathbf{v}_k,\end{aligned}\tag{4.1}$$

for $k \geq 1$, with $\mathbf{y}_0 = \mathbf{C}\mathbf{x}_0 + \mathbf{D}_u\mathbf{u}_0 + \mathbf{D}_v\mathbf{v}_0$, where $\mathbf{x}_k \in \mathbb{R}^n$ denotes the system state, $\mathbf{u}_k \in \mathbb{R}^{n_u}$ is a known input, $\mathbf{w}_k \in \mathbb{R}^{n_w}$ is the process disturbance, $\mathbf{y}_k \in \mathbb{R}^{n_y}$ is the measured output, and $\mathbf{v}_k \in \mathbb{R}^{n_v}$ is the measurement uncertainty, with \mathbf{x}_0 the initial state. The nonlinear mapping \mathbf{f} is assumed to be of class \mathcal{C}^2 , and the disturbances and uncertainties are assumed to be bounded, i.e., $\mathbf{w}_k \in W$ and $\mathbf{v}_k \in V$, where W and V are known convex polytopic sets.

This chapter proposes new methods to perform set-valued state estimation for nonlinear

systems as in (4.1). The exact characterization of sets X_k containing the evolution of the system states is very difficult in the nonlinear case, if not intractable (Kieffer et al., 1998; Kühn, 1998; Platzer & Clarke, 2007). Therefore, in the set-membership framework the objective is to enclose such sets as tightly as possible by guaranteed outer bounds \hat{X}_k on the possible trajectories of the system states \mathbf{x}_k . Such outer bounds must be consistent with the previous estimate \hat{X}_{k-1} , known inputs \mathbf{u}_{k-1} , the current measurement \mathbf{y}_k , and also with the bounds on the disturbances and uncertainties W, V . Given an initial condition $\mathbf{x}_0 \in \hat{X}_0$, a common approach is to proceed through the well-known prediction-update algorithm (a generalization of the linear case presented in Section 3.5), which consists in computing compact sets \bar{X}_k and \hat{X}_k such that

$$\bar{X}_k \supseteq \{\mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}) : \mathbf{x}_{k-1} \in \hat{X}_{k-1}, \mathbf{w}_{k-1} \in W\}, \quad (4.2)$$

$$\hat{X}_k \supseteq \{\mathbf{x}_k \in \bar{X}_k : \mathbf{C}\mathbf{x}_k + \mathbf{D}_u\mathbf{u}_k + \mathbf{D}_v\mathbf{v}_k = \mathbf{y}_k, \mathbf{v}_k \in V\}, \quad (4.3)$$

in which (4.2) is referred to as the *prediction step*, and (4.3) as the *update step*.

The goal is to obtain accurate outer bounds \bar{X}_k and \hat{X}_k according to (4.2) and (4.3), respectively. Following the definitions of these enclosures as in (4.2)–(4.3), and considering the initial condition $\mathbf{x}_0 \in \hat{X}_0$, the property $\mathbf{x}_k \in \hat{X}_k$ is guaranteed by construction for all $k \geq 1$ (Chisci et al., 1996; Le et al., 2013; Alamo et al., 2008).

4.2 Prediction step

This section presents three new different approaches for computing the prediction step (4.2) based on constrained zonotopes. The first method (Section 4.2.1) combines properties of interval arithmetic and constrained zonotopes, and yields a highly tunable and accurate state estimation algorithm. The other methods (Sections 4.2.2 and 4.2.3) expand the tools proposed in Scott et al. (2016) to the class of nonlinear discrete-time systems (4.1). These methods extend, in a consistent way, two existing approaches for propagating zonotopes through nonlinear mappings. The key advantage of these new extensions is that they allow the entire state estimation procedure to be done using constrained zonotopes in CG-rep. Therefore, the update step can be done by exact intersection (with linear measurements), which is known to generate highly asymmetrical sets that cannot be accurately enclosed by ellipsoids, intervals, parallelotopes, and zonotopes. Using the methods developed in this section, such sets can be directly propagated to the next time step without prior simplification to a symmetric set. This overcomes a major source of conservatism in existing methods based on the aforementioned enclosures, while largely retaining the efficiency of computations with zonotopes. The content regarding the first method (Section 4.2.1) was published in Rego et al. (2018b). The content concerning the two new extensions (Sections 4.2.2 and 4.2.3) was published in Rego et al. (2020a).

4.2.1 CZIB approach

This section presents the first proposed approach for set-based state estimation of the class of nonlinear systems described in Section 4.1. The method is referred to as the constrained zonotope and interval bundle (CZIB) approach, and combines important properties from interval arithmetic and constrained zonotopes. A highly tunable and accurate nonlinear state estimation algorithm is developed, capable of providing tight enclosures for the set-based state estimation problem described by (4.2)–(4.3).

4.2.1.1 Outline and assumptions

We first present the underlying ideas of the proposed CZIB approach. An additional assumption is made with respect to the set-based state estimation problem described in Section 4.1.

Assumption 4.1. *The nonlinear function \mathbf{f} is affine in \mathbf{w} , i.e., $\mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{w}) \triangleq \bar{\mathbf{f}}(\mathbf{x}, \mathbf{u}) + \mathbf{D}_w \mathbf{w}$, where $\bar{\mathbf{f}} : \mathbb{R}^n \times \mathbb{R}^{n_u} \rightarrow \mathbb{R}^n$, and $\mathbf{D}_w \in \mathbb{R}^{n \times n_w}$.*

Assumption 4.1 is clearly restrictive and leads to a particular reformulation of the prediction step (4.2), given by

$$\bar{X}_k \supseteq \{\bar{\mathbf{f}}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}) + \mathbf{D}_w \mathbf{w}_{k-1} : \mathbf{x}_{k-1} \in \hat{X}_{k-1}, \mathbf{w}_{k-1} \in W\}. \quad (4.4)$$

Nevertheless, note that even if \hat{X}_0 and W are constrained zonotopes, the reachable sets of the nonlinear system (4.1) computed over the domains \hat{X}_{k-1} and W are not convex in general (Althoff, 2013). Thus, the main challenge is to compute the prediction step (4.4) such that the obtained enclosure \bar{X}_k is a constrained zonotope, which is the main contribution of the CZIB approach as the first proposed solution to this problem.

To compute an enclosure \bar{X}_k in CG-rep satisfying (4.4), therefore bounding the possible trajectories of the nonlinear system (4.1) using constrained zonotopes, we subdivide the prediction step into three main intermediate steps:

IS1) *Compute an interval bundle $\mathcal{B} = \cup_{j=1}^{n_b} B_{(j)}$ satisfying $\hat{X}_{k-1} \subseteq \mathcal{B}$;*

IS2) *Compute a refinement $\cup_{j=1}^{n_b} \square(\bar{\mathbf{f}}(B_{(j)}, \mathbf{u}_{k-1}))$, where $\square(\bar{\mathbf{f}}(\cdot))$ is an inclusion function of $\bar{\mathbf{f}}(\cdot)$;*

IS3) *Compute a constrained zonotope \bar{X}_k satisfying $\cup_{j=1}^{n_b} \square(\bar{\mathbf{f}}(B_{(j)})) \oplus \mathbf{D}_w W \subseteq \bar{X}_k$.*

The intermediate step IS2 can be performed straightforwardly through interval arithmetic. Therefore, in this section we focus on developing methods to implement the intermediate steps IS1 and IS3. These methods are presented in the next subsection, along with the complete CZIB approach. In addition, a method for computing a constrained zonotope satisfying the update step (4.3) is discussed in Section 4.3.

4.2.1.2 Intermediate steps

In order to obtain a tight enclosure satisfying *IS1* described by an interval bundle, we require the ability to compute the interval hull of a constrained zonotope (Proposition 3.3). Our goal is to obtain a *tight interval bundle* containing \hat{X}_{k-1} . The following proposition shows how to generate the tight interval bundle containing \hat{X}_{k-1} from a given n_d -partition of the interval hull of \hat{X}_{k-1} (recall Definition 3.5).

Definition 4.1. Let $Z = \{\mathbf{G}, \mathbf{c}, \mathbf{A}, \mathbf{b}\} \subset \mathbb{R}^n$. For a given $n_d \in \mathbb{N}^+$, the *tight interval bundle* $\mathcal{B} = \cup_{j=1}^{n_b} B_{(j)}$ with respect to Z is the union of all intervals $B_{(j)}$ composing the n_d -partition of the interval hull of Z , such that

$$Z \cap_{\mathbf{I}} B_{(j)} \neq \emptyset, \quad (4.5)$$

where \mathbf{I} denotes the identity matrix.

Proposition 4.1. Let $Z = \{\mathbf{G}, \mathbf{c}, \mathbf{A}, \mathbf{b}\} \subset \mathbb{R}^n$. Moreover, let $\mathcal{B} = \cup_{j=1}^{n_b} B_{(j)}$ be the tight interval bundle with respect to Z for a given $n_d \in \mathbb{N}^+$, and let $\tilde{\mathcal{B}} = \cup_{j=1}^{\tilde{n}_b} \tilde{B}_{(j)}$ and $\bar{\mathcal{B}} = \cup_{j=1}^{\bar{n}_b} \bar{B}_{(j)} \supset Z$ be interval bundles originated from the n_d -partition of the interval hull of Z satisfying $\tilde{n}_b < n_b < \bar{n}_b$. Then, $\mathcal{B} \subset \bar{\mathcal{B}}$, and $Z \not\subset \tilde{\mathcal{B}}$.

Proof. Since $\bar{\mathcal{B}}$ and \mathcal{B} are associated to the same n_d -partition, and both $Z \subset \bar{\mathcal{B}}$ and $Z \subset \mathcal{B}$ hold, then at least n_b intervals in $\bar{\mathcal{B}}$ must be equivalent to the intervals in \mathcal{B} . Therefore, $\cup_{j=1}^{\bar{n}_b} \bar{B}_{(j)} = (\cup_{j=1}^{n_b} B_{(j)}) \cup (\cup_{j=n_b+1}^{\bar{n}_b} \bar{B}_{(j)})$. From the properties of the set union operation, we have that $(\cup_{j=1}^{n_b} B_{(j)}) \subset (\cup_{j=1}^{n_b} B_{(j)}) \cup (\cup_{j=n_b+1}^{\bar{n}_b} \bar{B}_{(j)})$. Consequently, $\mathcal{B} \subset \bar{\mathcal{B}}$. On the other hand, by definition of \mathcal{B} , for every B_j in \mathcal{B} , $Z \cap_{\mathbf{I}} B_{(j)} \neq \emptyset$ must hold. Thus, $\mathbf{z} \in Z \implies \mathbf{z} \in B_{(j)}$ for at least one j . Let $\tilde{\mathcal{B}}$ be obtained by removing any $B_{(j)}$ from \mathcal{B} . Then, $\exists \mathbf{z} \in Z : \mathbf{z} \notin \tilde{B}_{(j)}$ for all $\tilde{B}_{(j)}$ in $\tilde{\mathcal{B}}$. Consequently, $\tilde{\mathcal{B}} \not\supset Z$. ■

Remark 4.1. Since every interval vector is a constrained zonotope, the relation (4.5) can be easily verified using (3.10) and Property 3.1 for each $B_{(j)}$, from which the tight interval bundle with respect to \hat{X}_{k-1} is effectively obtained. Further reductions in conservativeness can be achieved by increasing the number of divisions n_d when computing the tight interval bundle.

Figure 4.1 illustrates the underlying ideas regarding Proposition 4.1. We are then able to state Algorithm 4.1, which performs the intermediate step *IS1*.

Algorithm 4.1 Intermediate step *IS1*

- 1: Compute the interval hull of \hat{X}_{k-1} using Property 3.3
 - 2: Compute the n_d -partition of the interval hull according to Definition 3.5
 - 3: Compute the tight interval bundle $\mathcal{B} = \cup_{j=1}^{n_b} B_{(j)}$ containing \hat{X}_{k-1} through Definition 4.1
-

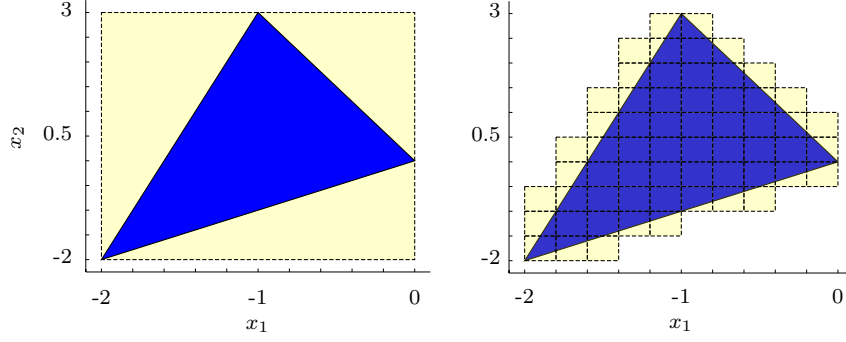


Figure 4.1: Solid lines (blue) depicts a constrained zonotope $Z = \{\mathbf{G}, \mathbf{c}, \mathbf{A}, \mathbf{b}\}$ defined by $\mathbf{G} = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 2 & -1 \end{bmatrix}$, $\mathbf{c} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$, $\mathbf{A} = [-2 \ 1 \ -1]$, and $\mathbf{b} = 2$. Dashed lines denote the interval hull of Z (left), and the tight interval bundle containing Z considering a 10-partition (right).

Intermediate step *IS2* can be done straightforwardly using interval arithmetic. The associated procedure is presented in Algorithm 4.2, which essentially generates a refinement of $\bar{\mathbf{f}}(\mathbf{x}, \mathbf{u}_{k-1})$ over $\mathcal{B} = \cup_{j=1}^{n_b} B_{(j)}$. This procedure can be performed using different inclusion functions¹, which may be tested offline for choosing which leads to less conservatism, according to the nonlinear function $\bar{\mathbf{f}}$.

Algorithm 4.2 Intermediate step *IS2*

- 1: **for** $j = 1, \dots, n_b$ **do**
 - 2: Compute an inclusion function of $\bar{\mathbf{f}}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1})$ for $\mathbf{x}_{k-1} \in B_{(j)}$, from $\mathcal{B} = \cup_{j=1}^{n_b} B_{(j)}$ obtained through Algorithm 4.1
 - 3: **end for**
-

Remark 4.2. As in the case of algorithms involving zonotope bundles (Althoff & Krogh, 2011), the computational burden of Algorithm 4.2 can be reduced using code parallelization techniques.

In order to propose an algorithm for the intermediate step *IS3*, we first describe a method to compute the interval hull of an interval bundle. Then, we define the concept of *endpoint intervals*, which, in practice, are the intervals that intersect the boundaries of the interval hull of the bundle.

Proposition 4.2. Let $\mathcal{B} = \cup_{j=1}^{n_b} B_{(j)}$ be an interval bundle in \mathbb{R}^n , with $B^{(j)} \triangleq [\beta_{(j)}^L, \beta_{(j)}^U]$. The *interval hull* of \mathcal{B} , denoted by $[\zeta^L, \zeta^U]$, is obtained component-wise by searching for $\zeta_i^L = \min\{\beta_{(j),i}^L\}$, $\zeta_i^U = \max\{\beta_{(j),i}^U\}$ for each $i = 1, 2, \dots, n$, where $\beta_{(j),i}^L$ and $\beta_{(j),i}^U$ denote the i -th component of $\beta_{(j)}^L$ and $\beta_{(j)}^U$, respectively.

Proof. The proof follows from Definition 3.3. ■

¹E.g., natural inclusion function, mean value extension, and others (see Moore et al. (2009) for definitions).

Definition 4.2. Let $\mathcal{B} = \cup_{j=1}^{n_b} B_{(j)}$ be an interval bundle in \mathbb{R}^n . The j -th interval vector $B_{(j)}$ is an *endpoint interval* of \mathcal{B} iff at least one component of any of the endpoints of $B_{(j)}$ coincides with at least one component of any endpoint of the interval hull of \mathcal{B} .

To obtain an accurate enclosure, the convex hull of $\cup_{j=1}^{n_b} \square(\bar{\mathbf{f}}(B_{(j)}, \mathbf{u}_{k-1}))$ must be obtained to solve the intermediate step *IS3* with reduced conservatism. However, due to the complexity of this operation, our solution consists in subdividing *IS3* into three other steps:

IS3a) Build a convex polytope using the midpoints of the endpoint intervals of the bundle

$$\cup_{j=1}^{n_b} \square(\bar{\mathbf{f}}(B_{(j)}, \mathbf{u}_{k-1}))$$

as vertices. Denote it by X_k^α .

IS3b) Move the hyperplanes of X_k^α to obtain a polytope X_k^β effectively enclosing the bundle

$$\cup_{j=1}^{n_b} \square(\bar{\mathbf{f}}(B_{(j)}, \mathbf{u}_{k-1})),$$

and convert the resulting set to CG-rep.

Algorithm 4.3 clarifies the steps necessary to perform *IS3a*.

Algorithm 4.3 Step *IS3a*

- 1: Compute the endpoint intervals of $\cup_{j=1}^{n_b} \square(\bar{\mathbf{f}}(B_{(j)}, \mathbf{u}_{k-1}))$ according to Definition 4.2
 - 2: Build a convex polytope in V-rep using the midpoints of the endpoint intervals
 - 3: Convert the resulting set from V-rep to H-rep
-

Figure 4.2 demonstrates the essential stages to obtain the auxiliary convex polytope. Algorithm 4.3 requires the conversion from V-rep to H-rep, which can be expensive. However, in the proposed state estimation algorithm such conversion is performed only once per iteration. In addition, *IS3b* is solved by means of Algorithm 4.4, which requires the computation of n_h LPs at each time step.

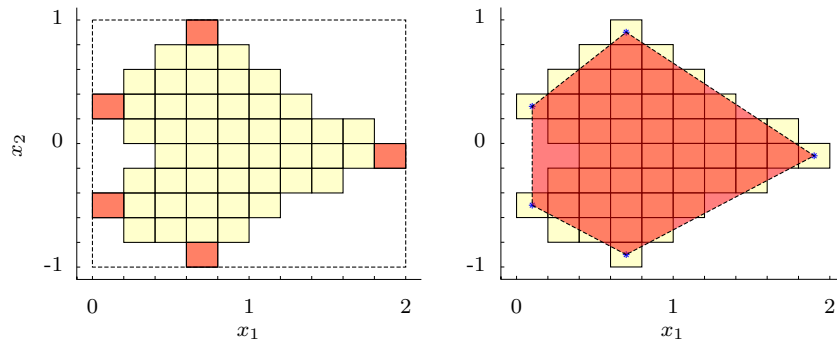


Figure 4.2: Solid lines depicts an interval bundle \mathcal{B} . Left: the interval hull (dashed) and the endpoint intervals (red) of \mathcal{B} . Right: the auxiliary convex polytope, formed by the midpoints of the endpoint intervals (red, dashed).

Remark 4.3. In some cases, the refinement $\cup_{j=1}^{n_b} \square(\bar{\mathbf{f}}(B_{(j)}, \mathbf{u}_{k-1}))$ may not have enough endpoint intervals to form a simplex in \mathbb{R}^{n_x} using the corresponding midpoints. In such situation, to generate additional midpoints before performing the conversion to H-rep,

a “small” zero-centered interval vector can be added to the existing midpoints in V-rep through Minkowski sum.

Algorithm 4.4 Step *IS3b*

- Given X_k^α in the H-rep $\{\mathbf{x} \in \mathbb{R}^n : \mathbf{H}\mathbf{x} \leq \mathbf{k}\}$, $\mathbf{k} \in \mathbb{R}^{n_h}$
- 1: **for** $i = 1, 2, \dots, n_h$ **do**
 - 2: $k'_i = \min_{k_i} \{k_i : \mathbf{H}_i \mathbf{x}^{\{j,l\}} \leq k_i, \forall j, l\}$,
 where $\mathbf{x}^{\{j,l\}}$ denotes the l -th vertex of the j -th interval vector in $\cup_{j=1}^{n_b} \square(\bar{\mathbf{f}}(B_{(j)}, \mathbf{u}_{k-1}))$
 - 3: **end for**
 - 4: Convert the resulting $\{\mathbf{H}, \mathbf{k}'\}$ to CG-rep using (3.11)
-

We now conclude our solution to *IS3*. Given the constrained zonotope

$$X_k^\beta \supset \cup_{j=1}^{n_b} \square(\bar{\mathbf{f}}(B_{(j)}, \mathbf{u}_{k-1})),$$

obtained using Algorithms 4.3 and 4.4, a constrained zonotope \bar{X}_k bounding the trajectories of the system can be obtained through

$$\bar{X}_k = (X_k^\beta \cap H) \oplus \mathbf{D}_w W, \quad (4.6)$$

where H denotes the interval hull of $\cup_{j=1}^{n_b} \square(\bar{\mathbf{f}}(B_{(j)}, \mathbf{u}_{k-1}))$. The intersection with the interval hull H is introduced to obtain tighter bounds, since the vertices of the constrained zonotope X_k^β may extrapolate H . This intersection is computed using (3.10). The linear image and the Minkowski sum in (4.6) are computed exactly using (3.8) and (3.9), respectively.

By solving the intermediate steps *IS1*, *IS2* and *IS3*, the constrained zonotope \bar{X}_k satisfying the prediction step (4.4) is finally obtained, and is given by (4.6). The complete CZIB approach for the prediction step is summarized in Algorithm 4.5.

Algorithm 4.5 CZIB approach for the prediction step (4.2)

Prediction step

- 1: Compute $\mathcal{B} = \cup_{j=1}^{n_b} B_{(j)} \supset \hat{X}_{k-1}$ using Algorithm 4.1
 - 2: Compute $\cup_{j=1}^{n_b} \square(\bar{\mathbf{f}}(B_{(j)}, \mathbf{u}_{k-1}))$ using Algorithm 4.2
 - 3: Compute \bar{X}_k through Algorithms 4.3, 4.4 and (4.6)
-

The compromise between efficiency and accuracy in the CZIB approach is given by the number of divisions chosen in Algorithm 4.1, and the inclusion function chosen to be used in Algorithm 4.2. Moreover, due to the proposed transformations performed in the prediction step (from CG-rep to interval bundle and vice-versa), the increase in the complexity of the constrained zonotope \hat{X}_k is not associated with the time k , but rather with the number of existing endpoint intervals (see Figure 4.2). Therefore, the complexity of the enclosure \bar{X}_k can be limited by choosing a desired number of endpoint intervals

among the existing ones according to a chosen criterion (e.g, minimum allowed distance between vertices).

Remark 4.4. It is possible to perform both the prediction step (4.4) and the update step discussed in Section 4.3, using only interval bundles, i.e., without performing transformations to CG-rep and vice-versa. However, in this case the polytopic sets W and V must be outer-approximated by interval bundles, resulting in undesired conservatism. Besides, the linear mapping in (4.4) would be subject to the wrapping effect due to the use of pure interval arithmetic. Moreover, in order to perform the update step using interval bundles, a set of states consistent with the measurement \mathbf{y}_k must be obtained explicitly (Alamo et al., 2005). If the latter is also an interval bundle, the intersection of the two bundles requires the computation of intersections of every possible pair-wise combination of intervals contained in the different bundles, therefore leading to a worst-case factorial increase in the number of intervals composing the resulting set.

4.2.2 Mean value extension

This section presents the second new method for enclosing the range of a nonlinear function $\boldsymbol{\mu}$ over a set of inputs described by constrained zonotopes. This method is referred to as the *mean value extension* of $\boldsymbol{\mu}$ (because it relies on the Mean Value Theorem), and is a consistent generalization of the zonotope-based method proposed in Alamo et al. (2005). Due to significant differences with respect to its zonotopic counterpart, a new theorem (Theorem 4.2) together with a detailed proof is provided for the new method.

The method in Alamo et al. (2005) relies on a *zonotope inclusion* operator that computes a zonotopic enclosure of the product of an interval matrix with a unitary box. We first generalize this operator to the product of an interval matrix with a constrained zonotope.

Theorem 4.1. Let $Z = \{\mathbf{G}, \mathbf{c}, \mathbf{A}, \mathbf{b}\} \subset \mathbb{R}^{n_z}$ be a constrained zonotope with n_g generators and n_c constraints, let $\mathbf{J} \in \mathbb{I}\mathbb{R}^{n_s \times n_z}$ be an interval matrix, and consider the set $S = \mathbf{J}Z \triangleq \{\hat{\mathbf{J}}\mathbf{z} : \hat{\mathbf{J}} \in \mathbf{J}, \mathbf{z} \in Z\} \subset \mathbb{R}^{n_s}$. Let $\bar{Z} = \{\bar{\mathbf{G}}, \bar{\mathbf{c}}\} \subset \mathbb{R}^{n_z}$ be a zonotope with \bar{n}_g generators satisfying $Z \subseteq \bar{Z}$, let $\mathbf{m} \in \mathbb{I}\mathbb{R}^{n_s}$ be an interval vector such that $\mathbf{m} \supseteq (\mathbf{J} - \text{mid}(\mathbf{J}))\bar{\mathbf{c}}$ and $\text{mid}(\mathbf{m}) = \mathbf{0}$, and let $\mathbf{P} \in \mathbb{R}^{n \times n}$ be a diagonal matrix defined by

$$P_{ii} = \frac{1}{2} \text{diam}(m_i) + \frac{1}{2} \sum_{j=1}^{\bar{n}_g} \sum_{k=1}^{n_z} \text{diam}(J_{ik}) |\bar{G}_{kj}|, \quad (4.7)$$

for all $i = 1, 2, \dots, n_s$. Then, S is contained in the *CZ-inclusion* operator

$$S \subseteq \triangleleft(\mathbf{J}, Z) \triangleq \text{mid}(\mathbf{J})Z \oplus \mathbf{P}B_{\infty}^{n_s}. \quad (4.8)$$

Proof. Choose any $\mathbf{s} \in S$. It will be shown that $\mathbf{s} \in \triangleleft(\mathbf{J}, Z)$. By the definition of S , there

must exist $\mathbf{z} \in Z$ and $\hat{\mathbf{J}} \in \mathbf{J}$ such that $\mathbf{s} = \hat{\mathbf{J}}\mathbf{z}$. Adding and subtracting $\text{mid}(\mathbf{J})\mathbf{z}$,

$$\mathbf{s} = \text{mid}(\mathbf{J})\mathbf{z} + (\hat{\mathbf{J}} - \text{mid}(\mathbf{J}))\mathbf{z}.$$

Since $\mathbf{z} \in Z \subseteq \bar{Z}$, there exists $\boldsymbol{\delta} \in B_{\infty}^{\bar{n}_g}$ such that $\mathbf{z} = \bar{\mathbf{c}} + \bar{\mathbf{G}}\boldsymbol{\delta}$. Therefore, $\mathbf{s} = \text{mid}(\mathbf{J})\mathbf{z} + (\hat{\mathbf{J}} - \text{mid}(\mathbf{J}))(\bar{\mathbf{c}} + \bar{\mathbf{G}}\boldsymbol{\delta})$. By the choice of \mathbf{m} , there must exist $\hat{\mathbf{m}} \in \mathbf{m}$ such that

$$\mathbf{s} = \text{mid}(\mathbf{J})\mathbf{x} + \hat{\mathbf{m}} + (\hat{\mathbf{J}} - \text{mid}(\mathbf{J}))\bar{\mathbf{G}}\boldsymbol{\delta}. \quad (4.9)$$

Let $\boldsymbol{\eta} = \hat{\mathbf{m}} + (\hat{\mathbf{J}} - \text{mid}(\mathbf{J}))\bar{\mathbf{G}}\boldsymbol{\delta}$. Then

$$\begin{aligned} \eta_i &= \hat{m}_i + \sum_{j=1}^{\bar{n}_g} ((\hat{\mathbf{J}} - \text{mid}(\mathbf{J}))\bar{\mathbf{G}})_{ij} \delta_j, \\ &= \hat{m}_i + \sum_{j=1}^{\bar{n}_g} \left(\sum_{k=1}^{n_z} (\hat{J}_{ik} - \text{mid}(J_{ik})) \bar{G}_{kj} \right) \delta_j. \end{aligned}$$

By the triangle inequality and the fact that $|\delta_j| \leq 1$,

$$\begin{aligned} |\eta_i| &\leq |\hat{m}_i| + \sum_{j=1}^{\bar{n}_g} \left(\sum_{k=1}^{n_z} |(\hat{J}_{ik} - \text{mid}(J_{ik}))| |\bar{M}_{kj}| \right) |\delta_j|, \\ &\leq \frac{1}{2} \text{diam}(m_i) + \frac{1}{2} \sum_{j=1}^{\bar{n}_g} \sum_{k=1}^{n_z} \text{diam}(J_{ik}) |\bar{M}_{kj}|. \end{aligned}$$

Therefore, $\boldsymbol{\eta} \in \mathbf{PB}_{\infty}^{n_s}$. From (4.9), this implies that

$$\mathbf{s} = \text{mid}(\mathbf{J})\mathbf{z} + \boldsymbol{\eta} \in \text{mid}(\mathbf{J})Z \oplus \mathbf{PB}_{\infty}^{n_s} = \triangleleft(\mathbf{J}, Z).$$

Thus $S \subseteq \triangleleft(\mathbf{J}, Z)$. ■

Remark 4.5. In Theorem 4.1, a zonotope \bar{Z} satisfying $Z \subseteq \bar{Z}$ can be easily obtained by eliminating all the n_c constraints from Z using the method in Scott et al. (2016). Moreover, \mathbf{m} can be obtained by simply evaluating $(\mathbf{J} - \text{mid}(\mathbf{J}))\bar{\mathbf{c}}$ with interval arithmetic. These methods are used in this doctoral thesis. Finally, the enclosure (4.8) has $n_g + n_s$ generators and n_c constraints.

The following theorem provides the mean value extension for constrained zonotopes.

Theorem 4.2. Let $\boldsymbol{\mu} : \mathbb{R}^n \times \mathbb{R}^{n_w} \rightarrow \mathbb{R}^n$ be continuously differentiable and $\nabla_x \boldsymbol{\mu}$ denote the gradient of $\boldsymbol{\mu}$ with respect to its first argument. Let $X \subset \mathbb{R}^n$ and $W \subset \mathbb{R}^{n_w}$ be constrained zonotopes and choose any $\boldsymbol{\gamma}_x \in X$. If Z_w is a constrained zonotope such that $\boldsymbol{\mu}(\boldsymbol{\gamma}_x, W) \subseteq Z_w$, and $\mathbf{J}_x \in \mathbb{I}\mathbb{R}^{n \times n}$ is an interval matrix satisfying $\nabla_x^T \boldsymbol{\mu}(X, W) \subseteq \mathbf{J}_x$, then

$$\boldsymbol{\mu}(X, W) \subseteq Z_w \oplus \triangleleft(\mathbf{J}_x, X - \boldsymbol{\gamma}_x).$$

Proof. Choose any $(\mathbf{x}, \mathbf{w}) \in X \times W$. It will be shown that $\boldsymbol{\mu}(\mathbf{x}, \mathbf{w}) \in Z \oplus \triangleleft(\mathbf{J}_x, X - \gamma_x)$. For any $i = 1, 2, \dots, n$, the Mean Value Theorem ensures that $\exists \boldsymbol{\delta}^{[i]} \in X$ such that

$$\mu_i(\mathbf{x}, \mathbf{w}) = \mu_i(\gamma_x, \mathbf{w}) + \nabla_x^T \mu_i(\boldsymbol{\delta}^{[i]}, \mathbf{w})(\mathbf{x} - \gamma_x).$$

But the vector $\nabla_x^T \mu_i(\boldsymbol{\delta}^{[i]}, \mathbf{w})$ is contained in the i -th row of \mathbf{J}_x by hypothesis, and since this is true for all $i = 1, 2, \dots, n$, there exists a real matrix $\hat{\mathbf{J}}_x \in \mathbf{J}_x$ such that $\boldsymbol{\mu}(\mathbf{x}, \mathbf{w}) = \boldsymbol{\mu}(\gamma_x, \mathbf{w}) + \hat{\mathbf{J}}_x(\mathbf{x} - \gamma_x)$.

By Theorem 4.1 and the choice of Z , it follows that $\boldsymbol{\mu}(\mathbf{x}, \mathbf{w}) \in Z_w \oplus \triangleleft(\mathbf{J}, X - \gamma_x)$, as desired. \blacksquare

Remark 4.6. The interval matrix \mathbf{J}_x required by Theorem 4.2 can be obtained by computing the interval hulls of X and W as in Property 3.3 and, then, bounding $\nabla_x^T \boldsymbol{\mu}(X, W)$ using interval arithmetic. Similarly, the constrained zonotope $Z_w \supseteq \boldsymbol{\mu}(\gamma_x, W)$ can be obtained by bounding $\boldsymbol{\mu}(\gamma_x, W)$ with interval arithmetic. Alternatively, another mean value extension can be applied around some $\gamma_w \in W$ to obtain $\boldsymbol{\mu}(\gamma_x, W) \subseteq Z_w \triangleq \boldsymbol{\mu}(\gamma_x, \gamma_w) \oplus \triangleleft(\mathbf{J}_w, W - \gamma_w)$, where \mathbf{J}_w is an interval enclosure of $\nabla_w^T \boldsymbol{\mu}(\gamma_x, W)$. Finally, if $\boldsymbol{\mu}$ is affine in \mathbf{w} , i.e., $\boldsymbol{\mu}(\mathbf{x}, \mathbf{w}) \triangleq \boldsymbol{\beta}_x(\mathbf{x}) + \mathbf{B}_w(\mathbf{x})\mathbf{w}$, then an exact enclosure of $\boldsymbol{\mu}(\gamma_x, W)$ is $Z_w = \boldsymbol{\beta}_x(\gamma_x) \oplus \mathbf{B}_w(\gamma_x)W$, which is a particular case of the mean value extension.

Since the CG-rep (3.7) is an alternative representation for convex polytopes (Scott et al., 2016), the mean value extension developed in Theorem 4.2 provides a new method for propagating convex polytopes implicitly through nonlinear mappings. A related approach can be found in Wan et al. (2018), where convex polytopes are represented by intersections of zonotopes (i.e., zonotope bundles (Althoff & Krogh, 2011)). However, while effective complexity reduction algorithms are available for constrained zonotopes (Scott et al., 2016), efficient methods for complexity control of zonotope bundles have not yet been proposed.

Remark 4.7. The enclosure obtained in Theorem 4.2 has at most $n_g + n_{g_w} + 2n$ generators and $n_c + n_{c_w}$ constraints (considering Z_w computed as in the alternatives presented in Remark 4.6), with n_g and n_{g_w} denoting the number of generators of X and W , and n_c and n_{c_w} the number of constraints, respectively. Thus, the complexity of the resulting set increases linearly with respect to the number of constraints and generators.

4.2.2.1 Selection of approximation point

The method proposed in this section requires a choice of $\gamma_x \in X = \{\mathbf{G}_x, \mathbf{c}_x, \mathbf{A}_x, \mathbf{b}_x\}$ in order to compute a constrained zonotope enclosure for the prediction step (4.2). As shown in Section 4.5.2, this choice may drastically affect the accuracy of the obtained enclosure.

In the mean value extension for intervals and zonotopes, a usual choice of $\gamma_x \in X$ is the center of X (Alamo et al., 2005; Moore et al., 2009). However, with constrained

zonotopes, since the center of the CG-rep may not belong to X^2 , a different point $\gamma_x \in X$ must be chosen. A simple and inexpensive choice is the center of the interval hull of X . Unfortunately, even this point may not belong to X in some cases³. Nevertheless, this choice can be applied rigorously by simply checking if $\gamma_x \in X$ beforehand by solving an LP (Scott et al., 2016).

In the following, we analyze alternative choices of $\gamma_x \in X$ valid for the mean value extension (Theorem 4.2). This extension relies on the CZ-inclusion operator (Theorem 4.1) and, therefore, requires the computation of a zonotope enclosing $X - \gamma_x$. In this work, we assume that this zonotope is computed through constraint elimination (see Remark 4.5). Let $\{\mathbf{G}^{(\ell)}, \mathbf{c}^{(\ell)}, \mathbf{A}^{(\ell)}, \mathbf{b}^{(\ell)}\}$ denote the constrained zonotope obtained by reducing to ℓ the number of remaining constraints in $X - \gamma_x$. Following the constraint elimination algorithm in Scott et al. (2016), for each $\ell = n_c, n_c - 1, \dots, 1$, the remaining constraints $\mathbf{A}^{(\ell)} \boldsymbol{\xi} = \mathbf{b}^{(\ell)}$ are first preconditioned through Gauss-Jordan elimination with full pivoting and then subjected to a rescaling procedure before the next constraint is eliminated. The entire procedure can be represented by the following recursive equations (see Proposition 5 and the Appendix in Scott et al. (2016) for details), where $\bar{(\cdot)}$ denotes variables after preconditioning, $\tilde{(\cdot)}$ denotes variables after rescaling, and $\boldsymbol{\Lambda}_G$, $\boldsymbol{\Lambda}_A$, $\boldsymbol{\xi}_m$, and $\boldsymbol{\xi}_r$ are defined as in Scott et al. (2016):

$$\begin{aligned} \tilde{\mathbf{c}}^{(\ell)} &= \mathbf{c}^{(\ell)} + \bar{\mathbf{G}}^{(\ell)} \boldsymbol{\xi}_m^{(\ell)}, & \mathbf{c}^{(\ell-1)} &= \tilde{\mathbf{c}}^{(\ell)} + \boldsymbol{\Lambda}_G^{(\ell)} \tilde{\mathbf{b}}^{(\ell)}, \\ \tilde{\mathbf{G}}^{(\ell)} &= \bar{\mathbf{G}}^{(\ell)} \text{diag}(\boldsymbol{\xi}_r^{(\ell)}), & \mathbf{G}^{(\ell-1)} &= \tilde{\mathbf{G}}^{(\ell)} - \boldsymbol{\Lambda}_G^{(\ell)} \tilde{\mathbf{A}}^{(\ell)}, \\ \tilde{\mathbf{A}}^{(\ell)} &= \bar{\mathbf{A}}^{(\ell)} \text{diag}(\boldsymbol{\xi}_r^{(\ell)}), & \mathbf{A}^{(\ell-1)} &= \tilde{\mathbf{A}}^{(\ell)} - \boldsymbol{\Lambda}_A^{(\ell)} \tilde{\mathbf{A}}^{(\ell)}, \\ \tilde{\mathbf{b}}^{(\ell)} &= \bar{\mathbf{b}}^{(\ell)} - \bar{\mathbf{A}}^{(\ell)} \boldsymbol{\xi}_m^{(\ell)}, & \mathbf{b}^{(\ell-1)} &= \tilde{\mathbf{b}}^{(\ell)} - \boldsymbol{\Lambda}_A^{(\ell)} \tilde{\mathbf{b}}^{(\ell)}. \end{aligned} \tag{4.10}$$

Careful examination of the algorithm in Scott et al. (2016) reveals that the actions taken during preconditioning, rescaling, and constraint elimination are all independent of the center of the original constrained zonotope, which in this case is $\mathbf{c}^{(n_c)} = \mathbf{c}_x - \gamma_x$. Therefore, with exception of the center, the variables $(\cdot)^{(\ell)}$ can be obtained by eliminating the constraints of X prior to choosing γ_x . Considering procedure (4.10), the following corollary provides a choice of γ_x that leads to a tight enclosure by reducing the conservativeness of the CZ-inclusion operator $\triangleleft (\mathbf{J}, X - \gamma_x)$.

Corollary 4.1. Let $X = \{\mathbf{G}_x, \mathbf{c}_x, \mathbf{A}_x, \mathbf{b}_x\} \subset \mathbb{R}^n$, and consider $\boldsymbol{\mu}$, W , and \mathbf{J} as defined in Theorem 4.2. Assume that $\bar{\mathbf{G}}^{(\ell)}$, $\boldsymbol{\xi}_m^{(\ell)}$, $\boldsymbol{\Lambda}_G^{(\ell)}$, and $\tilde{\mathbf{b}}^{(\ell)}$ are obtained by eliminating all n_c constraints from X according to (4.10), and set

$$\gamma_x = \mathbf{c}_x + \sum_{\ell=1}^{n_c} \left(\bar{\mathbf{G}}^{(\ell)} \boldsymbol{\xi}_m^{(\ell)} + \boldsymbol{\Lambda}_G^{(\ell)} \tilde{\mathbf{b}}^{(\ell)} \right). \tag{4.11}$$

²From $X = \mathbf{c}_x \oplus \mathbf{G}_x B_\infty(\mathbf{A}_x, \mathbf{b}_x)$, $\mathbf{c}_x \notin X$ as long as $\nexists \boldsymbol{\xi} \in B_\infty(\mathbf{A}_x, \mathbf{b}_x)$ satisfying $\mathbf{G}_x \boldsymbol{\xi} = \mathbf{0}$.

³An example is the polytope with vertices $(0, 0, 0)$, $(1, 1, 0)$, $(0, 1, 0)$, $(0, 1, 1)$.

Let $\bar{X} = \{\mathbf{G}^{(0)}, \mathbf{c}^{(0)}\}$ be obtained by eliminating all n_c constraints from $X - \gamma_x$ according to (4.10), let $\mathbf{m} \supseteq (\mathbf{J} - \text{mid}(\mathbf{J}))\mathbf{c}^{(0)}$ be computed by standard interval arithmetic, and suppose that $\triangleleft(\mathbf{J}, X - \gamma_x)$ is computed as in Theorem 4.1 with this choice of \bar{X} and \mathbf{m} . Finally, let $Z \supseteq \boldsymbol{\mu}(\gamma_x, W)$. If $\gamma_x \in X$, then $\boldsymbol{\mu}(X, W) \subseteq Z \oplus \triangleleft(\mathbf{J}, X - \gamma_x)$. Moreover, $\triangleleft(\mathbf{J}, X - \gamma_x) \subseteq \triangleleft(\mathbf{J}, X - \hat{\gamma}_x)$ for any $\hat{\gamma}_x \in X$, $\hat{\gamma}_x \neq \gamma_x$.

Proof. For $\gamma_x \in X$, $\boldsymbol{\mu}(X, W) \subseteq Z \oplus \triangleleft(\mathbf{J}, X - \gamma_x)$ follows directly from Theorem 4.2. Now, let us show that $\triangleleft(\mathbf{J}, X - \gamma_x) \subseteq \triangleleft(\mathbf{J}, X - \hat{\gamma}_x)$ holds for any $\hat{\gamma}_x \in X$, $\hat{\gamma}_x \neq \gamma_x$. Recursive computation of (4.10) leads to

$$\mathbf{c}^{(0)} = \mathbf{c}^{(n_c)} + \sum_{\ell=1}^{n_c} \left(\bar{\mathbf{G}}^{(\ell)} \boldsymbol{\xi}_m^{(\ell)} + \boldsymbol{\Lambda}_G^{(\ell)} \tilde{\mathbf{b}}^{(\ell)} \right), \quad (4.12)$$

where $\mathbf{c}^{(n_c)} = \mathbf{c}_x - \gamma_x$. Therefore, $\mathbf{c}^{(0)} = \mathbf{0}$ iff γ_x is given by (4.11), and then $\mathbf{m} = \mathbf{0}$, and $\text{diam}(\mathbf{m}) = \mathbf{0}$. Note that in (4.7), $\bar{\mathbf{M}} \triangleq \mathbf{G}^{(0)}$ is invariant with respect to γ_x , and since $\mathbf{J} \supseteq \nabla_x^T \boldsymbol{\mu}(X, W)$, then \mathbf{J} is also invariant with respect to γ_x . Consequently, the second term in (4.7) is not a function of γ_x . Therefore, $\mathbf{P}B_\infty^n \subseteq \hat{\mathbf{P}}B_\infty^n = (1/2)\text{diag}(\text{diam}(\hat{\mathbf{m}}))B_\infty^n \oplus \mathbf{P}B_\infty^n$, with $\hat{\mathbf{m}}$ computed using $\hat{\gamma}_x \in X$. The result then follows from (4.8). ■

By Corollary 4.1, the enclosure obtained in Theorem 4.2 is tightened by choosing γ_x such that $\mathbf{c}^{(0)}$ is equal to zero. Unfortunately, the γ_x given by (4.11) may not belong to X . Then, an alternative to obtain tight bounds is to reduce the size of the box \mathbf{m} by solving

$$\min_{\gamma_x} \{\|\text{diam}(\mathbf{m})\|_1 : \gamma_x \in X\}, \quad (4.13)$$

with $\mathbf{m} \supseteq (\mathbf{J} - \text{mid}(\mathbf{J}))\bar{\mathbf{p}}$ computed using interval arithmetic, where $\bar{\mathbf{p}} \triangleq \mathbf{c}^{(0)}$. Recall that $\mathbf{c}^{(0)}$ is the center of the zonotope obtained by eliminating all the constraints of $X - \gamma_x$.

Lemma 4.1. Let $X = \{\mathbf{G}_x, \mathbf{c}_x, \mathbf{A}_x, \mathbf{b}_x\} \subset \mathbb{R}^n$, $\mathbf{J} \in \mathbb{I}\mathbb{R}^{n \times n}$. Assume that $\bar{\mathbf{G}}^{(\ell)}$, $\boldsymbol{\xi}_m^{(\ell)}$, $\boldsymbol{\Lambda}_G^{(\ell)}$, and $\tilde{\mathbf{b}}^{(\ell)}$ are obtained by eliminating all n_c constraints of X according to (4.10). Then, $\gamma_x = \mathbf{c}_x + \mathbf{G}_x \boldsymbol{\xi}^*$ is the solution to (4.13) iff $\boldsymbol{\xi}^*$ is the solution to the linear program

$$\min_{\boldsymbol{\xi}} \|\Theta \bar{\mathbf{p}}\|_1, \quad \text{s.t.} \quad \mathbf{A}_x \boldsymbol{\xi} = \mathbf{b}_x, \quad \|\boldsymbol{\xi}\|_\infty \leq 1, \quad (4.14)$$

with $\bar{\mathbf{p}} = -\mathbf{G}_x \boldsymbol{\xi} + \sum_{\ell=1}^{n_c} \left(\bar{\mathbf{G}}^{(\ell)} \boldsymbol{\xi}_m^{(\ell)} + \boldsymbol{\Lambda}_G^{(\ell)} \tilde{\mathbf{b}}^{(\ell)} \right)$, $\Theta_{jj} = \sum_{i=1}^n \text{diam}(J_{ij})$, and $\Theta_{ij} = 0$ for $i \neq j$.

Proof. Each element of $(\mathbf{J} - \text{mid}(\mathbf{J})) \in \mathbb{I}\mathbb{R}^{n \times n}$ is a symmetric interval satisfying $(J_{ij} - \text{mid}(J_{ij})) = (1/2)\text{diam}(J_{ij})[-1, 1]$, and for every $a \in \mathbb{R}$, $a[-1, 1] = |a|[-1, 1]$ holds. Therefore $m_i = \sum_{j=1}^n (1/2)\text{diam}(J_{ij})|\bar{p}_j|[-1, 1]$. Consequently, $\text{diam}(m_i) = \sum_{j=1}^n \text{diam}(J_{ij})|\bar{p}_j|$, and

$$\begin{aligned} \|\text{diam}(\mathbf{m})\|_1 &= \sum_{i=1}^n \sum_{j=1}^n \text{diam}(J_{ij})|\bar{p}_j| = \sum_{j=1}^n \left(\sum_{i=1}^n \text{diam}(J_{ij}) \right) |\bar{p}_j| \\ &= \sum_{j=1}^n \Theta_{jj} |\bar{p}_j| = \|\Theta \bar{\mathbf{p}}\|_1. \end{aligned}$$

The equality $\bar{\mathbf{p}} = -\mathbf{G}_x \boldsymbol{\xi} + \sum_{\ell=1}^{n_c} \left(\bar{\mathbf{G}}^{(\ell)} \boldsymbol{\xi}_m^{(\ell)} + \boldsymbol{\Lambda}_G^{(\ell)} \tilde{\mathbf{b}}^{(\ell)} \right)$ and the constraints in (4.14) follow directly from (4.12) and $\boldsymbol{\gamma}_x \in X$. \blacksquare

Lemma 4.1 yields an optimal choice of $\boldsymbol{\gamma}_x \in X$ that can be used in Theorem 4.2 to reduce conservatism in the CZ-inclusion operator and requires only the solution of an LP. Note that formulating (4.14) requires the knowledge of $\bar{\mathbf{G}}^{(\ell)}$, $\boldsymbol{\xi}_m^{(\ell)}$, $\boldsymbol{\Lambda}_G^{(\ell)}$, and $\tilde{\mathbf{b}}^{(\ell)}$, which are obtained from the iterated constraint elimination process. As stated before, constraint elimination can be performed over X to obtain the required data prior to the solution of (4.14). Once the optimal $\boldsymbol{\gamma}_x$ is obtained, constraint elimination can be repeated, or equivalently, the zonotope obtained using $\boldsymbol{\gamma}_x = \mathbf{0}$ can simply be translated by $-\boldsymbol{\gamma}_x$.

Remark 4.8. Note that if the $\boldsymbol{\gamma}_x$ given by Corollary 4.1 belongs to X , then this coincides with the solution provided by Lemma 4.1.

We summarize the proposed choices of $\boldsymbol{\gamma}_x \in X$ for use in Theorem 4.2 as follows:

C1) $\boldsymbol{\gamma}_x$ is given by the center of the interval hull of X if it satisfies $\boldsymbol{\gamma}_x \in X$;

C2) $\boldsymbol{\gamma}_x$ is obtained by solving (4.14).

A comparison between the different choices of $\boldsymbol{\gamma}_x \in X$ is illustrated in Section 4.5.2.

4.2.3 First-order Taylor extension

This section presents the third new method for enclosing the range of a nonlinear function $\boldsymbol{\eta}$ over a set of inputs described by constrained zonotopes. This method is referred to as the *first-order Taylor extension* of $\boldsymbol{\eta}$ because it relies on a first-order Taylor expansion with a rigorous remainder bound, and is a consistent generalization of the zonotope-based method proposed in Combastel (2005). In contrast to Theorem 4.2, for the sake of simplicity of the proof, this function has only one argument. Even so, it is possible to consider both states and process uncertainties by concatenating them into a single vector. Due to substantial changes with respect to the zonotopic method, the new approach comes with a new theorem (Theorem 4.3) and a detailed proof. In the main result below, $(\cdot)_{i,:}$ denotes the i -th row of a matrix, and $(\cdot)_{ij}$ denotes the element from its i -th row and j -th column.

Theorem 4.3. Let $\boldsymbol{\eta} : \mathbb{R}^{n_z} \rightarrow \mathbb{R}^n$ be of class \mathcal{C}^2 and $\mathbf{z} \in \mathbb{R}^{n_z}$ denote its argument. Let $Z = \{\mathbf{G}, \mathbf{c}, \mathbf{A}, \mathbf{b}\} \subset \mathbb{R}^{n_z}$ be a constrained zonotope with m_g generators and m_c constraints. For each $q = 1, 2, \dots, n$, let $\mathbf{Q}^{[q]} \in \mathbb{I}\mathbb{R}^{n_g \times n_g}$ and $\tilde{\mathbf{Q}}^{[q]} \in \mathbb{I}\mathbb{R}^{m_g \times m_g}$ be interval matrices satisfying $\mathbf{Q}^{[q]} \supseteq \mathbf{H}\boldsymbol{\eta}_q(Z)$ and $\tilde{\mathbf{Q}}^{[q]} \supseteq \mathbf{G}^T \mathbf{Q}^{[q]} \mathbf{G}$. Moreover, define

$$\tilde{c}_q = \text{trace} \left(\text{mid}(\tilde{\mathbf{Q}}^{[q]}) \right) / 2, \quad \tilde{\mathbf{G}}_{q,:} = \left[\cdots \underbrace{\text{mid}(\tilde{Q}_{ii}^{[q]})/2}_{\forall i} \cdots \underbrace{\left(\text{mid}(\tilde{Q}_{ij}^{[q]}) + \text{mid}(\tilde{Q}_{ji}^{[q]}) \right)}_{\forall i < j} \cdots \right],$$

$$\tilde{\mathbf{G}}_d = \text{diag}(\mathbf{d}), \quad d_q = \sum_{i,j} \left| \text{rad}(\tilde{Q}_{ij}^{[q]}) \right|, \quad \tilde{\mathbf{A}} = \left[\tilde{\mathbf{A}}_\zeta \quad \tilde{\mathbf{A}}_\xi \quad \mathbf{0}_{\frac{m_c}{2}(1+m_c) \times n} \right],$$

$$\begin{aligned} \tilde{\mathbf{A}}_{\boldsymbol{\zeta}} &= \begin{bmatrix} \vdots \\ \cdots & \frac{1}{2}A_{ri}A_{si} & \cdots \\ \vdots \end{bmatrix} \Bigg|_{\forall r \leq s}, & \tilde{\mathbf{A}}_{\boldsymbol{\xi}} &= \begin{bmatrix} \vdots \\ \cdots & A_{ri}A_{sj} + A_{rj}A_{si} & \cdots \\ \vdots \end{bmatrix} \Bigg|_{\forall r \leq s}, \\ & \underbrace{\hspace{10em}}_{\forall i} & & \underbrace{\hspace{10em}}_{\forall i < j} \\ \tilde{\mathbf{b}} &= \begin{bmatrix} \vdots \\ b_r b_s - \frac{1}{2} \sum_i A_{ri} A_{si} \\ \vdots \end{bmatrix} \Bigg|_{\forall r \leq s}, \end{aligned}$$

with indices $i, j = 1, 2, \dots, m_g$ and $r, s = 1, 2, \dots, m_c$. Finally, choose any $\boldsymbol{\gamma}_z \in Z$ and let $\mathbf{L} \in \mathbb{IR}^{n \times m}$ be an interval matrix satisfying $\mathbf{L}_{q,:} \supseteq (\mathbf{c} - \boldsymbol{\gamma}_z)^T \mathbf{Q}^{[q]}$ for all $q = 1, \dots, n$. Then,

$$\boldsymbol{\eta}(Z) \subseteq \boldsymbol{\eta}(\boldsymbol{\gamma}_z) \oplus \nabla^T \boldsymbol{\eta}(\boldsymbol{\gamma}_z)(Z - \boldsymbol{\gamma}_z) \oplus R, \quad (4.15)$$

where $R = \tilde{\mathbf{c}} \oplus [\tilde{\mathbf{G}} \ \tilde{\mathbf{G}}_d] B_\infty(\tilde{\mathbf{A}}, \tilde{\mathbf{b}}) \oplus \triangleleft(\mathbf{L}, (\mathbf{c} - \boldsymbol{\gamma}_z) \oplus 2\mathbf{G}B_\infty(\mathbf{A}, \mathbf{b}))$.

Proof. Choose any $\mathbf{z} \in Z$ and $q \in \{1, \dots, n\}$. By Taylor's theorem applied to η_q with reference point $\boldsymbol{\gamma}_z$, there must exist $\boldsymbol{\Gamma}^{[q]} \in \mathbf{H}\eta_q(Z) \subseteq \mathbf{Q}^{[q]}$ such that⁴

$$\eta_q(\mathbf{z}) = \eta_q(\boldsymbol{\gamma}_z) + \nabla^T \eta_q(\boldsymbol{\gamma}_z)(\mathbf{z} - \boldsymbol{\gamma}_z) + (\mathbf{z} - \boldsymbol{\gamma}_z)^T \boldsymbol{\Gamma}^{[q]}(\mathbf{z} - \boldsymbol{\gamma}_z).$$

Since $\mathbf{z} \in Z$, there must exist $\boldsymbol{\xi} \in B_\infty(\mathbf{A}, \mathbf{b})$ such that $\mathbf{z} = \mathbf{c} + \mathbf{G}\boldsymbol{\xi}$. Thus, defining $\mathbf{p} = \mathbf{c} - \boldsymbol{\gamma}_z$ for brevity,

$$\eta_q(\mathbf{z}) = \eta_q(\boldsymbol{\gamma}_z) + \nabla^T \eta_q(\boldsymbol{\gamma}_z)(\mathbf{p} + \mathbf{G}\boldsymbol{\xi}) + (\mathbf{p} + \mathbf{G}\boldsymbol{\xi})^T \boldsymbol{\Gamma}^{[q]}(\mathbf{p} + \mathbf{G}\boldsymbol{\xi}).$$

Expanding the product $(\mathbf{p} + \mathbf{G}\boldsymbol{\xi})^T \boldsymbol{\Gamma}^{[q]}(\mathbf{p} + \mathbf{G}\boldsymbol{\xi})$ yields $\mathbf{p}^T \boldsymbol{\Gamma}^{[q]} \mathbf{p} + 2\boldsymbol{\xi}^T \tilde{\boldsymbol{\Gamma}}^{[q]} \boldsymbol{\xi}$, with $\tilde{\boldsymbol{\Gamma}}^{[q]} = \mathbf{G}^T \boldsymbol{\Gamma}^{[q]} \mathbf{G} \in \tilde{\mathbf{Q}}^{[q]}$. Since $\tilde{\boldsymbol{\Gamma}}^{[q]} \in \tilde{\mathbf{Q}}^{[q]}$, it follows that $\tilde{\Gamma}_{ij}^{[q]} = \text{mid}(\tilde{Q}_{ij}^{[q]}) + \text{rad}(\tilde{Q}_{ij}^{[q]})\Lambda_{ij}^{[q]}$ for some $\Lambda_{ij}^{[q]} \in B_\infty^1$. Additionally, $\xi_i \in [-1, 1]$ implies that $\xi_i^2 \in [0, 1]$, and hence $\xi_i^2 = \frac{1}{2} + \frac{1}{2}\zeta_i$ for some $\zeta_i \in [-1, 1]$. Considering these two facts,

$$\begin{aligned} \boldsymbol{\xi}^T \tilde{\boldsymbol{\Gamma}}^{[q]} \boldsymbol{\xi} &= \frac{1}{2} \sum_i \text{mid}(\tilde{Q}_{ii}^{[q]}) + \frac{1}{2} \sum_i \text{mid}(\tilde{Q}_{ii}^{[q]})\zeta_i \\ &\quad + \sum_{i < j} (\text{mid}(\tilde{Q}_{ij}^{[q]}) + \text{mid}(\tilde{Q}_{ji}^{[q]}))\xi_i \xi_j + \sum_{i, j} \text{rad}(\tilde{Q}_{ij}^{[q]})\xi_i \xi_j \Lambda_{ij}^{[q]}, \end{aligned}$$

where the third summation results from the fact that $\xi_i \xi_j = \xi_j \xi_i$. Thus, by defining the new generator variables

$$\bar{\boldsymbol{\xi}} = \left[\cdots \underbrace{\zeta_i}_{\forall i} \cdots \underbrace{\xi_i \xi_j}_{\forall i < j} \cdots \underbrace{\xi_i \xi_j \Lambda_{ij}^{[q]}}_{\forall i, j, q} \cdots \right]^T,$$

with $i, j = 1, 2, \dots, m_g$, $q = 1, 2, \dots, n$, we have that $\boldsymbol{\xi}^T \tilde{\boldsymbol{\Gamma}}^{[q]} \boldsymbol{\xi} = \tilde{c}_q + [\tilde{\mathbf{G}} \ \tilde{\mathbf{G}}_d]_{q,:} \bar{\boldsymbol{\xi}}$, where

⁴Let $\boldsymbol{\Upsilon}^{[q]}$ belong to the standard Hessian matrix of $\eta_q(Z)$. Then, $(1/2)(\mathbf{z} - \boldsymbol{\gamma}_z)^T \boldsymbol{\Upsilon}^{[q]}(\mathbf{z} - \boldsymbol{\gamma}_z) = (\mathbf{z} - \boldsymbol{\gamma}_z)^T \boldsymbol{\Gamma}^{[q]}(\mathbf{z} - \boldsymbol{\gamma}_z)$ holds. See (Combastel, 2005) for a motivation on this approach.

$\bar{\mathbf{G}}_d = \text{blkdiag}(\mathbf{N}^{[1]}, \mathbf{N}^{[2]}, \dots, \mathbf{N}^{[n]})^5$,

$$\mathbf{N}^{[q]} = [\dots \underbrace{\text{rad}(\tilde{Q}_{ij}^{[q]})}_{\forall i,j} \dots] \in \mathbb{R}^{1 \times m_g^2}.$$

Therefore, we have established that $\eta_q(\mathbf{z}) = \eta_q(\gamma_z) + \nabla^T \eta_q(\gamma_z)(\mathbf{z} - \gamma_z) + \mathbf{p}^T \mathbf{\Gamma}^{[q]}(\mathbf{p} + 2\mathbf{G}\boldsymbol{\xi}) + \tilde{\mathbf{c}}_q + [\tilde{\mathbf{G}} \ \bar{\mathbf{G}}_d]_{q,:} \bar{\boldsymbol{\xi}}$. This holds for every $q = 1, 2, \dots, n$. Moreover, \mathbf{L} satisfies $\mathbf{L}_{q,:} \supseteq \mathbf{p}^T \mathbf{Q}^{[q]}$ for all $q = 1, 2, \dots, n$ by definition, so there must exist $\hat{\mathbf{L}} \in \mathbf{L}$ such that $\hat{\mathbf{L}}_{q,:} = \mathbf{p}^T \mathbf{\Gamma}^{[q]}$ for all $q = 1, 2, \dots, n$. Therefore,

$$\boldsymbol{\eta}(\mathbf{z}) = \boldsymbol{\eta}(\gamma_z) + \nabla^T \boldsymbol{\eta}(\gamma_z)(\mathbf{z} - \gamma_z) + \hat{\mathbf{L}}(\mathbf{p} + 2\mathbf{G}\boldsymbol{\xi}) + \tilde{\mathbf{c}} + [\tilde{\mathbf{G}} \ \bar{\mathbf{G}}_d] \bar{\boldsymbol{\xi}}. \quad (4.16)$$

Furthermore, the equality constraints $\mathbf{A}\boldsymbol{\xi} = \mathbf{b}$ imply that $\mathbf{A}\boldsymbol{\xi}\boldsymbol{\xi}^T \mathbf{A}^T = \mathbf{b}\mathbf{b}^T$. Thus, considering $\xi_i^2 = \frac{1}{2} + \frac{1}{2}\zeta_i$, the r -th row and s -th column of this matrix equality yields

$$\frac{1}{2} \sum_i A_{ri} A_{si} \zeta_i + \sum_{i < j} (A_{ri} A_{sj} + A_{rj} A_{si}) \xi_i \xi_j = b_r b_s - \frac{1}{2} \sum_i A_{ri} A_{si},$$

with $r, s = 1, 2, \dots, m_c$. Such constraints are linear in $\bar{\boldsymbol{\xi}}$ and non-repeating for $r \leq s$, therefore $\bar{\mathbf{A}}\bar{\boldsymbol{\xi}} = \tilde{\mathbf{b}}$ holds, where $\bar{\mathbf{A}} = [\tilde{\mathbf{A}}_\zeta \ \tilde{\mathbf{A}}_\xi \ \mathbf{0}_{\tilde{m}_c \times nm_g^2}]$, with $\tilde{m}_c = \frac{m_c}{2}(1 + m_c)$. Hence, $\boldsymbol{\xi} \in B_\infty(\mathbf{A}, \mathbf{b}) \implies \bar{\boldsymbol{\xi}} \in B_\infty(\bar{\mathbf{A}}, \tilde{\mathbf{b}})$. Combining this with (4.16), we have proven the enclosure $\boldsymbol{\eta}(Z) \subseteq \boldsymbol{\eta}(\gamma_z) \oplus \nabla^T \boldsymbol{\eta}(\gamma_z)(Z - \gamma_z) \oplus \mathbf{L}(\mathbf{p} \oplus 2\mathbf{G}B_\infty(\mathbf{A}, \mathbf{b})) \oplus \tilde{\mathbf{c}} \oplus [\tilde{\mathbf{G}} \ \bar{\mathbf{G}}_d] B_\infty(\bar{\mathbf{A}}, \tilde{\mathbf{b}})$.

In fact, this enclosure can be greatly simplified by noting that the columns of $\bar{\mathbf{A}}$ corresponding to the variables $[\dots \xi_i \xi_j \Lambda_{ij}^{[q]} \dots]$ are all zero, and hence

$$[\tilde{\mathbf{G}} \ \bar{\mathbf{G}}_d] B_\infty(\bar{\mathbf{A}}, \tilde{\mathbf{b}}) = \tilde{\mathbf{G}} B_\infty([\tilde{\mathbf{A}}_\zeta \ \tilde{\mathbf{A}}_\xi], \tilde{\mathbf{b}}) \oplus \bar{\mathbf{G}}_d B_\infty^{nm_g^2}.$$

Since $\bar{\mathbf{G}}_d$ is block diagonal and each $\mathbf{N}^{[q]}$ is a row vector, $\bar{\mathbf{G}}_d B_\infty^{nm_g^2}$ is an interval and is equivalent to $\tilde{\mathbf{G}}_d B_\infty^n$, with $\tilde{\mathbf{G}}_d$ defined as in the statement of the theorem. Thus,

$$\begin{aligned} [\tilde{\mathbf{G}} \ \bar{\mathbf{G}}_d] B_\infty(\bar{\mathbf{A}}, \tilde{\mathbf{b}}) &= \tilde{\mathbf{G}} B_\infty([\tilde{\mathbf{A}}_\zeta \ \tilde{\mathbf{A}}_\xi], \tilde{\mathbf{b}}) \oplus \tilde{\mathbf{G}}_d B_\infty^n, \\ &= [\tilde{\mathbf{G}} \ \tilde{\mathbf{G}}_d] B_\infty([\tilde{\mathbf{A}}_\zeta \ \tilde{\mathbf{A}}_\xi \ \mathbf{0}_{\tilde{m}_c \times n}], \tilde{\mathbf{b}}) = [\tilde{\mathbf{G}} \ \tilde{\mathbf{G}}_d] B_\infty(\tilde{\mathbf{A}}, \tilde{\mathbf{b}}). \end{aligned}$$

Therefore, $\boldsymbol{\eta}(Z) \subseteq \boldsymbol{\eta}(\gamma_z) \oplus \nabla^T \boldsymbol{\eta}(\gamma_z)(Z - \gamma_z) \oplus \mathbf{L}(\mathbf{p} \oplus 2\mathbf{G}B_\infty(\mathbf{A}, \mathbf{b})) \oplus \tilde{\mathbf{c}} \oplus [\tilde{\mathbf{G}} \ \tilde{\mathbf{G}}_d] B_\infty(\tilde{\mathbf{A}}, \tilde{\mathbf{b}})$, and (4.15) follows immediately from the definition of R . \blacksquare

Remark 4.9. Regarding the definitions of $\tilde{\mathbf{G}}$, $\tilde{\mathbf{A}}_\zeta$, $\tilde{\mathbf{A}}_\xi$, and $\tilde{\mathbf{b}}$ in Theorem 4.3, the ordering of the indices $i < j$ and $r \leq s$ is irrelevant, as long as it is the same for all variables.

Remark 4.10. The interval matrices $\mathbf{Q}^{[q]}$ required by Theorem 4.3 can be obtained by computing the interval hull of Z (Property 3.3) and then bounding $\mathbf{H}\eta_q(Z)$ using interval

⁵In this work $\text{blkdiag}(\mathbf{A}, \mathbf{B}, \dots)$ denotes a block diagonal matrix with blocks $\mathbf{A}, \mathbf{B}, \dots$.

arithmetic. Moreover, $\tilde{\mathbf{Q}}^{[q]}$ and \mathbf{L} can be obtained by evaluating $\mathbf{G}^T \mathbf{Q}^{[q]} \mathbf{G}$ and $(\mathbf{c} - \boldsymbol{\gamma}_z)^T \mathbf{Q}^{[q]}$ using interval arithmetic.

As stated before, process disturbances can be taken into account in (4.15) by considering the augmented vector $\mathbf{z} = (\mathbf{x}, \mathbf{w})$ with $Z \triangleq X \times W \subset \mathbb{R}^{n+n_w}$ and $\boldsymbol{\gamma}_z \triangleq (\boldsymbol{\gamma}_x, \boldsymbol{\gamma}_w) \in Z$. With $X = \{\mathbf{G}_x, \mathbf{c}_x, \mathbf{A}_x, \mathbf{b}_x\}$ and $W = \{\mathbf{G}_w, \mathbf{c}_w, \mathbf{A}_w, \mathbf{b}_w\}$, the Cartesian product Z is easily computed by

$$X \times W = \left\{ \left[\begin{array}{cc} \mathbf{G}_x & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_w \end{array} \right], \left[\begin{array}{c} \mathbf{c}_x \\ \mathbf{c}_w \end{array} \right], \left[\begin{array}{cc} \mathbf{A}_x & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_w \end{array} \right], \left[\begin{array}{c} \mathbf{b}_x \\ \mathbf{b}_w \end{array} \right] \right\}.$$

Remark 4.11. In Theorem 4.3, $\tilde{\mathbf{G}}$ has $\sum_{j=1}^{m_g} j = \frac{1}{2}m_g(m_g + 1)$ columns, $\tilde{\mathbf{G}}_d \in \mathbb{R}^{n \times n}$, $\tilde{\mathbf{A}}$ has $\sum_{s=1}^{m_c} s = \frac{1}{2}m_c(m_c + 1)$ rows, and $\triangleleft(\mathbf{L}, (\mathbf{c} - \boldsymbol{\gamma}_z) \oplus 2\mathbf{G}B_\infty(\mathbf{A}, \mathbf{b}))$ has $m_g + n$ generators and m_c constraints (Remark 4.5). Therefore, the resulting enclosure in (4.15) has $\frac{1}{2}m_g^2 + \frac{5}{2}m_g + 2n$ generators and $\frac{1}{2}m_c^2 + \frac{5}{2}m_c$ constraints. If $Z \triangleq X \times W$, then the enclosure has $\frac{1}{2}(n_g + n_{g_w})^2 + \frac{5}{2}(n_g + n_{g_w}) + 2n$ generators and $\frac{1}{2}(n_c + n_{c_w})^2 + \frac{5}{2}(n_c + n_{c_w})$ constraints, which is a polynomial increase in complexity in terms of both generators and constraints.

4.2.3.1 Selection of approximation point

In this section, we focus on proposing suitable choices of $\boldsymbol{\gamma}_z \triangleq (\boldsymbol{\gamma}_x, \boldsymbol{\gamma}_w) \in X \times W$ valid for the first-order Taylor extension (Theorem 4.3). As with the mean value extension, the usual choice of $\boldsymbol{\gamma}_z \in X \times W$ in first-order Taylor extensions for intervals and zonotopes is the center of X (Moore et al., 2009; Combastel, 2005). The next corollary shows that this choice leads to a tight enclosure if it in fact belongs to $X \times W$.

Corollary 4.2. Let $Z = \{\mathbf{G}, \mathbf{c}, \mathbf{A}, \mathbf{b}\} = X \times W \subset \mathbb{R}^{n+n_w}$, and consider $\boldsymbol{\eta}$, $\tilde{\mathbf{c}}$, $\tilde{\mathbf{G}}$, $\tilde{\mathbf{G}}_d$, $\tilde{\mathbf{A}}$, $\tilde{\mathbf{b}}$, \mathbf{L} , and $\mathbf{Q}^{[q]}$ as defined in Theorem 4.3, with $q \in \{1, 2, \dots, n\}$. If $\boldsymbol{\gamma}_z = \mathbf{c} \in Z$, then $\boldsymbol{\eta}(Z) \subseteq \boldsymbol{\eta}(\boldsymbol{\gamma}_z) \oplus \nabla^T \boldsymbol{\eta}(\boldsymbol{\gamma}_z)(Z - \boldsymbol{\gamma}_z) \oplus \tilde{\mathbf{c}} \oplus [\tilde{\mathbf{G}} \tilde{\mathbf{G}}_d]B_\infty(\tilde{\mathbf{A}}, \tilde{\mathbf{b}})$.

Proof. For $\boldsymbol{\gamma}_z = \mathbf{c}$, $\mathbf{L}_{q,:} \supseteq (\mathbf{c} - \boldsymbol{\gamma}_z)^T \mathbf{Q}^{[q]} = \mathbf{0}$, $q = 1, 2, \dots, n$. Therefore $\mathbf{L} = \mathbf{0}$ holds, and $\triangleleft(\mathbf{L}, (\mathbf{c} - \boldsymbol{\gamma}_z) \oplus 2\mathbf{G}B_\infty(\mathbf{A}, \mathbf{b})) = \{\mathbf{0}\}$. The result then follows from (4.15). ■

By inspecting Corollary 4.2, it is clear that the enclosure in (4.15) is tightened since $\triangleleft(\mathbf{L}, (\mathbf{c} - \boldsymbol{\gamma}_z) \oplus 2\mathbf{G}B_\infty(\mathbf{A}, \mathbf{b})) = \{\mathbf{0}\}$. However, the presence of the equality constraints in $X \times W$ may imply that its center is not in $X \times W$ (i.e., when $\nexists \boldsymbol{\xi} \in B_\infty(\mathbf{A}, \mathbf{b})$ such that $\mathbf{G}\boldsymbol{\xi} = \mathbf{0}$). Therefore, a good alternative may be to consider the closest point in $X \times W$ to its center, obtained by means of Proposition 4.3. By the definition of \mathbf{L} , this heuristic leads to smaller values of $\text{diam}(\mathbf{L})$, and therefore reduces the size of $\triangleleft(\mathbf{L}, (\mathbf{c} - \boldsymbol{\gamma}_z) \oplus 2\mathbf{G}B_\infty(\mathbf{A}, \mathbf{b}))$ (see (4.7)).

Proposition 4.3. Let $Z = \{\mathbf{G}, \mathbf{c}, \mathbf{A}, \mathbf{b}\} \subset \mathbb{R}^n$ and $\mathbf{s} \in \mathbb{R}^n$. A point $\mathbf{z} \in Z$ that minimizes $\|\mathbf{z} - \mathbf{s}\|_1$ is given by $\mathbf{z}^* = \mathbf{c} + \mathbf{G}\boldsymbol{\xi}^*$, where $\boldsymbol{\xi}^*$ is a solution to the linear program

$$\min_{\boldsymbol{\xi}} \|\mathbf{c} - \mathbf{s} + \mathbf{G}\boldsymbol{\xi}\|_1 \quad \text{s.t.} \quad \mathbf{A}\boldsymbol{\xi} = \mathbf{b}, \quad \|\boldsymbol{\xi}\|_\infty \leq 1.$$

Proof. By definition,

$$\begin{aligned}\|\mathbf{z}^* - \mathbf{s}\|_1 &= \|\mathbf{c} - \mathbf{s} + \mathbf{G}\boldsymbol{\xi}^*\|_1 \\ &\leq \|\mathbf{c} - \mathbf{s} + \mathbf{G}\boldsymbol{\xi}\|_1, \quad \forall \boldsymbol{\xi} \in B_\infty(\mathbf{A}, \mathbf{b}).\end{aligned}$$

But, for any $\mathbf{z} \in Z$, there exists $\boldsymbol{\xi} \in B_\infty(\mathbf{A}, \mathbf{b})$ such that $\mathbf{z} = \mathbf{c} + \mathbf{G}\boldsymbol{\xi}$. Therefore, $\|\mathbf{z}^* - \mathbf{s}\|_1 \leq \|\mathbf{z} - \mathbf{s}\|_1$, $\forall \mathbf{z} \in Z$. \blacksquare

On the other hand, a third option is to use Proposition 4.4 to obtain an alternative CG-rep of $X \times W$ with any desired center. In this case, the new center is chosen as some point in X , $\bar{\gamma}_z \in X \times W$, and then γ_z is chosen as $\gamma_z = \bar{\gamma}_z$.

Proposition 4.4 (Move center). Let $Z = \{\mathbf{G}, \mathbf{c}, \mathbf{A}, \mathbf{b}\} \subset \mathbb{R}^n$, and let $\tilde{\boldsymbol{\xi}}^L, \tilde{\boldsymbol{\xi}}^U \in \mathbb{R}^{n_g}$ satisfy $B_\infty(\mathbf{A}, \mathbf{b}) \subseteq [\tilde{\boldsymbol{\xi}}^L, \tilde{\boldsymbol{\xi}}^U]$. Choose any desired center $\hat{\mathbf{c}} \in \mathbb{R}^n$ lying in the range of \mathbf{G} and let $\boldsymbol{\xi}^L, \boldsymbol{\xi}^U \in \mathbb{R}^{n_g}$ be solutions to the linear program

$$\begin{aligned}\min_{\boldsymbol{\xi}^L, \boldsymbol{\xi}^U} & \left\| \frac{1}{2}(\boldsymbol{\xi}^U - \boldsymbol{\xi}^L) \right\|_1 \\ \text{s.t.} & \quad \mathbf{c} + \frac{1}{2}\mathbf{G}(\boldsymbol{\xi}^L + \boldsymbol{\xi}^U) = \hat{\mathbf{c}}, \quad \boldsymbol{\xi}^L \leq \tilde{\boldsymbol{\xi}}^L, \quad \boldsymbol{\xi}^U \geq \tilde{\boldsymbol{\xi}}^U.\end{aligned}$$

Letting $\boldsymbol{\xi}_m = \frac{1}{2}(\boldsymbol{\xi}^L + \boldsymbol{\xi}^U)$ and $\mathbf{E}_r = \frac{1}{2}\text{diag}(\boldsymbol{\xi}^U - \boldsymbol{\xi}^L)$, an equivalent CG-rep of Z with center $\hat{\mathbf{c}}$ is given by

$$Z = \left\{ \left[\mathbf{G}\mathbf{E}_r \quad \mathbf{0} \right], \hat{\mathbf{c}}, \begin{bmatrix} \mathbf{A}\mathbf{E}_r & \mathbf{0} \\ \mathbf{0} & \mathbf{A} \\ \mathbf{G}\mathbf{E}_r & -\mathbf{G} \end{bmatrix}, \begin{bmatrix} \mathbf{b} - \mathbf{A}\boldsymbol{\xi}_m \\ \mathbf{b} \\ \mathbf{c} - \hat{\mathbf{c}} \end{bmatrix} \right\}. \quad (4.17)$$

Proof. It is first shown that Z is contained in the set

$$\bar{Z} = \{\mathbf{G}\mathbf{E}_r, \mathbf{c} + \mathbf{G}\boldsymbol{\xi}_m, \mathbf{A}\mathbf{E}_r, \mathbf{b} - \mathbf{A}\boldsymbol{\xi}_m\}.$$

Choose any $\mathbf{z} \in Z$. There must exist $\boldsymbol{\xi} \in B_\infty(\mathbf{A}, \mathbf{b})$ such that $\mathbf{z} = \mathbf{c} + \mathbf{G}\boldsymbol{\xi}$. Since $\boldsymbol{\xi} \in [\boldsymbol{\xi}^L, \boldsymbol{\xi}^U]$, there must exist $\boldsymbol{\delta} \in B_\infty^{n_g}$ such that $\boldsymbol{\xi} = \boldsymbol{\xi}_m + \mathbf{E}_r\boldsymbol{\delta}$. Thus,

$$\begin{aligned}\mathbf{z} \in Z & \implies \exists \boldsymbol{\delta} \in B_\infty^{n_g} : \mathbf{z} = \mathbf{c} + \mathbf{G}(\boldsymbol{\xi}_m + \mathbf{E}_r\boldsymbol{\delta}), \\ & \mathbf{A}(\boldsymbol{\xi}_m + \mathbf{E}_r\boldsymbol{\delta}) = \mathbf{b} \implies \mathbf{z} \in \bar{Z}.\end{aligned}$$

Therefore, $Z \subseteq \bar{Z}$ and it is true that $Z = \bar{Z} \cap Z$. Since $\boldsymbol{\xi}^L$ and $\boldsymbol{\xi}^U$ satisfy $\mathbf{c} + \frac{1}{2}\mathbf{G}(\boldsymbol{\xi}^L + \boldsymbol{\xi}^U) = \mathbf{c} + \mathbf{G}\boldsymbol{\xi}_m = \hat{\mathbf{c}}$, then representing $\bar{Z} \cap Z$ as in (3.10) gives (4.17). \blacksquare

Remark 4.12. The linear program in Proposition 4.4 does not require $[\boldsymbol{\xi}^L, \boldsymbol{\xi}^U] \subseteq B_\infty^{n_g}$. Therefore, the midpoint $\frac{1}{2}(\boldsymbol{\xi}^L + \boldsymbol{\xi}^U)$ can assume any desired value, and it is always possible to satisfy $\mathbf{c} + \frac{1}{2}\mathbf{G}(\boldsymbol{\xi}^L + \boldsymbol{\xi}^U) = \hat{\mathbf{c}}$ if $\hat{\mathbf{c}}$ is in the range of \mathbf{G} . Thus, the linear program is always

feasible.

Remark 4.13. The optimization problems in this doctoral thesis can be readily rewritten as standard form programs by using additional decision variables and constraints (Boyd & Vandenberghe, 2004).

Therefore, based on Corollary 4.2, the proposed alternatives for use in Theorem 4.3 are summarized as follows:

C3) $\gamma_z = (\gamma_x, \gamma_w)$ is given by the closest point in $X \times W$ to the center of $X \times W$, computed through Proposition 4.3;

C4) $\gamma_z = (\gamma_x, \gamma_w)$ is the center of $X \times W$, if it satisfies $\gamma_z \in X \times W$. Otherwise, $\bar{\gamma}_z \in X \times W$ is chosen as the center of the interval hull of $X \times W$, the center of $X \times W$ is moved to $\bar{\gamma}_z$ using Proposition 4.4, and γ_z is given by $\gamma_z \triangleq \bar{\gamma}_z$.⁶

Remark 4.14. Using heuristic C4, the new center is guaranteed to belong to $X \times W$. Therefore, the latter can be chosen as approximation point, leading to a less conservative enclosure. However, since Proposition 4.4 provides a substantially more complex enclosure than the original set, given by (4.17), C4 results in an increased computational burden with respect to C3.

4.3 Update step

As described in the previous section, an enclosure for the prediction step (4.2) can be obtained in CG-rep using either the CZIB approach (Algorithm 4.5), the mean value extension (Theorem 4.2), or the first-order Taylor extension (Theorem 4.3). Consequently, taking into account the linearity of the measurement in (4.1), an exact bound for the update step (4.3) can be obtained efficiently by computing the generalized intersection of two constrained zonotopes, as explained below.

Given the prediction set $\bar{X}_k \subset \mathbb{R}^n$, the constrained zonotope $V \subset \mathbb{R}^{n_v}$ describing bounds on measurement errors, the current input $\mathbf{u}_k \in \mathbb{R}^{n_u}$, and the measurement $\mathbf{y}_k \in \mathbb{R}^{n_y}$, an exact enclosure for the update step is obtained using the definition (3.3), given by

$$\hat{X}_k = \bar{X}_k \cap_{\mathbf{C}} ((\mathbf{y}_k - \mathbf{D}_u \mathbf{u}_k) \oplus (-\mathbf{D}_v V)). \quad (4.18)$$

It is well known that the intersection in (4.18) can not be computed exactly using zonotopes, and must be over-approximated (Le et al., 2013; Alamo et al., 2005). As a consequence, the enclosures of the system states obtained after many iterations of prediction

⁶Note that this choice may lead to the same value of γ_z provided by C1, but $X \times W$ is described by a different CG-rep with center γ_z . If this point is not in $X \times W$, $\bar{\gamma}_z$ can be chosen as the point obtained from Proposition 4.3 instead.

and update may be quite conservative using zonotopes. However, with constrained zonotopes all operations in (4.18) are easily computed through (3.8)–(3.10). These lead to an enclosure with $n_g + n_{g_v}$ generators, and $n_c + n_{c_v} + n_y$ constraints, where n_g and n_c are the number of generators and constraints of \bar{X}_k , respectively.

Remark 4.15. Unlike the CZIB approach, iterated computations of the extensions proposed in Theorems 4.2 and 4.3, and the update step (4.18) result in at most a linear and quadratic increase in the complexity of the CG-rep (3.7), respectively. As with zonotopes, this complexity increase can be efficiently addressed using the complexity reduction methods discussed in Section 3.3, that outer-approximate a constrained zonotope by another with lower complexity with reasonable conservativeness.

4.4 Complexity analysis

Table 4.1 shows the computational complexity⁷ of the basic operations used in the constrained zonotope methods. The complexities of basic operations using zonotopes are also presented for comparison. These complexities assume generic inputs with dimensions $\mathbf{R} \in \mathbb{R}^{n_r \times n}$ in (3.1); $Y = \{\mathbf{G}_y, \mathbf{c}_y, \mathbf{A}_y, \mathbf{b}_y\}$ in (3.3), with $\mathbf{G}_y \in \mathbb{R}^{n_r \times n_{g_r}}$, $\mathbf{c}_y \in \mathbb{R}^{n_r}$, $\mathbf{A}_y \in \mathbb{R}^{n_{c_r} \times n_{g_r}}$, and $\mathbf{b}_y \in \mathbb{R}^{n_{c_r}}$; $\mathbf{J} \in \mathbb{R}^{n_s \times n}$ in Theorem 4.1; $X = \{\mathbf{G}_x, \mathbf{c}_x\}$ or $X = \{\mathbf{G}_x, \mathbf{c}_x, \mathbf{A}_x, \mathbf{b}_x\}$, with $\mathbf{G}_x \in \mathbb{R}^{n \times n_g}$, $\mathbf{c}_x \in \mathbb{R}^n$, $\mathbf{A}_x \in \mathbb{R}^{n_c \times n_g}$, and $\mathbf{b}_x \in \mathbb{R}^{n_c}$; k_g and k_c are the number of generators and constraints removed in the order reduction process, respectively. ‘Set inclusion’ refers to the zonotope inclusion operator in Alamo et al. (2005) for zonotopes and the CZ-inclusion operator (Theorem 4.1) for constrained zonotopes. ‘Closest point’ and ‘Change center’ correspond to Propositions 4.3 and 4.4, respectively, which are LPs. For the latter, the bounds $\tilde{\xi}^L, \tilde{\xi}^U$ are obtained using Algorithm 1 in Scott et al. (2016). Note that the interval hull of zonotopes does not require the solution of LPs (see Remark 3 in Kühn (1998)). In addition, we consider that each LP is solved at least with the performance of the simplex method presented in Kelner & Spielman (2006), which is $O(N_d N_c^3)$ with N_d and N_c the number of decision variables and constraints, respectively. Note that these numbers can be inferred for each respective LP directly from Table 4.1.

Table 4.1: Computational complexity $O(\cdot)$ of basic operations.

Operation	Zonotopes	Constrained zonotopes
Linear mapping	$nn_g n_r$	$nn_g n_r$
Minkowski sum	n	n
Generalized intersection	–	$nn_g n_r + n_r n_{g_r}$
Interval hull	nn_g	$nn_g (n_g + n_c)^3$
Set inclusion	nn_g	$nn_s n_g + n_c (n_g + n_c)^3 + nn_g^2 n_c$
Closest point	–	$(n + n_g)(n + n_g + n_c)^3$
Change center	–	$n_g (n + n_g)^3 + n_g^2 n_c$
Generator reduction	$n^2 n_g + k_g n n_g$	$(n + n_c)^2 n_g + k_g (n + n_c) n_g$
Constraint elimination	–	$k_c (n_g + n_c)^3 + k_c n n_g^2$

⁷We use the standard $O(\cdot)$ notation defined in Cormen et al. (2009).

Table 4.2 shows the computational complexity of the mean value extension and first-order Taylor extension, for the prediction and update steps, as well as complexity reduction to the same number of generators and constraints of the set prior to prediction. Specifically, we use the mean value extension (Theorem 4.2) and the first-order Taylor extension (Theorem 4.3) for the prediction steps, while the update steps are both given by the generalized intersection (4.18). These methods are denoted by CZMV and CZFO, respectively. The computational complexities of their zonotope counterparts are also presented for comparison, denoted analogously by ZMV and ZFO, which use the mean value approach in Alamo et al. (2005) and the first-order Taylor approach in Combastel (2005), respectively, for the prediction step. The update algorithm proposed in Bravo et al. (2006) is used for both ZMV and ZFO because it provided the best trade-off between accuracy and efficiency in our numerical experiments with zonotopes. Complexity reduction is applied after the update step in all four methods using the reduction methods in Scott et al. (2016) for constrained zonotopes and Method 4 in Yang & Scott (2018) for zonotopes. For constrained zonotopes, constraint elimination is performed prior to generator reduction. The complexities in Table 4.2 take into account the growth of the number of generators and constraints after each step (see Remarks 4.7 and 4.11). The dimensions in Table 4.2 are specified by the definitions $\hat{X}_{k-1} = \{\mathbf{G}_x, \mathbf{c}_x\}$, $W = \{\mathbf{G}_w, \mathbf{c}_w\}$, and $V = \{\mathbf{G}_v, \mathbf{c}_v\}$ or $\hat{X}_{k-1} = \{\mathbf{G}_x, \mathbf{c}_x, \mathbf{A}_x, \mathbf{b}_x\}$, $W = \{\mathbf{G}_w, \mathbf{c}_w, \mathbf{A}_w, \mathbf{b}_w\}$, and $V = \{\mathbf{G}_v, \mathbf{c}_v, \mathbf{A}_v, \mathbf{b}_v\}$ with $\mathbf{G}_x \in \mathbb{R}^{n \times n_g}$, $\mathbf{c}_x \in \mathbb{R}^n$, $\mathbf{A}_x \in \mathbb{R}^{n_c \times n_g}$, $\mathbf{b}_x \in \mathbb{R}^{n_c}$, $\mathbf{G}_w \in \mathbb{R}^{n_w \times n_{g_w}}$, $\mathbf{c}_w \in \mathbb{R}^{n_w}$, $\mathbf{A}_w \in \mathbb{R}^{n_{c_w} \times n_{g_w}}$, $\mathbf{b}_w \in \mathbb{R}^{n_{c_w}}$, $\mathbf{G}_v \in \mathbb{R}^{n_v \times n_{g_v}}$, $\mathbf{c}_v \in \mathbb{R}^{n_v}$, $\mathbf{A}_v \in \mathbb{R}^{n_{c_v} \times n_{g_v}}$, $\mathbf{b}_v \in \mathbb{R}^{n_{c_v}}$, $\mathbf{u}_k \in \mathbb{R}^{n_u}$, and $\mathbf{y}_k \in \mathbb{R}^{n_y}$. For simplicity, we define $m = n + n_w$, $m_g = n_g + n_{g_w}$, $m_c = n_c + n_{c_w}$, $\delta_n = n - n_y$, $\delta_w = n_{g_w} - n_{c_w}$, $\delta_v = n_{g_v} - n_{c_v}$, and $\tilde{\delta} = m_g^2 - m_c^2$. Moreover, we consider that scalar real function and scalar inclusion function evaluations have complexity $O(1)$. These correspond to evaluations of the nonlinear dynamics in (4.1) and its derivatives using real and interval arithmetic, respectively. For a detailed derivation of the computational complexities in Tables 4.1 and 4.2, see Appendix A.

The dominant terms in the prediction step of CZMV come from the computation of the interval hulls of X and W and the CZ-inclusion operators $\triangleleft(\mathbf{J}_x, X - \gamma_x)$ and $\triangleleft(\mathbf{J}_w, W - \gamma_w)$ in Theorem 4.2 and Remark 4.6. In the case of CZFO, the dominant terms come from the computation of the interval matrices $\tilde{\mathbf{Q}}^{[q]}$, the interval hull of $Z = X \times W$, and the CZ-inclusion operator $\triangleleft(\mathbf{L}, (\mathbf{c} - \gamma_z) \oplus 2\mathbf{G}B_\infty(\mathbf{A}, \mathbf{b}))$ in Theorem 4.3. Note that the worst-case complexities of the prediction steps of the proposed algorithms are higher than the zonotope methods, while the update steps are cheaper due to the generalized intersection (4.18). Even so, the complexity of the proposed methods are still polynomial. For a simplified analysis, assuming that all of the variables in Table 4.2 increase linearly with n , the total complexities for ZMV, ZFO, CZMV and CZFO are $O(n^4)$, $O(n^5)$, $O(n^5)$, and $O(n^8)$, respectively. On the other hand, even basic polytope operations are known to be exponential (Hagemann, 2015). Besides, despite the higher complexities of CZMV and

Table 4.2: Computational complexity $O(\cdot)$ of the state estimators.

Step	ZMV
Prediction	$n^2 n_g + n n_w n_{g_w}$
Update	$n_y (n^3 (m_g + n) + n^2 (m_g + n)^2 + n_u + n_v n_{g_v})$
Reduction	$n^2 (m_g + n) + n (n_{g_w} + n) (m_g + n)$
Step	ZFO
Prediction	$n (m^2 m_g + m m_g^2)$
Update	$n_y (n^3 (m_g^2 + n) + n^2 (m_g^2 + n)^2 + n_u + n_v n_{g_v})$
Reduction	$n^2 (m_g^2 + n) + n (m_g^2 + n)^2$
Step	CZMV
Prediction	$n (m^2 m_g + m m_g^2) + (m m_g + m_c) (m_g + m_c)^3$
Update	$n_y n (m_g^2 + n) + n_y n_u + n_y n_v n_{g_v}$
Reduction	$(n_{c_w} + n_{c_v} + n_y) (m_g + m_c + n_{g_v} + n_{c_v} + n + n_y)^3$ $+ (n + n_c)^2 (n_g + \delta_n + \delta_w + \delta_v) + (n + n_c) (\delta_n + \delta_w + \delta_v) (n_g + \delta_n + \delta_w + \delta_v)$
Step	CZFO
Prediction	$n (m^2 m_g + m m_g^2) + (m m_g + m_c) (m_g + m_c)^3$
Update	$n_y n (m_g^2 + n) + n_y n_u + n_y n_v n_{g_v}$
Reduction	$(m_c^2 + n_{c_v} + n_y) (m_g^2 + m_c^2 + n_{g_v} + n_{c_v} + n + n_y)^3$ $+ (n + n_c)^2 (\delta + \delta_n + \delta_v) + (n + n_c) (\delta + \delta_n + \delta_v)^2$

CZFO in comparison to the zonotope methods, they provide more accurate enclosures as shown in the next section.

4.5 Numerical examples

4.5.1 Example 1

This section presents a numerical example that demonstrates the performance of the CZIB approach.

We address the second example presented in [Alamo et al. \(2005\)](#). Consider a isothermal gas-phase reactor, charged with an initial amount of species A and B . The species are allowed to react according to the reversible reaction $2A \rightleftharpoons B$. The system states are defined as the partial pressure of each species in the reactor. The objective is to estimate all the system states while measuring the total pressure of the vessel as the reaction proceeds. The dynamic equations of the system are given by (discretized through Euler's method⁸)

$$x_{1,k} = x_{1,k-1} + T_s (-2k_1 x_{1,k-1}^2 + 2k_2 x_{2,k-1}) + k_3 w_{1,k-1},$$

$$x_{2,k} = x_{2,k-1} + T_s (k_1 x_{1,k-1}^2 - k_2 x_{2,k-1}) + k_3 w_{2,k-1},$$

where $k_1 = 0.16/60 \text{ s}^{-1} \text{ atm}^{-1}$, $k_2 = 0.0064/60 \text{ s}^{-1}$, $k_3 = 0.0001$ and $T_s = 6 \text{ s}$ ([Alamo et al., 2005](#)). The measured output is given by $y_k = [1 \ 1] \mathbf{x}_k + v_k$. The existing disturbances are bounded by $\|\mathbf{w}_k\|_\infty \leq 1$, and $\|v_k\|_\infty \leq 0.3$. The initial states are assumed to belong to the

⁸All the simulations in this doctoral thesis are executed using discrete-time models. Therefore, the discretization errors are not present in the simulations and are out of the scope of the thesis.

constrained zonotope

$$\bar{X}_0 \triangleq \left\{ \begin{bmatrix} 2.5 & 0 & 1.0 \\ 0 & 0.5 & 1.0 \end{bmatrix}, \begin{bmatrix} 2.5 \\ 1.0 \end{bmatrix}, [1 \ 0 \ 1], 1 \right\}.$$

In the simulation, we use 12-partitions in Algorithm 4.1 and the natural inclusion function (Moore et al., 2009) in Algorithm 4.2. A simulation using 3-partitions is also conducted to evaluate the resulting conservatism. The inclusion functions were computed using INTLAB (Rump, 1999). The conversion from V-rep to H-rep in Algorithm 4.3 was performed using the MPT toolbox (Herceg et al., 2013). Moreover, we perform an initial correction step $\hat{X}_0 = \bar{X}_0 \cap_{\mathcal{C}} Y_0$ to obtain tighter initial bounds, where $Y_0 = \mathbf{y}_0 \oplus (-\mathbf{D}_v V_0)$. For comparison, a third simulation is conducted using the algorithm proposed in Combastel (2005), denoted by ZComb, which is based on first-order Taylor approximation using zonotopes for the prediction step, and update step based on singular-value decomposition (SVD), with complexity limited to 20 generators using the reduction algorithm described in Method 3.1 in Chapter 3. The initial state is $\mathbf{x}_0 = [4.9 \ 1.5]^T$.

Figure 4.3 shows the time evolution of the system states, and the estimated bounds provided by the CZIB approach⁹ and ZComb. Although reasonable for x_1 , the bounds obtained using the latter are substantially conservative for x_2 . Besides, the bounds obtained using the 12-partition CZIB are significantly less conservative in all cases, demonstrating the attained reduction in overestimation using the proposed method. In order to overcome the “simplex problem” (see Remark 4.3), a Minkowski sum was performed using a zero-centered interval vector with 10% of the total diameter of the interval bundle, at every time step it occurred. In addition, the CG-rep of the constrained zonotope \hat{X}_{40} in the 12-partition simulation was given by 13 generators and 9 constraints, without requiring the use of complexity reduction algorithms. The average execution times for the 3-partition and 12-partition simulations were 0.0855 and 0.6013 seconds per iteration, respectively, in MATLAB 9.1 and CPLEX 12.8, with 8GB RAM and Intel Core i7 4510U 3.1 GHz processor, using only sequential computation.

4.5.2 Example 2

This section presents numerical results for the two new set-based state estimation methods enabled by the new extensions proposed in Sections 4.2.2 and 4.2.3. Specifically, we use the mean value extension (Theorem 4.2) and the first-order Taylor extension (Theorem 4.3) for the prediction steps, while the update step is given by the exact intersection (4.18) for both methods. These methods are denoted by CZMV and CZFO, respectively. The results are compared against two zonotopic methods denoted analogously by ZMV and ZFO, which use the mean value approach in Alamo et al. (2005) and the first-order

⁹The plotted curves are the bounds of the interval hull of \hat{X}_k .

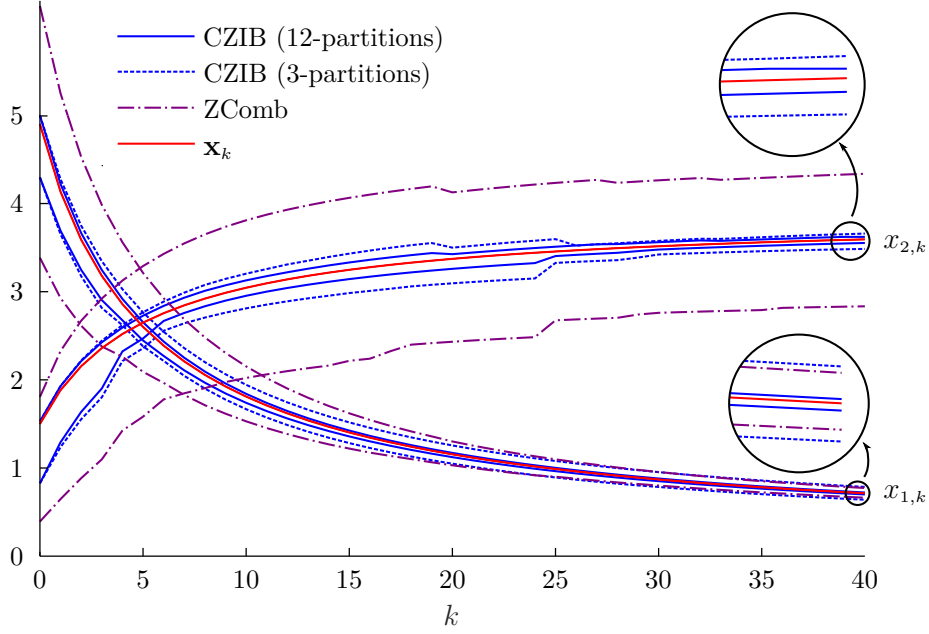


Figure 4.3: Time evolution of the estimated bounds and the real states.

Taylor approach in [Combastel \(2005\)](#), respectively, for the prediction step. The update algorithm proposed in [Bravo et al. \(2006\)](#) is used for both ZMV and ZFO because it provided the best trade-off between accuracy and efficiency in our numerical experiments with zonotopes. Complexity reduction is applied after the update step in all four methods using Methods [3.5](#) and [3.4](#) for constrained zonotopes and Method [3.2](#) for zonotopes. The imposed complexity limits are described separately for each example below.

To demonstrate the effect of the different choices of the approximation point γ_x , we first analyze one iteration of the prediction step for the nonlinear system ([Raimondo et al., 2012](#))

$$\begin{aligned} x_{1,k} &= 3x_{1,k-1} - \frac{x_{1,k-1}^2}{7} - \frac{4x_{1,k-1}x_{2,k-1}}{4 + x_{1,k-1}} + w_{1,k-1}, \\ x_{2,k} &= -2x_{2,k-1} + \frac{3x_{1,k-1}x_{2,k-1}}{4 + x_{1,k-1}} + w_{2,k-1}, \end{aligned} \quad (4.19)$$

with

$$X_0 = \left\{ \left[\begin{array}{ccc} 0.2 & 0.4 & 0.2 \\ 0.2 & 0 & -0.2 \end{array} \right], \left[\begin{array}{c} -1 \\ 1 \end{array} \right], [2 \ 2 \ 2], -3 \right\}, \quad (4.20)$$

where $w_k \in \mathbb{R}^2$ denotes process uncertainties, which are zero in this first scenario.

Figure [4.4](#) shows the constrained zonotope X_0 and the enclosures of the one-step reachable set obtained by Theorem [4.2](#) using $C1$ – $C4$. Since the complexity of the enclosure for $C4$ is higher than for the other methods (see Proposition [4.4](#)), the reduction methods in [Scott et al. \(2016\)](#) were used to reduce the number of generators and constraints in this enclosure to match the other methods before comparison. In this example, the choice of γ_x has a moderate impact in the enclosure computed by Theorem [4.2](#), with $C2$ providing the least conservative result, as expected. Therefore, $C2$ is employed in Theorem [4.2](#)

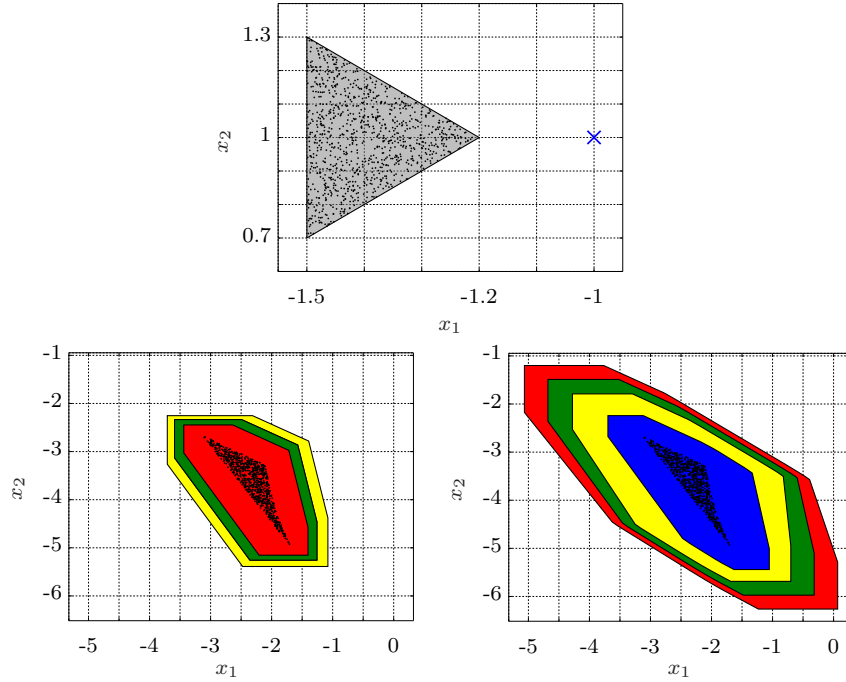


Figure 4.4: Top: The constrained zonotope X_0 with ‘ \times ’ denoting its center. Left: Enclosures obtained by applying Theorem 4.2 to (4.19). Right: Enclosures obtained by applying Theorem 4.3 to (4.19). The real vector γ_x is determined by $C1$ (green), $C2$ (red), $C3$ (yellow), and $C4$ (blue). For the mean value extension (left), $C4$ is overlapped with $C1$. Black dots denote uniform samples from X_0 propagated through (4.19).

henceforth.

Figure 4.4 also shows the enclosures of the one-step reachable set obtained by Theorem 4.3 with $C1$ – $C4$. Clearly, the enclosures produced by Theorem 4.3 are strongly affected by the choice of γ_x , with $C4$ providing the least conservative result. In addition, note that the enclosures provided by the first-order Taylor extension are more conservative than those obtained by the mean value extension. However, experience with zonotopes and intervals (see Raimondo et al. (2012) for detailed examples) suggests that the relative merits of these two methods will depend on the dynamics of the system, as well as the shape and size of the set X_0 , and the maximum allowed number of generators and constraints. This is corroborated by the next results.

We consider now the linear measurement equation

$$\begin{bmatrix} y_{1,k} \\ y_{2,k} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} x_{1,k} \\ x_{2,k} \end{bmatrix} + \begin{bmatrix} v_{1,k} \\ v_{2,k} \end{bmatrix}, \quad (4.21)$$

with bounds $\|\mathbf{w}_k\|_\infty \leq 0.4$ and $\|\mathbf{v}_k\|_\infty \leq 0.4$, where $\mathbf{v}_k \in \mathbb{R}^2$ denotes measurement uncertain-

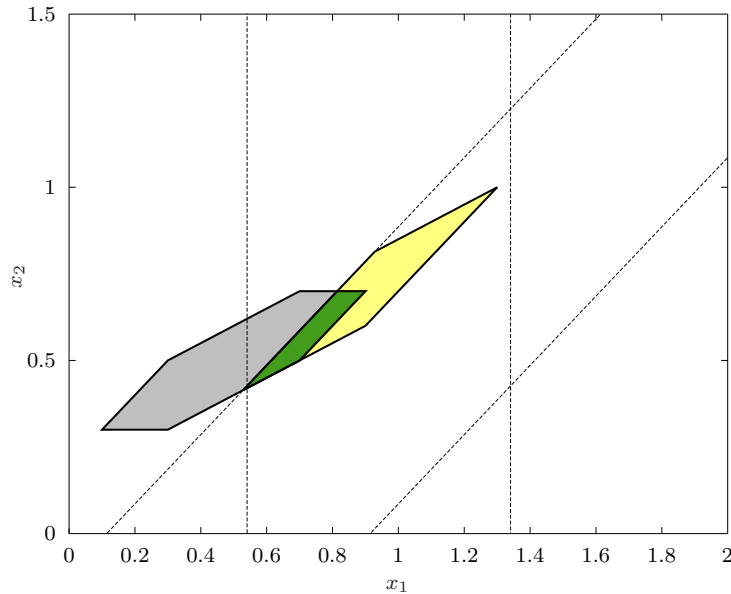


Figure 4.5: The initial set X_0 (gray), the initial bounded uncertain measurements (dashed lines), the intersection computed as in Bravo et al. (2006) (yellow), and the constrained zonotope (green) computed by (4.18).

ties. The initial states \mathbf{x}_0 are bounded by the zonotope¹⁰

$$X_0 = \left\{ \left[\begin{array}{ccc} 0.1 & 0.2 & -0.1 \\ 0.1 & 0.1 & 0 \end{array} \right], \left[\begin{array}{c} 0.5 \\ 0.5 \end{array} \right] \right\}. \quad (4.22)$$

To generate process measurements, system (4.19) was simulated with $\mathbf{x}_0 = (0.8, 0.65) \in X_0$ and process and measurement uncertainties drawn from uniform random distributions. The number of generators and constraints of the constrained zonotopes was limited to 20 and 5, respectively, while the number of generators of the zonotopes was limited to 20. Figure 4.5 shows the results of the initial update step using the intersection algorithm in Bravo et al. (2006) and the generalized intersection (4.18) computed using (3.10), which yields a constrained zonotope. Clearly, since the generalized intersection is not a symmetric set, it cannot be described by a zonotope. In contrast, the resulting constrained zonotope corresponds to the exact intersection, providing far less conservative bounds in the first update step.

Figure 4.6 shows the first four time steps of CZMV with γ_x given by $C2$ in a scenario without process uncertainties ($\mathbf{w}_k = \mathbf{0}$). For comparison purposes, the zonotopes computed using ZMV are also depicted. CZMV provides much less conservative enclosures than ZMV for this example, demonstrating the effectiveness of the proposed nonlinear state estimation strategy.

Figure 4.7 shows the radii (half the length of the longest edge of the interval hull) of the sets provided by CZMV using $C2$ and ZMV over 100 time steps considering process

¹⁰Note that X_0 , W and V are expressed as zonotopes for a fair comparison with the zonotope methods.

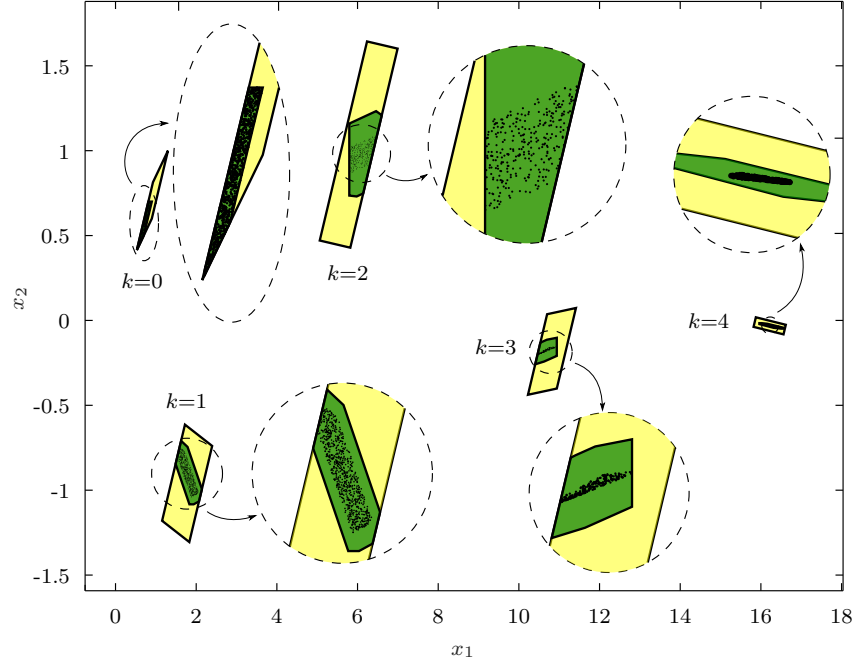


Figure 4.6: Results from the first four time steps of set-valued state estimation (after update) using the constrained zonotopic method CZMV (green) and the zonotopic method ZMV (yellow). Black dots denote uniform samples from X_0 propagated through (4.19) that are consistent with the current measurement.

disturbances, and also compares the radii of the update sets computed by ZFO and CZFO with (γ_x, γ_w) given by C_4 . Since (4.19) is affine in w_k , the enclosure $Z \supseteq \mu(\gamma_x, W)$ in Theorem 4.2 was computed as described at the end of Remark 4.6. CZMV provided less conservative bounds than the zonotopes computed by ZMV, with a CZMV-to-ZMV average radius ratio (ARR, i.e., the ratio of the radius of the CZMV set at k over the ZMV set at k averaged over all time steps k) of only 51.4%. In addition, as in the previous case, CZFO provides less conservative bounds than ZFO, with the CZFO-to-ZFO ARR being only 53.66%. The size of the sets provided by CZMV and CZFO were quite similar, with CZFO being less conservative (the CZFO-to-CZMV ARR was 98.75%). The ARR for different numbers of constraints are shown in Table 4.3, with the average computed considering in addition simulations with different numbers of generators. Execution times are shown in Table 4.4. These were obtained using MATLAB 9.1 with CPLEX 12.8 and INTLAB 9, in a laptop with 8GB RAM and an Intel Core i7 4510U 3.1 GHz processor.

The use of constrained zonotopes in CZMV and CZFO results in sets that are slightly more complex than those generated by ZMV and ZFO (specifically, the set description involves five equality constraints that are not present in the zonotopes from ZMV and ZFO). However, this example shows that this increase in complexity is compensated by greatly improved accuracy.

A third example, consisting of a realistic scenario of nonlinear set-based state estimation of a quadrotor UAV using the mean value and first-order Taylor extensions proposed in

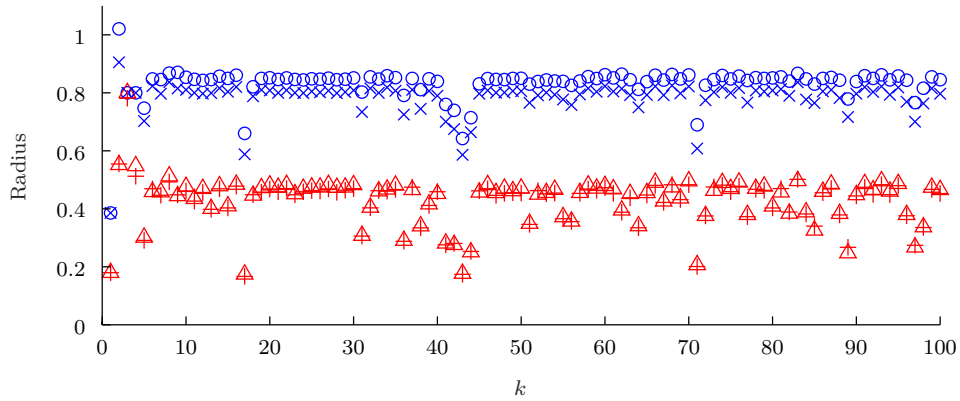


Figure 4.7: Radii of the update sets obtained by applying ZMV (\circ), ZFO (\times), CZMV (\triangle), and CZFO ($+$) to (4.19) with process disturbances.

Table 4.3: Average radius ratio of the estimators with varying numbers of constraints. Each average is taken over 3 separate simulations using $n_g \in \{8, 12, 20\}$.

n_c	CZMV/ZMV	CZFO/ZFO	CZFO/CZMV
1	54.1%	59.8%	104.5%
3	51.6%	54.0%	99.0%
5	51.6%	53.7%	98.5%

Table 4.4: Average total times per iteration of the estimators with varying numbers of constraints. Each average is taken over 30 separate simulations using $n_g \in \{8, 12, 20\}$. Times spent only on complexity reduction are shown in parenthesis.

n_c	ZMV	ZFO	CZMV	CZFO
0	5.79 (0.24) ms	14.03 (2.93) ms	–	–
1	–	–	19.2 (1.6) ms	76.4 (57.1) ms
3	–	–	20.7 (1.8) ms	4.28 (4.26) s
5	–	–	22.5 (1.9) ms	8.93 (8.91) s

this chapter, can be found in Chapter 9.

4.6 Final remarks

This chapter developed new set-based state estimation methods based on constrained zonotopes. Three new nonlinear set-based state estimation methods were developed, specifically the CZIB approach, the mean value extension, and the first-order Taylor extension using constrained zonotopes. The measurements were effectively incorporated in the state estimation methods by means of generalized intersection of constrained zonotopes considering linear measurement.

The advantages of the new nonlinear state estimation methods were corroborated by numerical experiments. The first example demonstrated the reduced conservativeness of the CZIB approach in comparison to the zonotope-based method by [Combastel \(2005\)](#). The second example demonstrated the accuracy of the proposed extensions for nonlinear

set-based state estimation using constrained zonotopes. In each experiment, the new methods were compared with existing zonotope-based approaches.

The next chapter develops new methods for set-based state estimation of systems with nonlinear measurement and invariants. These are a generalization of the mean value and first-order Taylor extensions introduced in this chapter.

5

State estimation of systems with nonlinear
measurement and invariants

This chapter presents new methods for set-valued state estimation of discrete-time nonlinear systems whose trajectories are known to satisfy nonlinear equality constraints, called *invariants* (e.g., conservation laws). The new methods based on constrained zonotopes improve the standard prediction-update framework for systems with invariants by adding a *consistency step*. This new step uses invariants to reduce conservatism and is enabled by new algorithms for refining constrained zonotopes based on nonlinear constraints. This chapter also presents significant improvements to prediction and update steps proposed in Chapter 4. Specifically, new update algorithms are developed that allow nonlinear measurement equations for the first time, and existing prediction methods based on conservative approximation techniques are modified to allow a more flexible choice of the approximation point, which can lead to tighter enclosures. Numerical results demonstrate that the resulting methods can provide significantly tighter enclosures than existing zonotope-based methods while maintaining comparable efficiency. The content presented in this chapter was published in *Automatica* ([Rego et al., 2021d](#)).

This chapter is organized as follows. The set-based state estimation problem and the class of nonlinear systems considered are described in Section 5.1. The main results are given in Section 5.2, including the new consistency and update algorithms and the improvements to the prediction algorithms developed in Chapter 4. Numerical examples are presented in Section 5.4, and Section 5.5 concludes the chapter.

5.1 Problem formulation

Consider a class of nonlinear discrete-time systems described by

$$\begin{aligned}\mathbf{x}_k &= \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}), \\ \mathbf{y}_k &= \mathbf{g}(\mathbf{x}_k, \mathbf{u}_k, \mathbf{v}_k),\end{aligned}\tag{5.1}$$

for $k \geq 1$, with $\mathbf{y}_0 = \mathbf{g}(\mathbf{x}_0, \mathbf{u}_0, \mathbf{v}_0)$, where $\mathbf{x}_k \in \mathbb{R}^n$ denotes the system state, $\mathbf{u}_k \in \mathbb{R}^{n_u}$ is the known input, $\mathbf{w}_k \in \mathbb{R}^{n_w}$ is the process uncertainty, $\mathbf{y}_k \in \mathbb{R}^{n_y}$ is the measured output, and $\mathbf{v}_k \in \mathbb{R}^{n_v}$ is the measurement uncertainty. The nonlinear mappings $\mathbf{f} : \mathbb{R}^n \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_w} \rightarrow \mathbb{R}^n$ and $\mathbf{g} : \mathbb{R}^n \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_v} \rightarrow \mathbb{R}^{n_y}$ are assumed to be of class \mathcal{C}^2 . The initial condition and uncertainties are assumed to be unknown-but-bounded, i.e., $\mathbf{x}_0 \in \bar{X}_0$, $\mathbf{w}_k \in W$, and $\mathbf{v}_k \in V$, where \bar{X}_0 , W , and V are known convex polytopic sets.

This chapter presents an improved set-valued state estimation method for systems satisfying known invariants, as defined in the following assumption.

Assumption 5.1. There exists a \mathcal{C}^2 function $\mathbf{h} : \mathbb{R}^n \rightarrow \mathbb{R}^{n_h}$ such that, for every solution of (5.1) with $\mathbf{x}_0 \in \bar{X}_0$, $\mathbf{w}_k \in W$, and $\mathbf{v}_k \in V$,

$$\mathbf{h}(\mathbf{x}_0) = \mathbf{0} \quad \implies \quad \mathbf{h}(\mathbf{x}_k) = \mathbf{0}, \quad \forall k \geq 0.\tag{5.2}$$

We refer to the components of \mathbf{h} as *invariants*.

Remark 5.1. A sufficient condition for (5.2) is that $\mathbf{h}(\mathbf{f}(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k)) = \mathbf{0}$ for all \mathbf{x}_k such that $\mathbf{h}(\mathbf{x}_k) = \mathbf{0}$, for all $\mathbf{w}_k \in W$ and \mathbf{u}_k with $k \geq 0$.

Many systems of practical interest obey invariants describing, e.g., material conservation laws in chemical systems, conservation of energy or momentum in mechanical systems, or the isometry inherent to orientation dynamics in aerospace and robotic systems (Shen & Scott, 2017; Goodarzi & Lee, 2017; Rotella et al., 2014). Prior work on nonlinear reachability analysis has shown that, if used properly, even simple physical information in the form of invariants can dramatically improve the accuracy of reachability bounds computed by interval methods (Scott et al., 2013b; Shen & Scott, 2017; Yang & Scott, 2020). Similarly, our aim here is to develop new algorithms for effectively using invariants to improve the accuracy of the state-of-the-art state estimation algorithms based on constrained zonotopes proposed in Chapter 4.

For any $k \geq 0$, let X_k denote the set of all states \mathbf{x}_k that are consistent with (i) the nonlinear model (5.1), (ii) the measured output sequence up to time k , $(\mathbf{y}_0, \dots, \mathbf{y}_k)$, and (iii) the unknown-but-bounded uncertainties $\mathbf{x}_0 \in \{\mathbf{x} \in \bar{X}_0 : \mathbf{h}(\mathbf{x}) = \mathbf{0}\}$, $\mathbf{w}_k \in W$, and $\mathbf{v}_k \in W$. Since exact characterization of X_k is generally intractable (Kühn, 1998; Platzer & Clarke, 2007), the objective of set-valued state estimation is to approximate X_k as accurately as possible by a guaranteed enclosure $\tilde{X}_k \supseteq X_k$. We accomplish this here by extending

the standard prediction-update estimation framework with a new consistency step for tightening the enclosures using invariants. The general scheme is given by the following recursion:

$$\bar{X}_k \supseteq \{\mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}) : \mathbf{x}_{k-1} \in \tilde{X}_{k-1}, \mathbf{w}_{k-1} \in W\}, \quad (5.3)$$

$$\hat{X}_k \supseteq \{\mathbf{x}_k \in \bar{X}_k : \mathbf{g}(\mathbf{x}_k, \mathbf{u}_k, \mathbf{v}_k) = \mathbf{y}_k, \mathbf{v}_k \in V\}, \quad (5.4)$$

$$\tilde{X}_k \supseteq \{\mathbf{x}_k \in \hat{X}_k : \mathbf{h}(\mathbf{x}_k) = \mathbf{0}\}, \quad (5.5)$$

where (5.3) is the *prediction step*, (5.4) is the *update step*, (5.5) is the *consistency step*, and the scheme is initialized with \bar{X}_0 in the update step. According to the definition of X_k , we have that $X_0 = \{\mathbf{x}_0 \in \bar{X}_0 : \mathbf{h}(\mathbf{x}_0) = \mathbf{0}, \mathbf{g}(\mathbf{x}_0, \mathbf{u}_0, \mathbf{v}_0) = \mathbf{y}_0, \mathbf{v}_0 \in V\}$. This immediately implies that $\tilde{X}_0 \supseteq X_0$. If \tilde{X}_{k-1} is a valid enclosure of X_{k-1} for some $k \geq 1$, then standard results in set-valued state estimation show that $\hat{X}_k \supseteq X_k$ (Chisci et al., 1996; Le et al., 2013). Since any $\mathbf{x}_{k-1} \in X_{k-1}$ emanates from some $\mathbf{x}_0 \in \bar{X}_0$ satisfying $\mathbf{h}(\mathbf{x}_0) = \mathbf{0}$ by definition, Assumption 5.1 implies that $\mathbf{h}(\mathbf{x}_k) = \mathbf{0}$, and it follows that $\tilde{X}_k \supseteq X_k$ as well. By induction, we conclude that $\tilde{X}_k \supseteq X_k$ for all $k \geq 0$ as desired.

In the remainder of the chapter, our goal is to develop methods for computing accurate enclosures for each of the three steps (5.3)–(5.5). Building on the results presented in Chapter 4, the main results of this chapter include generalizations of the prediction methods proposed in Sections 4.2.2 and 4.2.3 with improved accuracy, new update methods that are applicable to nonlinear measurement equations, and methods for the new consistency step to make effective use of invariants.

5.2 Recursive algorithm

This section presents new methods for computing enclosures for each step in the extended prediction-update-consistency algorithm (5.3)–(5.5) using constrained zonotopes. The proposed recursive scheme is summarized in Algorithm 5.1. In this algorithm, complexity reduction methods can be used after each step to limit the set complexity increase. We begin with two core lemmas required for all three steps.

Algorithm 5.1 Recursive state estimation algorithm

- 1: (Prediction step) Given the constrained zonotopes $\tilde{X}_{k-1} \times W \subset \mathbb{R}^n \times \mathbb{R}^{n_w}$, and input $\mathbf{u}_{k-1} \in \mathbb{R}^{n_u}$, compute the predicted constrained zonotope \bar{X}_k satisfying (5.3).
 - 2: (Update step) Given the constrained zonotopes $\bar{X}_k \times V \subset \mathbb{R}^n \times \mathbb{R}^{n_v}$, input $\mathbf{u}_k \in \mathbb{R}^{n_u}$, and measurement $\mathbf{y}_k \in \mathbb{R}^{n_y}$, compute a refined constrained zonotope \hat{X}_k satisfying (5.4).
 - 3: (Consistency step) Given the constrained zonotope $\hat{X}_k \subset \mathbb{R}^n$, compute a refined constrained zonotope \tilde{X}_k satisfying (5.5).
-

Lemma 5.1. Let $\alpha : \mathbb{R}^n \times \mathbb{R}^{n_w} \rightarrow \mathbb{R}^{n_\alpha}$ be of class \mathcal{C}^1 , and let $\nabla_x \alpha$ denote the gradient of α with respect to its first argument. Let $X \subset \mathbb{R}^n$ and $W \subset \mathbb{R}^{n_w}$ be constrained zonotopes, and let $\mathbf{J}_x \in \mathbb{IR}^{n_\alpha \times n}$ be an interval matrix satisfying

$$\nabla_x^T \alpha(\square X, W) \triangleq \{\nabla_x^T \alpha(\mathbf{x}, \mathbf{w}) : \mathbf{x} \in \square X, \mathbf{w} \in W\} \subseteq \mathbf{J}_x. \quad (5.6)$$

For every $\mathbf{x} \in X$, $\mathbf{w} \in W$, and $\gamma_x \in \square X$, there exists $\hat{\mathbf{J}}_x \in \mathbf{J}_x$ such that

$$\alpha(\mathbf{x}, \mathbf{w}) = \alpha(\gamma_x, \mathbf{w}) + \hat{\mathbf{J}}_x(\mathbf{x} - \gamma_x).$$

Proof. Choose any $(\mathbf{x}, \mathbf{w}) \in X \times W$. Since $\mathbf{x} \in X \subseteq \square X$ and $\gamma_x \in \square X$, the Mean Value Theorem ensures that, for any $i = 1, 2, \dots, n$, $\exists \delta^{[i]} \in \square X$ such that $\alpha_i(\mathbf{x}, \mathbf{w}) = \alpha_i(\gamma_x, \mathbf{w}) + \nabla_x^T \alpha_i(\delta^{[i]}, \mathbf{w})(\mathbf{x} - \gamma_x)$. But $\nabla_x^T \alpha_i(\delta^{[i]}, \mathbf{w})$ is contained in the i -th row of \mathbf{J}_x by hypothesis, and since this is true for all $i = 1, 2, \dots, n$, $\exists \hat{\mathbf{J}}_x \in \mathbf{J}_x$ such that $\alpha(\mathbf{x}, \mathbf{w}) = \alpha(\gamma_x, \mathbf{w}) + \hat{\mathbf{J}}_x(\mathbf{x} - \gamma_x)$. ■

Lemma 5.1 provides an exact linear representation of the nonlinear function α between two points based on the Mean Value Theorem, which is useful for computations with constrained zonotopes. This lemma is very similar to Theorem 4.2 in Chapter 4. The only difference is that Theorem 4.2 requires the approximation point γ_x to lie in X , while Lemma 5.1 allows γ_x to be chosen from the larger set $\square X$. This is important because obtaining a point in X (or testing a given point for membership) requires solving a linear program (Property 3.1), whereas obtaining a point in $\square X$ is trivial. The proof of Lemma 5.1 is given above for completeness, but it follows easily from the proof of Theorem 4.2 by replacing the condition $\gamma_x \in X$ with $\gamma_x \in \square X$ throughout.

The next lemma provides an alternative method for obtaining an exact linear representation of a nonlinear function between two points based on Taylor's Theorem. This lemma is similar to Theorem 4.3 in Chapter 4, with the difference again that the approximation point is chosen from $\square Z$ rather than Z .

Lemma 5.2. Let $\beta : \mathbb{R}^m \rightarrow \mathbb{R}^n$ be of class \mathcal{C}^2 and let $\mathbf{z} \in \mathbb{R}^m$ denote its argument. Let $Z = \{\mathbf{G}, \mathbf{c}, \mathbf{A}, \mathbf{b}\} \subset \mathbb{R}^m$ be a constrained zonotope with m_g generators and m_c constraints. For each $q = 1, 2, \dots, n$, let $\mathbf{Q}^{[q]} \in \mathbb{IR}^{m \times m}$ and $\tilde{\mathbf{Q}}^{[q]} \in \mathbb{IR}^{m_g \times m_g}$ be interval matrices satisfying $\mathbf{Q}^{[q]} \supseteq \mathbf{H} \beta_q(\square Z)$ and $\tilde{\mathbf{Q}}^{[q]} \supseteq \mathbf{G}^T \mathbf{Q}^{[q]} \mathbf{G}$. Moreover, define

$$\begin{aligned} \tilde{c}_q &= \text{trace} \left(\text{mid}(\tilde{\mathbf{Q}}^{[q]}) \right) / 2, & \tilde{\mathbf{G}}_{q,\cdot} &= \left[\cdots \underbrace{\text{mid}(\tilde{Q}_{ii}^{[q]})/2}_{\forall i} \cdots \underbrace{\left(\text{mid}(\tilde{Q}_{ij}^{[q]}) + \text{mid}(\tilde{Q}_{ji}^{[q]}) \right)}_{\forall i < j} \cdots \right], \\ \tilde{\mathbf{G}}_{\mathbf{d}} &= \text{diag}(\mathbf{d}), & d_q &= \sum_{i,j} \left| \text{rad}(\tilde{Q}_{ij}^{[q]}) \right|, & \tilde{\mathbf{A}} &= \left[\tilde{\mathbf{A}}_{\zeta} \quad \tilde{\mathbf{A}}_{\xi} \quad \mathbf{0}_{\frac{m_c}{2}(1+m_c) \times n} \right], \end{aligned}$$

$$\begin{array}{ccc}
 \tilde{\mathbf{A}}_{\mathbf{c}} = \begin{bmatrix} \vdots \\ \cdots \quad \frac{1}{2} A_{ri} A_{si} \quad \cdots \\ \vdots \end{bmatrix} & \begin{array}{c} \Big\| \\ \forall r \leq s, \\ \Big\downarrow \end{array} & \tilde{\mathbf{b}} = \begin{bmatrix} \vdots \\ b_r b_s - \frac{1}{2} \sum_i A_{ri} A_{si} \\ \vdots \end{bmatrix} \begin{array}{c} \Big\| \\ \forall r \leq s, \\ \Big\downarrow \end{array} \\
 \xrightarrow{\forall i} & & \\
 \tilde{\mathbf{A}}_{\mathbf{\xi}} = \begin{bmatrix} \vdots \\ \cdots \quad A_{ri} A_{sj} + A_{rj} A_{si} \quad \cdots \\ \vdots \end{bmatrix} & \begin{array}{c} \Big\| \\ \forall r \leq s, \\ \Big\downarrow \end{array} & \\
 \xrightarrow{\forall i < j} & &
 \end{array}$$

with indices $i, j = 1, 2, \dots, m_g$ and $r, s = 1, 2, \dots, m_c$. Finally, choose any $\gamma_z \in \square Z$ and let $\mathbf{L} \in \mathbb{IR}^{n \times m}$ be an interval matrix satisfying $\mathbf{L}_{q,:} \supseteq (\mathbf{c} - \gamma_z)^T \mathbf{Q}^{[q]}$ for all $q = 1, \dots, n$. For every $\mathbf{z} \in Z$, there exist $\boldsymbol{\xi} \in B_\infty(\mathbf{A}, \mathbf{b})$, $\tilde{\boldsymbol{\xi}} \in B_\infty(\tilde{\mathbf{A}}, \tilde{\mathbf{b}})$, and $\hat{\mathbf{L}} \in \mathbf{L}$ such that

$$\begin{aligned}
 \boldsymbol{\beta}(\mathbf{z}) &= \boldsymbol{\beta}(\gamma_z) + \nabla^T \boldsymbol{\beta}(\gamma_z)(\mathbf{z} - \gamma_z) \\
 &\quad + \tilde{\mathbf{c}} + [\tilde{\mathbf{G}} \quad \tilde{\mathbf{G}}_d] \tilde{\boldsymbol{\xi}} + \hat{\mathbf{L}}((\mathbf{c} - \gamma_z) + 2\mathbf{G}\boldsymbol{\xi}).
 \end{aligned}$$

Proof. This follows by replacing $\gamma_z \in Z$ with $\gamma_z \in \square Z$ in the proof of Theorem 4.3. \blacksquare

5.2.1 Prediction step

This section presents two different approaches for the prediction step in Algorithm 5.1. These methods are improved versions of the mean value and first-order Taylor extensions developed in Chapter 4, which allow for a more flexible choice of the approximation point enabled by Lemmas 5.1 and 5.2 above.

Proposition 5.1. Let $\mathbf{f} : \mathbb{R}^n \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_w} \rightarrow \mathbb{R}^n$ be of class \mathcal{C}^1 , and let $\nabla_x \mathbf{f}$ denote the gradient of \mathbf{f} with respect to its first argument. Let $\mathbf{u} \in \mathbb{R}^{n_u}$, and let $X \subset \mathbb{R}^n$ and $W \subset \mathbb{R}^{n_w}$ be constrained zonotopes. Choose any $\gamma_x \in \square X$. If Z_w is a constrained zonotope such that $\mathbf{f}(\gamma_x, \mathbf{u}, W) \subseteq Z_w$ and $\mathbf{J}_x \in \mathbb{IR}^{n \times n}$ is an interval matrix satisfying $\nabla_x^T \mathbf{f}(\square X, \mathbf{u}, W) \subseteq \mathbf{J}_x$, then $\mathbf{f}(X, \mathbf{u}, W) \subseteq Z_w \oplus \triangleleft(\mathbf{J}_x, X - \gamma_x)$.

Proof. Choose any $(\mathbf{x}, \mathbf{w}) \in X \times W$. Lemma 5.1 ensures that there exists a real matrix $\hat{\mathbf{J}}_x \in \mathbf{J}_x$ such that $\mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{w}) = \mathbf{f}(\gamma_x, \mathbf{u}, \mathbf{w}) + \hat{\mathbf{J}}_x(\mathbf{x} - \gamma_x)$. By Theorem 4.1 and the choice of Z_w , it follows that $\mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{w}) \in Z_w \oplus \triangleleft(\mathbf{J}_x, X - \gamma_x)$, as desired. \blacksquare

Proposition 5.2. Let $\mathbf{f} : \mathbb{R}^n \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_w} \rightarrow \mathbb{R}^n$ be of class \mathcal{C}^2 , let $\mathbf{u} \in \mathbb{R}^{n_u}$, and let $X = \{\mathbf{G}_x, \mathbf{c}_x, \mathbf{A}_x, \mathbf{b}_x\}$ and $W = \{\mathbf{G}_w, \mathbf{c}_w, \mathbf{A}_w, \mathbf{b}_w\}$ be constrained zonotopes with n_g generators and n_c constraints, and with n_{g_w} generators and n_{c_w} constraints, respectively. Denote $\mathbf{z} = (\mathbf{x}, \mathbf{w})$ and $Z = X \times W = \{\mathbf{G}, \mathbf{c}, \mathbf{A}, \mathbf{b}\} \subset \mathbb{R}^{n+n_w}$. For each $q = 1, 2, \dots, n$, let $\mathbf{Q}^{[q]} \in \mathbb{IR}^{(n+n_w) \times (n+n_w)}$ and $\tilde{\mathbf{Q}}^{[q]} \in \mathbb{IR}^{(n_g+n_{g_w}) \times (n_g+n_{g_w})}$ be interval matrices satisfying $\mathbf{Q}^{[q]} \supseteq \mathbf{H}_z f_q(\square X, \mathbf{u}, \square W)$ and $\tilde{\mathbf{Q}}^{[q]} \supseteq \mathbf{G}^T \mathbf{Q}^{[q]} \mathbf{G}$. Moreover, define $\tilde{\mathbf{c}}, \tilde{\mathbf{G}}, \tilde{\mathbf{G}}_d, \tilde{\mathbf{A}},$ and $\tilde{\mathbf{b}}$, as in Lemma 5.2. Finally, choose any

$\gamma_z = (\gamma_x, \gamma_w) \in \square Z$ and let $\mathbf{L} \in \mathbb{IR}^{n \times m}$ be an interval matrix satisfying $\mathbf{L}_{q,:} \supseteq (\mathbf{c} - \gamma_z)^T \mathbf{Q}^{[q]}$ for all $q = 1, \dots, n$. Then,

$$\mathbf{f}(X, \mathbf{u}, W) \subseteq \mathbf{f}(\gamma_x, \mathbf{u}, \gamma_w) \oplus \nabla_z^T \mathbf{f}(\gamma_x, \mathbf{u}, \gamma_w)(Z - \gamma_z) \oplus R, \quad (5.7)$$

where $R = \tilde{\mathbf{c}} \oplus [\tilde{\mathbf{G}} \ \tilde{\mathbf{G}}_d] B_\infty(\tilde{\mathbf{A}}, \tilde{\mathbf{b}}) \oplus \triangleleft(\mathbf{L}, (\mathbf{c} - \gamma_z) \oplus 2\mathbf{G}B_\infty(\mathbf{A}, \mathbf{b}))$.

Proof. Choose any $(\mathbf{x}, \mathbf{w}) = \mathbf{z} \in Z$. Lemma 5.2 ensures that there exist $\boldsymbol{\xi} \in B_\infty(\mathbf{A}, \mathbf{b})$, $\tilde{\boldsymbol{\xi}} \in B_\infty(\tilde{\mathbf{A}}, \tilde{\mathbf{b}})$, and $\hat{\mathbf{L}} \in \mathbf{L}$, such that

$$\begin{aligned} \mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{w}) &= \mathbf{f}(\gamma_x, \mathbf{u}, \gamma_w) + \nabla^T \mathbf{f}(\gamma_x, \mathbf{u}, \gamma_w)(\mathbf{z} - \gamma_z) \\ &\quad + \tilde{\mathbf{c}} + [\tilde{\mathbf{G}} \ \tilde{\mathbf{G}}_d] \tilde{\boldsymbol{\xi}} + \hat{\mathbf{L}}((\mathbf{c} - \gamma_z) + 2\mathbf{G}\boldsymbol{\xi}). \end{aligned}$$

Therefore, $\mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{w}) \in \mathbf{f}(\gamma_x, \mathbf{u}, \gamma_w) \oplus \nabla^T \mathbf{f}(\gamma_x, \mathbf{u}, \gamma_w)(Z - \gamma_z) \oplus \triangleleft(\mathbf{L}, (\mathbf{c} - \gamma_z) \oplus 2\mathbf{G}B_\infty(\mathbf{A}, \mathbf{b})) \oplus \tilde{\mathbf{c}} \oplus [\tilde{\mathbf{G}} \ \tilde{\mathbf{G}}_d] B_\infty(\tilde{\mathbf{A}}, \tilde{\mathbf{b}})$. Thus, (5.7) follows immediately from the definition of R . \blacksquare

Remark 5.2. The constrained zonotope Z_w in Proposition 5.1 can be obtained using the mean value extension $\mathbf{f}(\gamma_x, \mathbf{u}, W) \subseteq Z_w = \mathbf{f}(\gamma_x, \mathbf{u}, \gamma_w) \oplus \triangleleft(\mathbf{J}_w, W - \gamma_w)$ for a chosen point $\gamma_w \in \square W$, with \mathbf{J}_w being an interval matrix satisfying $\mathbf{J}_w \supseteq \nabla_w^T \mathbf{f}(\gamma_x, \mathbf{u}, \square W)$. In this chapter, the interval matrices \mathbf{J}_x , \mathbf{J}_w (Proposition 5.1), $\mathbf{Q}^{[q]}$, $\tilde{\mathbf{Q}}^{[q]}$, \mathbf{L} (Proposition 5.2), and similar interval matrices in Propositions 5.3 and 5.4, and Corollaries 5.1 and 5.2, are all computed using interval arithmetic.

Remark 5.3. The complexity of the enclosures obtained by Propositions 5.1 and 5.2 are similar to the methods in Chapter 4. Specifically, if X and W have n_g and n_{g_w} generators, and n_c and n_{c_w} constraints, respectively, then Proposition 5.1 gives $n_g + n_{g_w} + 2n$ generators and $n_c + n_{c_w}$ constraints, and Proposition 5.2 gives $0.5(n_g + n_{g_w})^2 + 2.5(n_g + n_{g_w}) + 2n$ generators and $0.5(n_c + n_{c_w})^2 + 2.5(n_c + n_{c_w})$ constraints.

Propositions 5.1 and 5.2 permit γ_x and γ_z to be chosen from $\square X$ and $\square Z$, respectively, whereas the corresponding results in Chapter 4 required these points to be chosen from the smaller sets X and Z . The following example illustrates the potential advantage of these extensions.

Example 5.1. Consider the nonlinear mapping $\mathbf{f} : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ defined by

$$f_1(\mathbf{x}) = 3x_1 - \frac{x_1^2}{7} - \frac{4x_1x_2}{4+x_1}, \quad f_2(\mathbf{x}) = -2x_2 + \frac{3x_1x_2}{4+x_1}, \quad (5.8)$$

and the constrained zonotope

$$X = \left\{ \left[\begin{array}{ccc} 0.5 & 1 & -0.5 \\ 0.5 & 0.5 & 0 \end{array} \right], \left[\begin{array}{c} 5 \\ 0.5 \end{array} \right], [-1 \ 1 \ -1], 2 \right\}.$$

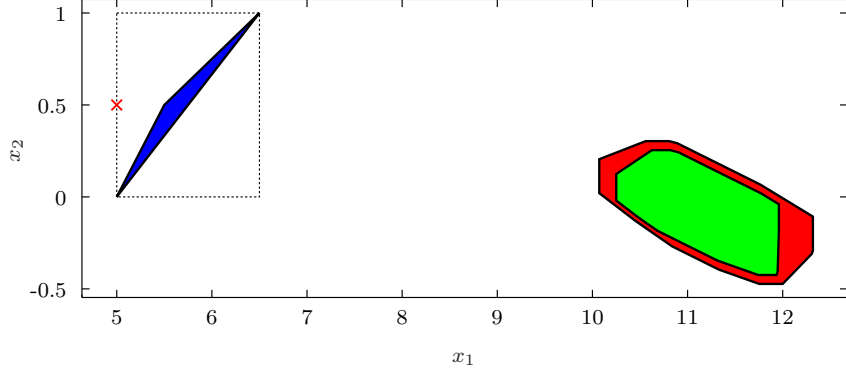


Figure 5.1: The sets X (blue), $\square X$ (dashed lines), the center of X (\times), the enclosures obtained using Proposition 5.2 with γ_x as the center of X (green), and using Theorem 4.3 with γ_x as the closest point in X to its center (red).

As shown in Figure 5.1, the center \mathbf{c} in this CG-rep of X does not actually lie in X , but does lie in $\square X$. Therefore, it is a valid choice of γ_x in Proposition 5.2, but not in Theorem 4.3 in Chapter 4. Figure 5.1 shows the enclosures of $\mathbf{f}(X)$ obtained using Proposition 5.2 with this choice of γ_x and using Theorem 4.3 with γ_x chosen as the closest point in X to \mathbf{c} , which is the best heuristic proposed in Chapter 4. The enclosure obtained using Proposition 5.2 is tighter. Thus, allowing γ_x to be chosen from the larger region $\square X$ can lead to less conservative results.

5.2.2 Update step

This section presents both mean-value and first-order Taylor methods for the update step in Algorithm 5.1, considering nonlinear measurement equations in contrast to the linear update step used in Chapter 4. Specifically, Lemmas 5.1 and 5.2 are used, respectively, to formulate the required enclosure in (5.4) as the generalized intersection of two constrained zonotopes.

Proposition 5.3. Let $\mathbf{g} : \mathbb{R}^n \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_v} \rightarrow \mathbb{R}^{n_y}$ be of class \mathcal{C}^1 , let $\mathbf{u} \in \mathbb{R}^{n_u}$, let $X \subset \mathbb{R}^n$ and $V \subset \mathbb{R}^{n_v}$ be constrained zonotopes, and choose any $\mathbf{y} \in \mathbb{R}^{n_y}$ such that $\mathbf{y} = \mathbf{g}(\mathbf{x}, \mathbf{u}, \mathbf{v})$ for some $(\mathbf{x}, \mathbf{v}) \in X \times V$. Choose any $\gamma_x \in \square X$ and any $\tilde{\mathbf{J}}_x \in \mathbb{R}^{n_y \times n}$. If Z_v is a constrained zonotope such that $-\mathbf{g}(\gamma_x, \mathbf{u}, V) \subseteq Z_v$ and $\mathbf{J}_x \in \mathbb{I}\mathbb{R}^{n_y \times n}$ is an interval matrix satisfying $\nabla_x^T \mathbf{g}(\square X, \mathbf{u}, V) \subseteq \mathbf{J}_x$, then

$$\{\mathbf{x} \in X : \mathbf{g}(\mathbf{x}, \mathbf{u}, \mathbf{v}) = \mathbf{y}, \mathbf{v} \in V\} \subseteq X \cap_{\mathbf{C}} Y,$$

where $\mathbf{C} = \tilde{\mathbf{J}}_x$, and $Y = (\mathbf{y} + \tilde{\mathbf{J}}_x \gamma_x) \oplus Z_v \oplus \triangleleft(\tilde{\mathbf{J}}_x - \mathbf{J}_x, X - \gamma_x)$.

Proof. Choose any $(\mathbf{x}, \mathbf{v}) \in X \times V$ satisfying $\mathbf{g}(\mathbf{x}, \mathbf{u}, \mathbf{v}) = \mathbf{y}$. Lemma 5.1 ensures that there exists a real matrix $\hat{\mathbf{J}}_x \in \mathbf{J}_x$ such that $\mathbf{g}(\mathbf{x}, \mathbf{u}, \mathbf{v}) = \mathbf{g}(\gamma_x, \mathbf{u}, \mathbf{v}) + \hat{\mathbf{J}}_x(\mathbf{x} - \gamma_x)$. Since

$\hat{\mathbf{J}}_x = \tilde{\mathbf{J}}_x + (\hat{\mathbf{J}}_x - \tilde{\mathbf{J}}_x)$ holds, then $\mathbf{g}(\mathbf{x}, \mathbf{u}, \mathbf{v}) = \mathbf{g}(\gamma_x, \mathbf{u}, \mathbf{v}) + \tilde{\mathbf{J}}_x(\mathbf{x} - \gamma_x) + (\hat{\mathbf{J}}_x - \tilde{\mathbf{J}}_x)(\mathbf{x} - \gamma_x)$. Consequently,

$$\begin{aligned} \tilde{\mathbf{J}}_x \mathbf{x} &= \mathbf{g}(\mathbf{x}, \mathbf{u}, \mathbf{v}) + \tilde{\mathbf{J}}_x \gamma_x - \mathbf{g}(\gamma_x, \mathbf{u}, \mathbf{v}) + (\tilde{\mathbf{J}}_x - \hat{\mathbf{J}}_x)(\mathbf{x} - \gamma_x) \\ &= \mathbf{y} + \tilde{\mathbf{J}}_x \gamma_x - \mathbf{g}(\gamma_x, \mathbf{u}, \mathbf{v}) + (\tilde{\mathbf{J}}_x - \hat{\mathbf{J}}_x)(\mathbf{x} - \gamma_x) \\ &\in (\mathbf{y} + \tilde{\mathbf{J}}_x \gamma_x) \oplus Z_v \oplus \triangleleft(\tilde{\mathbf{J}}_x - \hat{\mathbf{J}}_x, X - \gamma_x) = Y. \end{aligned}$$

Then, we conclude that $\{\mathbf{x} \in X : \mathbf{g}(\mathbf{x}, \mathbf{u}, \mathbf{v}) = \mathbf{y}, \mathbf{v} \in V\} \subseteq \{\mathbf{x} \in X : \tilde{\mathbf{J}}_x \mathbf{x} \in Y\} = X \cap_{\mathbf{C}} Y$. ■

Remark 5.4. The constrained zonotope Z_v in Proposition 5.3 can be obtained as $Z_v = -\mathbf{g}(\gamma_x, \mathbf{u}, \gamma_v) \oplus \triangleleft(-\mathbf{J}_v, V - \gamma_v) \supseteq -\mathbf{g}(\gamma_x, \mathbf{u}, V)$ for some $\gamma_v \in \square V$ and interval matrix $\mathbf{J}_v \supseteq \nabla_v^T \mathbf{g}(\gamma_x, \mathbf{u}, \square V)$. The matrix $\tilde{\mathbf{J}}_x$ is a free parameter in Proposition 5.3. Choosing $\tilde{\mathbf{J}}_x = \text{mid}(\mathbf{J}_x)$ gives $\text{mid}(\tilde{\mathbf{J}}_x - \mathbf{J}_x) = \mathbf{0}$, then $\triangleleft(\tilde{\mathbf{J}}_x - \mathbf{J}_x, X - \gamma_x) = \text{mid}(\tilde{\mathbf{J}}_x - \mathbf{J}_x)(X - \gamma_x) \oplus \mathbf{P}B_\infty^{n_y} = \mathbf{P}B_\infty^{n_y}$, with \mathbf{P} defined as in Theorem 4.1. This choice is adopted throughout this chapter.

Proposition 5.4. Let $\mathbf{g} : \mathbb{R}^n \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_v} \rightarrow \mathbb{R}^{n_y}$ be of class \mathcal{C}^2 , let $\mathbf{u} \in \mathbb{R}^{n_u}$, let $X = \{\mathbf{G}_x, \mathbf{c}_x, \mathbf{A}_x, \mathbf{b}_x\}$ and $V = \{\mathbf{G}_v, \mathbf{c}_v, \mathbf{A}_v, \mathbf{b}_v\}$ be constrained zonotopes with n_g generators and n_c constraints, and with n_{g_v} generators and n_{g_c} constraints, respectively, and choose any $\mathbf{y} \in \mathbb{R}^{n_y}$ such that $\mathbf{y} = \mathbf{g}(\mathbf{x}, \mathbf{u}, \mathbf{v})$ for some $(\mathbf{x}, \mathbf{v}) \in X \times V$. Denote $\mathbf{z} = (\mathbf{x}, \mathbf{v})$ and $Z = X \times V = \{\mathbf{G}, \mathbf{c}, \mathbf{A}, \mathbf{b}\} \subset \mathbb{R}^{n+n_v}$. For each $q = 1, 2, \dots, n_y$, let $\mathbf{Q}^{[q]} \in \mathbb{I}\mathbb{R}^{(n+n_v) \times (n+n_v)}$ and $\tilde{\mathbf{Q}}^{[q]} \in \mathbb{I}\mathbb{R}^{(n_g+n_{g_v}) \times (n_g+n_{g_v})}$ be interval matrices satisfying $\mathbf{Q}^{[q]} \supseteq \mathbf{H}_z g_q(\square X, \mathbf{u}, \square V)$ and $\tilde{\mathbf{Q}}^{[q]} \supseteq \mathbf{G}^T \mathbf{Q}^{[q]} \mathbf{G}$. Moreover, define $\tilde{\mathbf{c}}, \tilde{\mathbf{G}}, \tilde{\mathbf{G}}_d, \tilde{\mathbf{A}},$ and $\tilde{\mathbf{b}}$, as in Lemma 5.2. Finally, choose any $\gamma_z = (\gamma_x, \gamma_v) \in \square Z$, and let $\mathbf{L} \in \mathbb{I}\mathbb{R}^{n_y \times (n+n_v)}$ be an interval matrix satisfying $\mathbf{L}_{q,:} \supseteq (\mathbf{c} - \gamma_z)^T \mathbf{Q}^{[q]}$ for all $q = 1, \dots, n_y$. Then,

$$\{\mathbf{x} \in X : \mathbf{g}(\mathbf{x}, \mathbf{u}, \mathbf{v}) = \mathbf{y}, \mathbf{v} \in V\} \subseteq X \cap_{\mathbf{C}} Y,$$

where $\mathbf{C} = \nabla_x^T \mathbf{g}(\gamma_x, \mathbf{u}, \gamma_v)$, $Y = (\mathbf{y} - \mathbf{g}(\gamma_x, \mathbf{u}, \gamma_v) + \nabla_z^T \mathbf{g}(\gamma_x, \mathbf{u}, \gamma_v) \gamma_z) \oplus (-\nabla_v^T \mathbf{g}(\gamma_x, \mathbf{u}, \gamma_v) V) \oplus (-R)$, and $R = \tilde{\mathbf{c}} \oplus [\tilde{\mathbf{G}} \ \tilde{\mathbf{G}}_v] B_\infty(\tilde{\mathbf{A}}, \tilde{\mathbf{b}}) \oplus \triangleleft(\mathbf{L}, (\mathbf{c} - \gamma_z) \oplus 2\mathbf{G}B_\infty(\mathbf{A}, \mathbf{b}))$.

Proof. Choose $(\mathbf{x}, \mathbf{v}) = \mathbf{z} \in Z$ such that $\mathbf{g}(\mathbf{x}, \mathbf{u}, \mathbf{v}) = \mathbf{y}$. Lemma 5.2 ensures that there exist $\xi \in B_\infty(\mathbf{A}, \mathbf{b})$, $\tilde{\xi} \in B_\infty(\tilde{\mathbf{A}}, \tilde{\mathbf{b}})$, and $\hat{\mathbf{L}} \in \mathbf{L}$, such that

$$\begin{aligned} \mathbf{g}(\mathbf{x}, \mathbf{u}, \mathbf{v}) &= \mathbf{g}(\gamma_x, \mathbf{u}, \gamma_v) + \nabla_x^T \mathbf{g}(\gamma_x, \mathbf{u}, \gamma_v)(\mathbf{x} - \gamma_x) \\ &\quad + \nabla_v^T \mathbf{g}(\gamma_x, \mathbf{u}, \gamma_v)(\mathbf{v} - \gamma_v) + \hat{\mathbf{L}}(\mathbf{p} + 2\mathbf{G}\xi) + \tilde{\mathbf{c}} + [\tilde{\mathbf{G}} \ \tilde{\mathbf{G}}_v] \tilde{\xi}, \end{aligned}$$

where $\mathbf{p} = \mathbf{c} - \gamma_z$. Since $\mathbf{g}(\mathbf{x}, \mathbf{u}, \mathbf{v}) = \mathbf{y}$,

$$\begin{aligned} \nabla_x^T \mathbf{g}(\gamma_x, \mathbf{u}, \gamma_v) \mathbf{x} &= \mathbf{y} - \mathbf{g}(\gamma_x, \mathbf{u}, \gamma_v) + \nabla_z^T \mathbf{g}(\gamma_x, \mathbf{u}, \gamma_v) \gamma_z \\ &\quad - \nabla_v^T \mathbf{g}(\gamma_x, \mathbf{u}, \gamma_v) \mathbf{v} - \hat{\mathbf{L}}(\mathbf{p} + 2\mathbf{G}\boldsymbol{\xi}) - \tilde{\mathbf{c}} - [\tilde{\mathbf{G}} \ \bar{\mathbf{G}}_v] \bar{\boldsymbol{\xi}} \\ &\in (\mathbf{y} - \mathbf{g}(\gamma_x, \mathbf{u}, \gamma_v) + \nabla_z^T \mathbf{g}(\gamma_x, \mathbf{u}, \gamma_v) \gamma_z) \\ &\quad \oplus (-\nabla_v^T \mathbf{g}(\gamma_x, \mathbf{u}, \gamma_v) V) \oplus (-R) = Y \end{aligned}$$

Then, we conclude that $\{\mathbf{x} \in X : \mathbf{g}(\mathbf{x}, \mathbf{u}, \mathbf{v}) = \mathbf{y}, \mathbf{v} \in V\} \subseteq \{\mathbf{x} \in X : \nabla_x^T \mathbf{g}(\gamma_x, \mathbf{u}, \gamma_v) \mathbf{x} \in Y\} = X \cap_{\mathbf{C}} Y$. \blacksquare

Remark 5.5. If X and V have n_g and n_{g_v} generators, and n_c and n_{c_v} constraints, respectively, then the enclosure obtained by Proposition 5.3 has $2n_g + n_{g_v} + 2n_y$ generators and $2n_c + n_{c_v} + n_y$ constraints, and the enclosure obtained by Proposition 5.4 has $0.5(n_g + n_{g_v})^2 + 2.5(n_g + n_{g_v}) + 2n_y$ generators and $0.5(n_c + n_{c_v})^2 + 2.5(n_c + n_{c_v}) + n_y$ constraints.

Remark 5.6. If \mathbf{f} and \mathbf{g} are affine in \mathbf{w} and \mathbf{v} (i.e., $\mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{w}) = \boldsymbol{\phi}(\mathbf{x}, \mathbf{u}) + \boldsymbol{\Phi}(\mathbf{x}, \mathbf{u})\mathbf{w}$ and $\mathbf{g}(\mathbf{x}, \mathbf{u}, \mathbf{v}) = \boldsymbol{\psi}(\mathbf{x}, \mathbf{u}) + \boldsymbol{\Psi}(\mathbf{x}, \mathbf{u})\mathbf{v}$), then the constrained zonotopes $Z_w \supseteq \mathbf{f}(\gamma_x, \mathbf{u}, W)$ and $Z_v \supseteq -\mathbf{g}(\gamma_x, \mathbf{u}, V)$ in Propositions 5.1 and 5.3 can be computed exactly by $Z_w = \boldsymbol{\phi}(\gamma_x, \mathbf{u}) \oplus \boldsymbol{\Phi}(\gamma_x, \mathbf{u})W$ and $Z_v = -\boldsymbol{\psi}(\gamma_x, \mathbf{u}) \oplus (-\boldsymbol{\Psi}(\gamma_x, \mathbf{u})V)$, respectively.

5.2.3 Consistency step

This section presents both mean value and first-order Taylor methods for the consistency step in Algorithm 5.1. As in the previous section, the obtained enclosure is formulated as the generalized intersection of two constrained zonotopes. Since the proposed methods for the consistency step are direct consequences of Propositions 5.3 and 5.4, these are presented as corollaries.

Corollary 5.1. Let $\mathbf{h} : \mathbb{R}^n \rightarrow \mathbb{R}^{n_h}$ be of class \mathcal{C}^1 , and let $X \subset \mathbb{R}^n$ be a constrained zonotope. Choose any $\gamma_x \in \square X$ and any $\tilde{\mathbf{J}}_x \in \mathbb{R}^{n_h \times n}$. If $\mathbf{J}_x \in \mathbb{I}\mathbb{R}^{n_h \times n}$ is an interval matrix satisfying $\nabla_x^T \mathbf{h}(\square X) \subseteq \mathbf{J}_x$, then $\{\mathbf{x} \in X : \mathbf{h}(\mathbf{x}) = \mathbf{0}\} \subseteq X \cap_{\mathbf{D}} H$, where $\mathbf{D} = \tilde{\mathbf{J}}_x$, and $H = (\tilde{\mathbf{J}}_x \gamma_x - \mathbf{h}(\gamma_x)) \oplus \triangleleft(\tilde{\mathbf{J}}_x - \mathbf{J}_x, X - \gamma_x)$.

Proof. Choose any $\mathbf{x} \in X$ satisfying $\mathbf{h}(\mathbf{x}) = \mathbf{0}$. Lemma 5.1 ensures that there exists a real matrix $\hat{\mathbf{J}}_x \in \mathbf{J}_x$ such that $\mathbf{h}(\mathbf{x}) = \mathbf{h}(\gamma_x) + \hat{\mathbf{J}}_x(\mathbf{x} - \gamma_x)$. Since $\hat{\mathbf{J}}_x = \tilde{\mathbf{J}}_x + (\hat{\mathbf{J}}_x - \tilde{\mathbf{J}}_x)$ holds, then $\mathbf{h}(\mathbf{x}) = \mathbf{h}(\gamma_x) + \tilde{\mathbf{J}}_x(\mathbf{x} - \gamma_x) + (\hat{\mathbf{J}}_x - \tilde{\mathbf{J}}_x)(\mathbf{x} - \gamma_x)$. Consequently,

$$\begin{aligned} \tilde{\mathbf{J}}_x \mathbf{x} &= \mathbf{h}(\mathbf{x}) + \tilde{\mathbf{J}}_x \gamma_x - \mathbf{h}(\gamma_x) + (\tilde{\mathbf{J}}_x - \hat{\mathbf{J}}_x)(\mathbf{x} - \gamma_x) \\ &= \mathbf{0} + \tilde{\mathbf{J}}_x \gamma_x - \mathbf{h}(\gamma_x) + (\tilde{\mathbf{J}}_x - \hat{\mathbf{J}}_x)(\mathbf{x} - \gamma_x) \\ &\in (\tilde{\mathbf{J}}_x \gamma_x - \mathbf{h}(\gamma_x)) \oplus \triangleleft(\tilde{\mathbf{J}}_x - \mathbf{J}_x, X - \gamma_x) = H. \end{aligned}$$

Therefore, $\{\mathbf{x} \in X : \mathbf{h}(\mathbf{x}) = \mathbf{0}\} \subseteq \{\mathbf{x} \in X : \tilde{\mathbf{J}}_x \mathbf{x} \in H\} = X \cap_{\mathbf{D}} H$. \blacksquare

Remark 5.7. As in the update step, the matrix $\tilde{\mathbf{J}}_x$ is a free parameter in Proposition 5.1. If $\tilde{\mathbf{J}}_x = \text{mid}(\mathbf{J}_x)$, then $\text{mid}(\tilde{\mathbf{J}}_x - \mathbf{J}_x) = \mathbf{0}$, and $\langle (\tilde{\mathbf{J}}_x - \mathbf{J}_x, X - \gamma_x) = \text{mid}(\tilde{\mathbf{J}}_x - \mathbf{J}_x)(X - \gamma_x) \oplus \mathbf{P}B_\infty^{n_y} = \mathbf{P}B_\infty^{n_y}$. Therefore, this choice is adopted also for the consistency step.

Corollary 5.2. Let $\mathbf{h} : \mathbb{R}^n \rightarrow \mathbb{R}^{n_h}$ be of class \mathcal{C}^2 and let $X = \{\mathbf{G}, \mathbf{c}, \mathbf{A}, \mathbf{b}\}$ be a constrained zonotope with n_g generators and n_c constraints. For each $q = 1, 2, \dots, n_h$, let $\mathbf{Q}^{[q]} \in \mathbb{I}\mathbb{R}^{n \times n}$ and $\tilde{\mathbf{Q}}^{[q]} \in \mathbb{I}\mathbb{R}^{n_g \times n_g}$ be interval matrices satisfying $\mathbf{Q}^{[q]} \supseteq \mathbf{H}_x h_q(\square X)$ and $\tilde{\mathbf{Q}}^{[q]} \supseteq \mathbf{G}^T \mathbf{Q}^{[q]} \mathbf{G}$. Moreover, define $\tilde{\mathbf{c}}$, $\tilde{\mathbf{G}}$, $\tilde{\mathbf{G}}_d$, $\tilde{\mathbf{A}}$, and $\tilde{\mathbf{b}}$, as in Lemma 5.2. Finally, choose any $\gamma_x \in \square X$ and let $\mathbf{L} \in \mathbb{I}\mathbb{R}^{n_h \times n}$ be an interval matrix satisfying $\mathbf{L}_{q,:} \supseteq (\mathbf{c} - \gamma_x)^T \mathbf{Q}^{[q]}$ for all $q = 1, \dots, n_h$. Then,

$$\{\mathbf{x} \in X : \mathbf{h}(\mathbf{x}) = \mathbf{0}\} \subseteq X \cap_{\mathbf{D}} H,$$

where $\mathbf{D} = \nabla_x^T \mathbf{h}(\gamma_x)$, $H = (-\mathbf{h}(\gamma_x) + \nabla_x^T \mathbf{h}(\gamma_x) \gamma_x) \oplus (-R)$, and $R = \tilde{\mathbf{c}} \oplus [\tilde{\mathbf{G}} \ \tilde{\mathbf{G}}_d] B_\infty(\tilde{\mathbf{A}}, \tilde{\mathbf{b}}) \oplus \langle (\mathbf{L}, (\mathbf{c} - \gamma_x) \oplus 2\mathbf{G}B_\infty(\mathbf{A}, \mathbf{b}))$.

Proof. Choose $\mathbf{x} \in X$ such that $\mathbf{h}(\mathbf{x}) = \mathbf{0}$. Lemma 5.2 ensures that there exist $\boldsymbol{\xi} \in B_\infty(\mathbf{A}, \mathbf{b})$, $\tilde{\boldsymbol{\xi}} \in B_\infty(\tilde{\mathbf{A}}, \tilde{\mathbf{b}})$, and $\hat{\mathbf{L}} \in \mathbf{L}$, such that

$$\begin{aligned} \mathbf{h}(\mathbf{x}) &= \mathbf{h}(\gamma_x) + \nabla_x^T \mathbf{h}(\gamma_x)(\mathbf{x} - \gamma_x) \\ &\quad + \hat{\mathbf{L}}(\mathbf{p} + 2\mathbf{G}\boldsymbol{\xi}) + \tilde{\mathbf{c}} + [\tilde{\mathbf{G}} \ \tilde{\mathbf{G}}_d] \tilde{\boldsymbol{\xi}}. \end{aligned}$$

with $\mathbf{p} = \mathbf{c} - \gamma_x$. Since $\mathbf{h}(\mathbf{x}) = \mathbf{0}$, we have $\nabla_x^T \mathbf{h}(\gamma_x) \mathbf{x} = -\mathbf{h}(\gamma_x) + \nabla_x^T \mathbf{h}(\gamma_x) \gamma_x - \hat{\mathbf{L}}(\mathbf{p} + 2\mathbf{G}\boldsymbol{\xi}) - \tilde{\mathbf{c}} - [\tilde{\mathbf{G}} \ \tilde{\mathbf{G}}_d] \tilde{\boldsymbol{\xi}}$, and therefore

$$\nabla_x^T \mathbf{h}(\gamma_x) \mathbf{x} \in (-\mathbf{h}(\gamma_x) + \nabla_x^T \mathbf{h}(\gamma_x) \gamma_x) \oplus (-R) = H.$$

We conclude that $\{\mathbf{x} \in X : \mathbf{h}(\mathbf{x}) = \mathbf{0}\} \subseteq \{\mathbf{x} \in X : \nabla_x^T \mathbf{h}(\gamma_x) \mathbf{x} \in H\} = X \cap_{\mathbf{D}} H$. \blacksquare

Remark 5.8. If X has n_g generators and n_c constraints, then the enclosure obtained from Proposition 5.1 has $2n_g + n_y$ generators and $2n_c + n_y$ constraints, and the enclosure obtained from Proposition 5.2 has $0.5n_g^2 + 2.5n_g + 2n_y$ generators and $0.5n_c^2 + 2.5n_c + n_y$ constraints.

Remark 5.9. The enclosures in Propositions 5.1 and 5.2 can be tightened if an enclosure X_F in CG-rep of the feasible state set $\{\mathbf{x} \in \mathbb{R}^n : \mathbf{h}(\mathbf{x}) = \mathbf{0}\}$ is known *a priori*. Such an enclosure can be obtained offline by using, for instance, the contractor programming methods in Chabert & Jaulin (2009). In both propositions, \mathbf{h} is conservatively approximated over X , and the size of the resulting constrained zonotope Z is proportional to the size of X . If X is large, significant improvement can result from setting $X \leftarrow X \cap X_F$ prior to applying the proposition. This situation is likely in practice because, within the overall estimation framework (5.3)–(5.5), the set \hat{X}_k will play the role of X in Propositions 5.1 and 5.2, and \hat{X}_k can be very conservative before accounting for the invariant $\mathbf{h}(\mathbf{x}_k) = \mathbf{0}$ (see Section 5.4.2).

Remark 5.10. In the literature, different methods has been proposed to solve a more general problem of computing the intersection of sets with nonlinear equations, but applied to different problem such as reachability analysis (Kochdumper & Althoff, 2020b). Nevertheless, none of the existing methods can be applied to sets described by constrained zonotopes, and the obtained enclosure is generally not convex.

5.2.4 Selection of approximation point

The methods proposed in this chapter require heuristics to choose an approximation point $(\gamma_x, \gamma_w, \gamma_v) \in \square X \times \square W \times \square V$, where X stands for either \tilde{X}_{k-1} , \bar{X}_k , or \hat{X}_k depending on the step in (5.3)–(5.5). As discussed in Chapter 4, the center of the CG-rep of $X \times W \times V$ cannot be chosen in general because it may not belong to either $X \times W \times V$ or $\square X \times \square W \times \square V$. However, in contrast to Chapter 4, the center of the interval $\square X \times \square W \times \square V$ is a valid choice here. This first heuristic is summarized as follows:

C1) $(\gamma_x, \gamma_w, \gamma_v)$ is given by the center of $\square X \times \square W \times \square V$.

Despite its efficiency, *C1* is not optimal in any sense and can lead to conservative enclosures. Following Chapter 4, we next present an improved heuristic *C2* specifically for use with the methods based on mean value extensions in Propositions 5.1 and 5.3, and Corollary 5.1 (the exact heuristic in Chapter 4 is not optimal here because it restricts $(\gamma_x, \gamma_w, \gamma_v)$ to $X \times W \times V$ rather than $\square X \times \square W \times \square V$). Propositions 5.1 and 5.3, and Corollary 5.1 all apply the CZ-inclusion operator defined in Theorem 4.1 with the second argument taking the form $X - \gamma_x$. The idea behind *C2* is therefore to choose γ_x so as to minimize the conservatism of this CZ-inclusion operator.

In this sense, consider the CZ-inclusion operator $\triangleleft(\mathbf{J}, Z - \gamma)$ for arbitrary $Z = \{\mathbf{G}, \mathbf{c}, \mathbf{A}, \mathbf{b}\} \subset \mathbb{R}^{n_z}$, $\gamma \in \mathbb{R}^{n_z}$, and $\mathbf{J} \in \mathbb{IR}^{m \times n_z}$. As per Theorem 4.1 and Remark 4.5, computing $\triangleleft(\mathbf{J}, Z - \gamma)$ requires a zonotope $\{\bar{\mathbf{G}}, \bar{\mathbf{c}}\} \supseteq (Z - \gamma)$ that is computed by eliminating all constraints from $(Z - \gamma)$ using the constraint elimination algorithm given by Method 3.4. Based on that algorithm, Chapter 4 derived a closed form expression for the resulting center $\bar{\mathbf{c}}$ as a function of $(\mathbf{G}, \mathbf{c}, \mathbf{A}, \mathbf{b})$ and γ , obtained by displacing (4.12) by $-\gamma$, which takes the form

$$\bar{\mathbf{c}} = \mathbf{c} - \gamma + \delta(\mathbf{G}, \mathbf{A}, \mathbf{b}).$$

The definition of $\delta(\mathbf{G}, \mathbf{A}, \mathbf{b})$ can be deduced from (4.12) and is omitted here for brevity. This $\bar{\mathbf{c}}$ is then used to compute $\mathbf{m} \supseteq (\mathbf{J} - \text{mid}(\mathbf{J}))\bar{\mathbf{c}}$ using interval arithmetic, and the size of the final enclosure $\triangleleft(\mathbf{J}, Z - \gamma)$ is proportional to $\text{rad}(\mathbf{m}) = (1/2)\text{diam}(\mathbf{m})$. Thus, the aim is to choose γ so as to minimize $\text{diam}(\mathbf{m})$.

Proposition 5.5. Let $Z = \{\mathbf{G}, \mathbf{c}, \mathbf{A}, \mathbf{b}\} \subset \mathbb{R}^m$, $\mathbf{J} \in \mathbb{IR}^{m \times n}$, and $[\mathbf{z}^L, \mathbf{z}^U] = \square Z$. For any choice of $\gamma \in \square Z$, let $\mathbf{m}_\gamma \supseteq (\mathbf{J} - \text{mid}(\mathbf{J}))\bar{\mathbf{c}}_\gamma$ be an interval vector computed using interval

arithmetic, where $\bar{\mathbf{c}}_\gamma = \mathbf{c} - \gamma + \boldsymbol{\delta}(\mathbf{G}, \mathbf{A}, \mathbf{b})$. Then, $\gamma^* \in \square Z$ minimizes $\|\text{diam}(\mathbf{m}_\gamma)\|_1$ iff it is the solution to the linear program (LP)

$$\min_{\gamma} \|\Theta \bar{\mathbf{c}}_\gamma\|_1, \quad \text{s.t.} \quad \mathbf{z}^L \leq \gamma \leq \mathbf{z}^U, \quad (5.9)$$

where $\Theta_{jj} = \sum_{i=1}^m \text{diam}(J_{ij})$ and $\Theta_{ij} = 0$ for all $i \neq j$.

Proof. Each component of $(\mathbf{J} - \text{mid}(\mathbf{J})) \in \mathbb{R}^{m \times n}$ is an interval satisfying $(J_{ij} - \text{mid}(J_{ij})) = (1/2)\text{diam}(J_{ij})[-1, 1]$. Moreover, $a[-1, 1] = |a|[-1, 1]$ holds for every $a \in \mathbb{R}$. Therefore $m_{\gamma,i} = \sum_{j=1}^n (1/2)\text{diam}(J_{ij})|\bar{c}_{\gamma,j}|[-1, 1]$. Consequently, $\text{diam}(m_{\gamma,i}) = \sum_{j=1}^n \text{diam}(J_{ij})|\bar{c}_{\gamma,j}|$, and

$$\begin{aligned} \|\text{diam}(\mathbf{m}_\gamma)\|_1 &= \sum_{i=1}^m \sum_{j=1}^n \text{diam}(J_{ij})|\bar{c}_{\gamma,j}| = \sum_{j=1}^n \left(\sum_{i=1}^m \text{diam}(J_{ij}) \right) |\bar{c}_{\gamma,j}| \\ &= \sum_{j=1}^n \Theta_{jj} |\bar{c}_{\gamma,j}| = \|\Theta \bar{\mathbf{c}}_\gamma\|_1. \end{aligned}$$

The constraints in (5.9) follow directly from the requirement that $\gamma \in \square Z$. ■

This heuristic is summarized as follows:

C2) γ_x , γ_w , and γ_v are given by the points obtained from Proposition 5.5 for $(\mathbf{J}, X - \gamma_x)$ in Proposition 5.1, $(\tilde{\mathbf{J}} - \mathbf{J}, X - \gamma_x)$ in Proposition 5.3 and Corollary 5.1, $(\mathbf{J}_w, W - \gamma_w)$, and $(\mathbf{J}_v, V - \gamma_v)$, respectively.

Next, we present a heuristic specifically for the methods based on first-order Taylor extensions in Propositions 5.2 and 5.4, and Corollary 5.2. The conservatism of these methods is directly related to the conservatism in the remainder R , which is mostly affected by the size of the interval matrices $\mathbf{Q}^{[q]}$, $\tilde{\mathbf{Q}}^{[q]}$, and \mathbf{L} . The matrices $\mathbf{Q}^{[q]}$ and $\tilde{\mathbf{Q}}^{[q]}$ are unaffected by the choice of $(\gamma_x, \gamma_w, \gamma_v)$. However, similarly to Theorem 4.3, the radius of \mathbf{L} is proportional to the differences $\mathbf{c}_x - \gamma_x$, $\mathbf{c}_w - \gamma_w$, and $\mathbf{c}_v - \gamma_v$. Therefore, based on Corollary 4.2 and Proposition 4.3, we propose the following heuristic for the first-order Taylor extensions:

C3) $(\gamma_x, \gamma_w, \gamma_v)$ is the closest point to the center of $X \times W \times V$ that belongs to $\square X \times \square W \times \square V$, obtained by solving the respective LPs $\min \{\|\gamma - \mathbf{c}_x\|_1 : \gamma \in \square X\}$, $\min \{\|\gamma - \mathbf{c}_w\|_1 : \gamma \in \square W\}$, and $\min \{\|\gamma - \mathbf{c}_v\|_1 : \gamma \in \square V\}$.

5.2.5 Linear case

This section investigates the results obtained by the prediction, update, and consistency steps for nonlinear systems developed in this chapter when applied to linear systems. In this case, the resulting enclosures are straightforward. Consider the linear discrete-time system

$$\mathbf{x}_k = \mathbf{A}\mathbf{x}_{k-1} + \mathbf{B}_u \mathbf{u}_{k-1} + \mathbf{B}_w \mathbf{w}_{k-1}, \quad (5.10a)$$

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{D}_u\mathbf{u}_k + \mathbf{D}_v\mathbf{v}_k, \quad (5.10b)$$

where $\mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{B}_u \in \mathbb{R}^{n \times n_u}$, $\mathbf{B}_w \in \mathbb{R}^{n \times n_w}$, $\mathbf{C} \in \mathbb{R}^{n_y \times n}$, $\mathbf{D}_u \in \mathbb{R}^{n_y \times n_u}$, $\mathbf{D}_v \in \mathbb{R}^{n_y \times n_v}$, with known polytopic bounds $(\mathbf{x}_0, \mathbf{w}_k, \mathbf{v}_k) \in X_0 \times W \times V$. Moreover, assume that the trajectories of (5.10) satisfy the linear invariants $\mathbf{E}\mathbf{x}_k = \mathbf{d}$, with $\mathbf{E} \in \mathbb{R}^{n_d \times n}$, and $\mathbf{d} \in \mathbb{R}^{n_d}$. Given the previous set \tilde{X}_{k-1} , the prediction step (5.3) and the update step (5.4) are computed exactly for (5.10a)–(5.10b) as in Section 3.5:

$$\bar{X}_k = \mathbf{A}\tilde{X}_{k-1} \oplus \mathbf{B}_u\mathbf{u}_{k-1} \oplus \mathbf{B}_wW, \quad (5.11)$$

$$\hat{X}_k = \bar{X}_k \cap_{\mathbf{C}} ((\mathbf{y}_k - \mathbf{D}_u\mathbf{u}) \oplus (-\mathbf{D}_vV)). \quad (5.12)$$

All the set operations in (5.11)–(5.12) can be computed straightforwardly using (3.8)–(3.10). To compute the consistency step (5.5), note that in this case this can be written as $\tilde{X}_k \supseteq \{\mathbf{x} \in \hat{X}_k : \mathbf{E}\mathbf{x}_k \in \{\mathbf{d}\}\}$, where $\{\mathbf{d}\}$ denotes a singleton that contains only the point \mathbf{d} . Therefore, if $\hat{X}_k = \{\hat{\mathbf{G}}_k, \hat{\mathbf{c}}_k, \hat{\mathbf{A}}_k, \hat{\mathbf{b}}_k\}$, then \tilde{X}_k is given by

$$\tilde{X}_k = \hat{X}_k \cap_{\mathbf{E}} \{\mathbf{d}\} = \left\{ \hat{\mathbf{G}}_k, \hat{\mathbf{c}}_k, \begin{bmatrix} \hat{\mathbf{A}}_k \\ \mathbf{E}\hat{\mathbf{G}}_k \end{bmatrix}, \begin{bmatrix} \hat{\mathbf{b}}_k \\ \mathbf{d} - \mathbf{E}\hat{\mathbf{c}}_k \end{bmatrix} \right\}. \quad (5.13)$$

Hence, the consistency step can be computed exactly as well. Therefore, similarly to the method described in Section 3.5 and as a direct extension for linear invariants of the linear method proposed in Scott et al. (2016), the only source of conservatism in the set-valued state estimation of (5.10) using constrained zonotopes through the steps (5.11)–(5.13) arises if the complexity of the sets are limited, which requires the use of complexity reduction methods (Section 3.3).

Remark 5.11. If complexity reduction methods are not employed, the enclosures obtained for both the prediction step (5.11) and consistency step (5.13) are exact using constrained zonotopes. Therefore, for linear systems, in the case the complexity of the sets is not limited, since the trajectories of the system satisfy $\mathbf{E}\mathbf{x}_k = \mathbf{d}$ by assumption, it is sufficient to compute the consistency step (5.13) only at $k = 0$.

5.3 Computational complexity

Table 5.1 shows the computational complexity of the methods proposed in this chapter for the prediction, update, and consistency steps, using the mean value extension (Propositions 5.1 and 5.3, and Corollary 5.1) and the first-order Taylor extension (Propositions 5.2 and 5.4, and Corollary 5.2). To derive these complexities, we consider that the enclosures $(\tilde{X}_{k-1}, \bar{X}_k, \hat{X}_k)$ have $(\tilde{n}_g, \bar{n}_g, \hat{n}_g)$ generators and $(\tilde{n}_c, \bar{n}_c, \hat{n}_c)$ constraints, respectively. Moreover, (W, V) have (n_{g_w}, n_{g_v}) generators and (n_{c_w}, n_{c_v}) constraints, and we define

$(m_w, m_{g_w}, m_{c_w}) \triangleq (n + n_w, \tilde{n}_g + n_{g_w}, \tilde{n}_c + n_{c_w})$, and $(m_v, m_{g_v}, m_{c_v}) \triangleq (n + n_v, \bar{n}_g + n_{g_v}, \bar{n}_c + n_{c_v})$. As in Chapter 4, we assume that evaluations of nonlinear real functions and nonlinear inclusion functions have complexity $O(1)$, and that all the LPs (including the ones necessary to compute the interval hulls) are solved at least with the performance of the method proposed in Kelner & Spielman (2006). This method has (simplified) polynomial complexity $O(N_d N_c^3)$, with N_d and N_c the number of decision variables and constraints, respectively.

The complexities of the prediction, update, and consistency steps, using the mean value and first-order Taylor extensions, are similar to the previous prediction methods proposed in Chapter 4. In all the complexities shown in Table 5.1, the higher order terms such as $(m_w m_{g_w} + m_{c_w})(m_{g_w} + m_{c_w})^3$ come from both the interval hull computations and the constraint elimination procedure used to obtain the zonotope enclosure required by Theorem 1. The other terms come from matrix products that appear in the proposed expressions to compute the respective CG-rep variables.

Table 5.1 also shows a simplified complexity analysis of the proposed methods for each step. In this analysis, we consider that every variable is proportional to the space dimension n , and that $(\tilde{X}_{k-1}, \bar{X}_k, \hat{X}_k)$ have the same number of generators and constraints (this can be achieved by using generator reduction and constraint elimination methods after each step). Details on the complexities of the basic operations with constrained zonotopes are found in Appendix A.

Table 5.1: Computational complexity $O(\cdot)$ of the prediction, update, and consistency steps using constrained zonotopes.

Step	Mean value extension	Simplified
Prediction	$nm_w m_{g_w} + (m_w m_{g_w} + m_{c_w})(m_{g_w} + m_{c_w})^3$	n^5
Update	$n_y(m_v m_{g_v} + n_y) + (m_v m_{g_v} + m_{c_v})(m_{g_v} + m_{c_v})^3$	n^5
Consistency	$n_h(n\hat{n}_g + n_h) + (n\hat{n}_g + \hat{n}_c)(\hat{n}_g + \hat{n}_c)^3$	n^5
Step	First order extension	Simplified
Prediction	$n(m_w^2 m_{g_w} + m_w m_{g_w}^2) + (m_w m_{g_w} + m_{c_w})(m_{g_w} + m_{c_w})^3$	n^5
Update	$n_y^2 + n_y(m_v^2 m_{g_v} + m_v m_{g_v}^2) + (m_v m_{g_v} + m_{c_v})(m_{g_v} + m_{c_v})^3$	n^5
Consistency	$n_h^2 + n_h(n^2 \hat{n}_g + n \hat{n}_g^2) + (n\hat{n}_g + \hat{n}_c)(\hat{n}_g + \hat{n}_c)^3$	n^5

5.4 Numerical examples

This section evaluates the accuracy of the set-valued state estimation methods proposed in Section 5.2. Let CZMV denote the method based entirely on the mean value extension, using Proposition 5.1 for the prediction step and Proposition 5.3 for the update step, but with no consistency step. Moreover, let CZMV+C denote the method CZMV with the addition of the consistency step using Corollary 5.1, let CZMV+F denote CZMV with the addition of an intersection with an enclosure of the feasible state set as described in Remark 5.9, and let CZMV+FC denote CZMV with the addition of both the intersection in Remark 5.9 and then the consistency step using Corollary 5.1. These are referred to

collectively as CZMV-like methods. Analogously, let CZFO denote the method based entirely on first-order Taylor extensions, using Proposition 5.2 for the prediction step and Proposition 5.4 for the update step, and let CZFO+C, CZFO+F, and CZFO+FC denote CZFO with the addition of, respectively, the consistency step using Proposition 5.2, the intersection described in Remark 5.9, and both the intersection and the consistency step. In CZMV-like methods, complexity reduction is applied after the consistency step using the complexity reduction methods described in Section 3.3, with constraint elimination performed prior to generator reduction. Due to the quadratic complexity increase of the enclosure in each intermediate step of CZFO-like methods (see Remarks 5.3, 5.5 and 5.7), complexity reduction is applied after all three steps in these methods. Heuristic $C2$ is used for choosing $(\gamma_x, \gamma_w, \gamma_v)$ in CZMV-like methods, and heuristic $C3$ in CZFO-like methods.

We also compare our results with two nonlinear zonotope methods with prediction steps based on the Mean Value Theorem proposed in Alamo et al. (2005) and Taylor's Theorem proposed in Combastel (2005). These are denoted by ZMV and ZFO¹, respectively. In both zonotope methods, the nonlinear update step is given by the intersection method presented in Bravo et al. (2006), with strips computed as in Alamo et al. (2005). Generator reduction is applied after the update step using Method 4 in Yang & Scott (2018). In addition, we denote by ZMV+F and ZFO+F the methods ZMV and ZFO with the addition of the intersection discussed in Remark 5.9. Since zonotopes are not closed under intersection, this intersection is enclosed by converting the *a priori* enclosure X_F from CG-rep to half-space representation as described in Scott et al. (2016), representing X_F as an intersection of strips, and then using the method for bounding the intersection of a zonotope with a set of strips from Bravo et al. (2006). As with constrained zonotopes, these are referred to as ZMV-like and ZFO-like methods, respectively.

5.4.1 A system with nonlinear measurement equations

To demonstrate the effectiveness of the methods proposed in this chapter for set-valued state estimation of systems with nonlinear outputs, we first consider the nonlinear discrete-time system $\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}) + \mathbf{w}_{k-1}$, where \mathbf{f} is defined by (5.8) and $\mathbf{w}_k \in \mathbb{R}^2$ denotes process uncertainties with $\|\mathbf{w}_k\|_\infty \leq 0.4$. The measurements are given by

$$\begin{aligned} y_{1,k} &= x_{1,k} - \sin\left(\frac{x_{2,k}}{2}\right) + v_{1,k}, \\ y_{2,k} &= -x_{1,k}x_{2,k} + x_{2,k} + v_{2,k}, \end{aligned} \tag{5.14}$$

¹In a simplified analysis, ZMV and ZFO have computational complexities $O(n^4)$ and $O(n^5)$, respectively. See Section 4.4 for details.

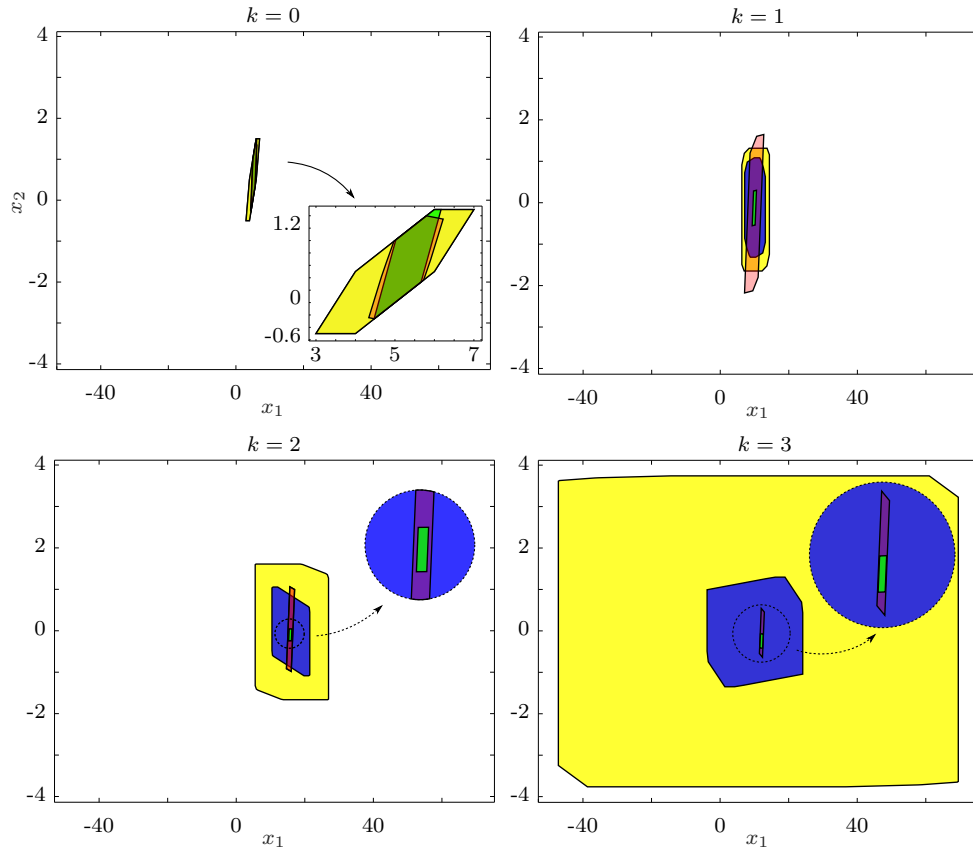


Figure 5.2: The enclosures \tilde{X}_k from the first four time steps of set-valued state estimation in the example in Section 5.4.1 using ZMV (yellow), ZFO (blue), CZMV (green), and CZFO (orange).

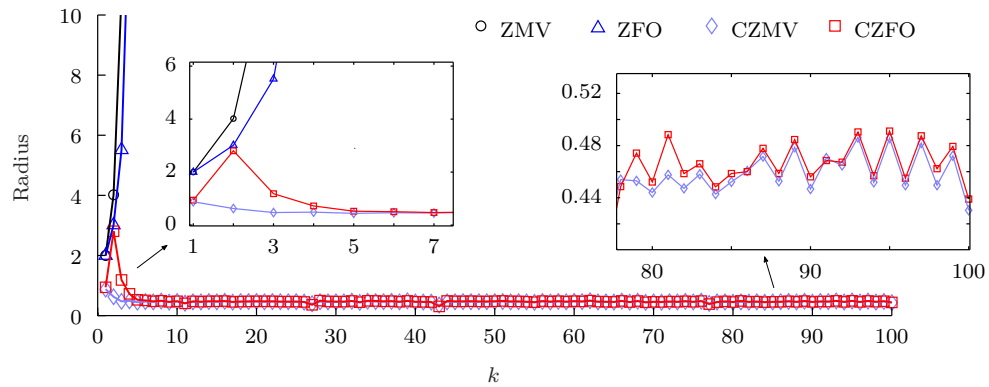


Figure 5.3: The radii of the estimated enclosures \tilde{X}_k in the example in Section 5.4.1 obtained using ZMV, ZFO, CZMV, and CZFO.

with $\|\mathbf{v}_k\|_\infty \leq 0.4$. Finally, let

$$X_0 = \left\{ \left[\begin{array}{ccc} 0.5 & 1 & -0.5 \\ 0.5 & 0.5 & 0 \end{array} \right], \left[\begin{array}{c} 5 \\ 0.5 \end{array} \right] \right\}. \quad (5.15)$$

Figure 5.2 shows the estimated enclosures \tilde{X}_k (since there are no invariants, these are $\tilde{X}_k = \hat{X}_k$) for $k = 0, 1, 2, 3$, obtained using ZMV, ZFO, CZMV, and CZFO. In this case,

Z_w and Z_v are computed as in Remark 5.6. The number of generators and constraints is limited to 8 and 3, respectively. The simulations were run in MATLAB 9.1 with INTLAB 12 and CPLEX 12.8, on a laptop with 32 GB RAM and an Intel Core i7-9750H processor. The first set \tilde{X}_0 coincides with X_0 for both ZMV and ZFO, which demonstrates that the update step method using zonotopes can be very conservative with nonlinear measurements, making the first update ineffective in this example. On the other hand, the sets \tilde{X}_0 obtained by CZMV and CZFO have reduced volume relative to X_0 , showing that X_0 was effectively tightened by the first measurement. In addition, in contrast to CZMV and CZFO, the size of the enclosures \tilde{X}_k for both ZMV and ZFO increases substantially with time. This is corroborated by Figure 5.3, which illustrates the radii of the sets \tilde{X}_k (half of the length of the longest edge of the interval hull). Note that the radii of the sets obtained by ZMV and ZFO increase to infinity, while the radii of the sets obtained by CZMV and CZFO remain finite. This result corroborates the improved accuracy achieved by using constrained zonotopes for computing the update step with nonlinear measurement equations. Lastly, Table 5.2 shows the average computational times per time step of each method, together with the computational times spent in complexity reduction of the enclosures. The latter is included to distinguish the computational burden of the proposed methods from the complexity reduction procedures. Note that CZMV and CZFO were able to provide accurate enclosures with an increase of 114% and 52.3% of the execution times with respect to ZMV and ZFO, respectively. Nevertheless, as mentioned above, the sets obtained by ZMV and ZFO increased to infinity in few steps, and therefore cannot provide any information about the state trajectories.

Table 5.2: Total and complexity reduction average execution times per time step of the state estimators for the first example.

	ZMV	CZMV	ZFO	CZFO
Total	30.5 ms	65.3 ms	47.4 ms	72.2 ms
Red.	4.4 ms	4.8 ms	6.6 ms	14.2 ms

5.4.2 A system with nonlinear measurements and invariants

The second example involves state estimation of the attitude of a flying robot. The robot is driven by angular velocity $\check{\mathbf{u}}_k \in \mathbb{R}^3$, with attitude expressed as a rotation quaternion $\mathbf{x}_k \in \mathbb{R}^4$ satisfying $\|\mathbf{x}_k\|_2^2 = 1$, which defines the invariant $h(\mathbf{x}_k) = \|\mathbf{x}_k\|_2^2 - 1 = 0$ to be used in the consistency step (5.5). The known value \mathbf{u}_k of the physical input $\check{\mathbf{u}}_k$ is measured by gyroscopes and therefore is considered to be corrupted by additive noise $\mathbf{w}_k \in \mathbb{R}^3$. Physically, the system is driven by the uncorrupted signal $\check{\mathbf{u}}_k = \mathbf{u}_k - \mathbf{w}_k$. The attitude \mathbf{x}_k evolves in discrete time according to (Teixeira et al., 2009; Lefferts et al., 1982)

$$\mathbf{x}_k = \left(\cos(p(\mathbf{u}_k, \mathbf{w}_k))\mathbf{I}_4 - \frac{T_s}{2} \frac{\sin(p(\mathbf{u}_k, \mathbf{w}_k))}{p(\mathbf{u}_k, \mathbf{w}_k)} \boldsymbol{\Omega}(\mathbf{u}_k, \mathbf{w}_k) \right) \mathbf{x}_{k-1}, \quad (5.16)$$

where T_s is the sampling time and

$$p(\mathbf{u}_k, \mathbf{w}_k) = \frac{T_s}{2} \|\check{\mathbf{u}}_k\|_2, \quad \Omega(\mathbf{u}_k, \mathbf{w}_k) = \begin{bmatrix} 0 & \check{u}_{3,k} & -\check{u}_{2,k} & \check{u}_{1,k} \\ -\check{u}_{3,k} & 0 & \check{u}_{1,k} & \check{u}_{2,k} \\ \check{u}_{2,k} & -\check{u}_{1,k} & 0 & \check{u}_{3,k} \\ -\check{u}_{1,k} & -\check{u}_{2,k} & -\check{u}_{3,k} & 0 \end{bmatrix}.$$

The known value \mathbf{u}_k is

$$\mathbf{u}_k = \check{\mathbf{u}}_k + \mathbf{w}_k = \begin{bmatrix} 0.3 \sin((2\pi/12)kT_s) \\ 0.3 \sin((2\pi/12)kT_s - 6) \\ 0.3 \sin((2\pi/12)kT_s - 12) \end{bmatrix} + \mathbf{w}_k, \quad (5.17)$$

with $\|\mathbf{w}_k\|_\infty \leq 3.0 \times 10^{-3}$. The measurement is given by $\mathbf{y}_k = (\mathbf{C}(\mathbf{x}_k)\mathbf{r}^{[1]}, \mathbf{C}(\mathbf{x}_k)\mathbf{r}^{[2]}) + \mathbf{v}_k$, with $\mathbf{r}^{[1]} = [1 \ 0 \ 0]^T$, $\mathbf{r}^{[2]} = [0 \ 1 \ 0]^T$, $\|\mathbf{v}_k\|_\infty \leq 0.15$, and $\mathbf{C}(\mathbf{x}_k)$ is a rotation matrix defined by

$$\mathbf{C}(\mathbf{x}_k) \triangleq \begin{bmatrix} x_{1,k}^2 - x_{2,k}^2 - x_{3,k}^2 + x_{4,k}^2 & 2(x_{1,k}x_{2,k} + x_{3,k}x_{4,k}) \\ 2(x_{1,k}x_{2,k} - x_{3,k}x_{4,k}) & -x_{1,k}^2 + x_{2,k}^2 - x_{3,k}^2 + x_{4,k}^2 \\ 2(x_{1,k}x_{3,k} + x_{2,k}x_{4,k}) & 2(-x_{1,k}x_{4,k} + x_{2,k}x_{3,k}) \\ & 2(x_{1,k}x_{3,k} - x_{2,k}x_{4,k}) \\ & 2(x_{1,k}x_{4,k} + x_{2,k}x_{3,k}) \\ & -x_{1,k}^2 - x_{2,k}^2 + x_{3,k}^2 + x_{4,k}^2 \end{bmatrix}.$$

The sampling time is $T_s = 0.2\text{s}$, and the initial state belongs to the zonotope $X_0 = \{0.18\mathbf{I}_4, [0 \ 1 \ 0 \ 0]^T\}$. For the purpose of generating trajectories of (5.16), the initial state is $\mathbf{x}_0 = [0 \ 1 \ 0 \ 0]^T$.

In the following, for the sake of clarity we first illustrate the observation described in Remark 5.9 for the consistency step in a sub-example with $\mathbf{x}_k \in \mathbb{R}^2$, $\|\mathbf{x}_k\|_2^2 = 1$, and

$$X_0 = \left\{ \begin{bmatrix} 0.2 & 0.2 \\ 0 & 0.2 \end{bmatrix}, \begin{bmatrix} 1.3 \\ 0 \end{bmatrix} \right\}.$$

Note that in this case the set $\{\mathbf{I}_2, \mathbf{0}\}$ is a valid enclosure for the feasible state set of the invariant $\|\mathbf{x}_k\|_2^2 = 1$. Figure 5.4 shows the initial set X_0 and the enclosure \tilde{X}_0 obtained using Corollary 5.1 for the consistency step with $\hat{X}_0 = X_0$. Note that, although tightened, the resulting set is still very conservative. Figure 5.4 also shows the intersection $X_0 \cap \{\mathbf{I}_2, \mathbf{0}\}$, which is tighter than the previous result. Finally, we illustrate the enclosure \tilde{X}_0 obtained using Corollary 5.1 with $\hat{X}_0 = X_0 \cap \{\mathbf{I}_2, \mathbf{0}\}$, which is the least conservative result. This demonstrates the improved accuracy that can be achieved if an enclosure of the feasible state set is known *a priori*.

Figure 5.5 illustrates the radii of the enclosures \tilde{X}_k obtained for the trajectories of the

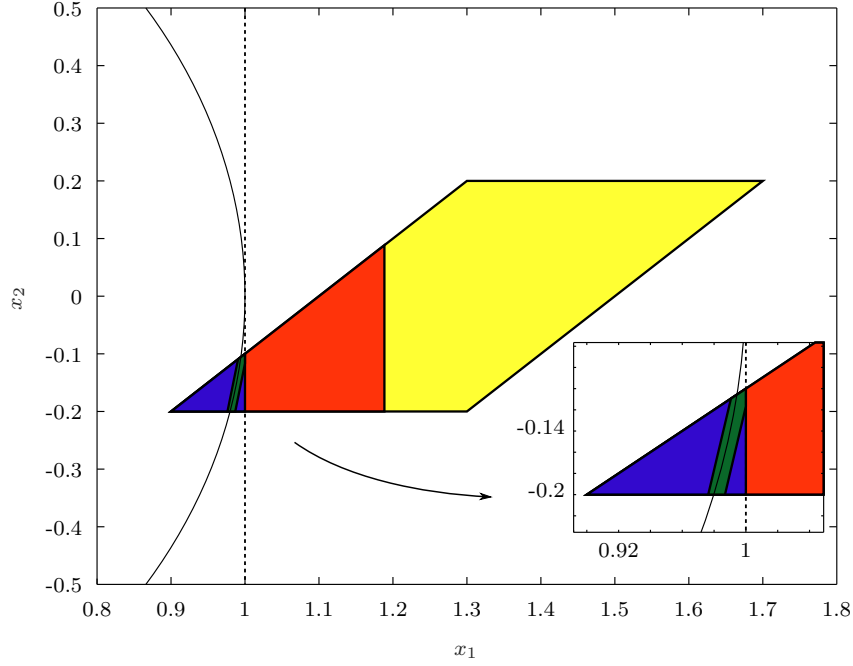


Figure 5.4: The zonotope X_0 (yellow), the enclosure \tilde{X}_0 obtained in the consistency step (5.5) using Corollary 5.1 with $\hat{X}_0 = X_0$ (red), the set $X_0 \cap \{\mathbf{I}_2, \mathbf{0}\}$ (blue), and the enclosure \tilde{X}_0 obtained in the consistency step (5.5) using Corollary 5.1 with $\hat{X}_0 = X_0 \cap \{\mathbf{I}_2, \mathbf{0}\}$ (green). The resulting enclosures contain each other according to the sequence above. The dashed line denotes the box $\{\mathbf{I}_2, \mathbf{0}\}$. The circle that describes the feasible state set of $\|\mathbf{x}_0\|_2 = 1$ is also depicted.

system (5.16) using ZMV-like and CZMV-like methods. We consider the enclosure $\{\mathbf{I}_4, \mathbf{0}\}$ of the feasible state set of the invariant $\|\mathbf{x}_k\|_2 = 1$. In this case, Z_w and Z_v are computed as in Remarks 5.2 and 5.6, respectively. The number of generators and constraints is limited to 12 and 5, respectively. Note that the zonotope methods were not able to provide useful enclosures for (5.16), i.e., the sizes of the enclosures increase with time and do not provide useful information, even when considering the intersection with $\{\mathbf{I}_4, \mathbf{0}\}$. Note that the enclosures provided by CZMV and CZMV+F also are not useful in this case, even though CZMV+F is much tighter than the others. On the other hand, CZMV+C and CZMV+FC both provided good enclosures with stable size, with the latter providing more accurate sets in the initial time steps, as expected. This demonstrates the advantage of including the consistency step (5.5) in state estimation using the mean value extension to take into account the invariant $\|\mathbf{x}_k\|_2 = 1$. In addition, note that the radii of the enclosures provided by CZMV+C and CZMV+FC are much smaller than the radius of $\{\mathbf{I}_4, \mathbf{0}\}$, showing that significant accuracy can be obtained by combining the state estimation procedure with the invariant, in comparison with using only the information available about the feasible state set.

Figure 5.6 shows the radii of the enclosures \tilde{X}_k obtained for the trajectories of (5.16) using ZFO-like and CZFO-like methods. Once again, the enclosures computed by zonotopes do not provide useful information since these increase with time, even when considering

the intersection with $\{\mathbf{I}_4, \mathbf{0}\}$. On the other hand, even CZFO provides tight enclosures for this example. This demonstrates that the first-order Taylor extension is able to provide significantly less conservative bounds than the mean value extension in this case, since the nonlinear measurements are polynomials of second order, and therefore the interval matrices $\mathbf{Q}^{[q]}$ in Proposition 5.4 are singletons. Nevertheless, CZFO+C and CZFO+FC both provide still sharper enclosures, with comparable sizes due to the limited complexity of the sets. To provide a comprehensive comparison between all of the methods, Table 5.3 shows the average radius ratio for this example (ARR, i.e., the ratio of the radius of the set provided by one method over the radius of the set provided by another method at k , averaged over all time steps), and Table 5.4 shows the average computational times per time step of each method. Note that, in contrast to the analogous state estimation algorithms for linear measurements in Chapter 4, the computational times of CZMV-like methods were competitive with ZMV-like methods, and CZFO-like methods with ZFO-like methods as well. The increased times of the zonotope methods arise from the iterative computation of strips based on interval analysis in Alamo et al. (2005) and the intersection with strips given in Bravo et al. (2006) to perform the update step. In this sense, using the mean value extension with constrained zonotopes, one can achieve about 93% less conservative bounds (CZMV+C-to-ZMV+F ARR of only 6.7%) in comparison to zonotopes, with a mild increase of 22.2% in the average execution time. On the other hand, using the first-order Taylor extension, one can achieve about 95% less conservative bounds (CZFO+C-to-ZFO+F ARR of 5.1%), with an increase of 10.5% in the average execution time. This demonstrates the joint accuracy and efficiency provided by the proposed methods based on constrained zonotopes. These ARR are highlighted in Table 5.3, and correspond to a comparison between the most accurate results obtained by ZMV-like and CZMV-like methods, and between ZFO-like and CZFO-like methods.

Lastly, note that the CZFO+C-to-CZFO ARR was of 69.64%, showing again the improved accuracy obtained by taking into account the invariant through the consistency step. In addition, CZFO-like methods provided better enclosures than CZMV-like methods in this example. Nevertheless, this comes with an important increase in computational time as shown in Table 5.4. This demonstrates that the choice between CZMV-like methods and CZFO-like methods for state estimation can provide a trade-off between accuracy and efficiency, and therefore will depend on the current application.

5.5 Final remarks

This chapter developed new approaches for set-valued state estimation of nonlinear discrete-time systems with nonlinear measurements and nonlinear invariants. The state trajectories were enclosed using the standard prediction-update algorithm with the addition of a new consistency step accounting for the nonlinear invariants. New methods were proposed for

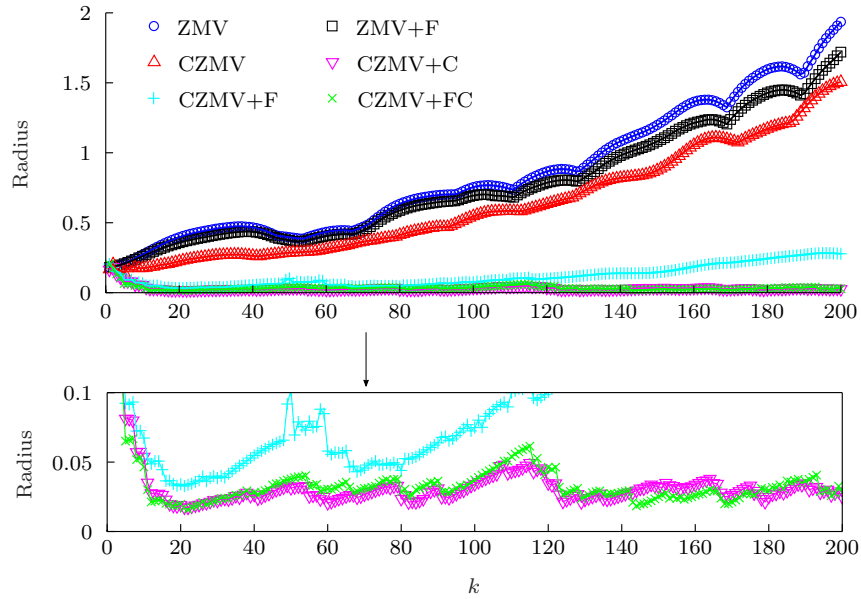


Figure 5.5: The radii of the estimated enclosures \tilde{X}_k for (5.16) obtained using ZMV-like and CZMV-like methods.

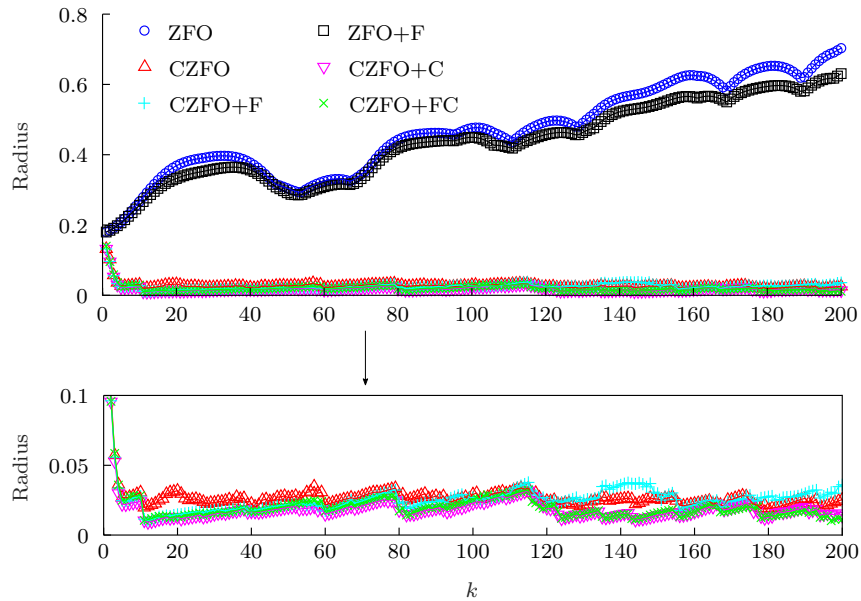


Figure 5.6: The radii of the estimated enclosures \tilde{X}_k for (5.16) obtained using ZFO-like and CZFO-like methods.

the update and consistency steps using generalized intersections of constrained zonotopes. In addition, our previous methods for the prediction step were generalized to allow the approximation points for the mean value and first-order Taylor extensions to lie in a larger region. Numerical results demonstrate that the methods proposed in this chapter can provide significantly tighter enclosures compared to existing zonotope methods. The improved accuracy is achieved with a mild increase in computational cost. Nevertheless, future work will seek to reduce the execution times, since these can be a major issue in many practical applications.

Table 5.3: Average radius ratio of the enclosures obtained by the state estimators (column per row) for the system (5.16).

\	ZMV	ZMV+F	CZMV	CZMV+C	CZMV+F	CZMV+FC
ZMV	1	0.9187	0.7193	0.0636	0.1491	0.0669
ZMV+F	1.0891	1	0.7834	0.0671	0.1606	0.0707
CZMV	1.4091	1.2946	1	0.0857	0.2024	0.0901
CZMV+C	29.2088	26.5557	21.4655	1	3.8897	1.0665
CZMV+F	8.0755	7.4009	5.6895	0.3745	1	0.4023
CZMV+FC	28.7622	26.1185	21.1725	0.9697	3.8402	1
\	ZFO	ZFO+F	CZFO	CZFO+C	CZFO+F	CZFO+FC
ZFO	1.6829	1.5384	1.2325	0.0849	0.2420	0.0898
ZFO+F	1.8040	1.6484	1.3211	0.0891	0.2581	0.0942
CZFO	34.1796	31.0484	25.3451	1.1983	4.6785	1.2562
CZFO+C	50.4008	45.7732	37.1612	1.7337	6.8068	1.8039
CZFO+F	33.4713	30.4709	24.4434	1.2814	4.5068	1.3433
CZFO+FC	50.2958	45.6286	37.2526	1.6647	6.8396	1.7317
\	ZFO	ZFO+F	CZFO	CZFO+C	CZFO+F	CZFO+FC
ZMV	0.6447	0.6057	0.0489	0.0358	0.0436	0.0385
ZMV+F	0.6988	0.6562	0.0521	0.0379	0.0465	0.0408
CZMV	0.9272	0.8701	0.0680	0.0484	0.0593	0.0523
CZMV+C	16.3264	15.2013	0.8974	0.6056	0.8472	0.6363
CZMV+F	5.2585	4.9195	0.3382	0.2289	0.2919	0.2460
CZMV+FC	15.8969	14.7785	0.8571	0.5744	0.8144	0.6018
ZFO	1	0.9367	0.0671	0.0487	0.0621	0.0519
ZFO+F	1.0683	1	0.0709	0.0512	0.0656	0.0546
CZFO	18.5748	17.2934	1	0.6964	0.9717	0.7290
CZFO+C	27.7376	25.7968	1.4945	1	1.4372	1.0499
CZFO+F	19.2573	17.9445	1.1000	0.7504	1	0.7929
CZFO+FC	27.1388	25.2164	1.4323	0.9617	1.3947	1

Table 5.4: Total and complexity reduction average execution times per time step of the state estimators for the system (5.16).

	ZMV	ZMV+F	CZMV	CZMV+C	CZMV+F	CZMV+FC
Total	0.4552 s	0.4610 s	0.4478 s	0.5635 s	0.4519 s	0.5942 s
Red.	0.37 ms	0.31 ms	10.6 ms	71.8 ms	19.8 ms	96.5 ms
	ZFO	ZFO+F	CZFO	CZFO+C	CZFO+F	CZFO+FC
Total	1.0403 s	1.0563 s	1.0907 s	1.1671 s	1.1009 s	1.3750 s
Red.	2.3 ms	2.2 ms	90.3 ms	0.1266 s	98.7 ms	0.3306 s

The next chapter develops a new method for set-based fault diagnosis of nonlinear discrete-time systems. Enclosures described by constrained zonotopes are employed in the passive fault detection and active fault isolation processes, based on the nonlinear state estimation methods introduced in this doctoral thesis.

6

Fault diagnosis of nonlinear systems

This chapter introduces a new approach for set-based active fault diagnosis (AFD) of a class of nonlinear discrete-time systems. The new enclosures based on constrained zonotopes proposed in the previous chapters are employed for passive fault detection of a class of nonlinear discrete-time systems. Moreover, an adaptation of the first-order Taylor extension (Section 4.2.3) is used to obtain an affine parametrization of the reachable sets, in the direction of the design of an optimal input for set-based active fault diagnosis of nonlinear systems.

6.1 Problem formulation

Consider a discrete-time system with nonlinear dynamics and linear measurement given by

$$\begin{aligned}\mathbf{x}_k &= \mathbf{f}^{[i]}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}), \\ \mathbf{y}_k &= \mathbf{s}^{[i]} + \mathbf{C}^{[i]}\mathbf{x}_k + \mathbf{D}_v^{[i]}\mathbf{v}_k,\end{aligned}\tag{6.1}$$

where $\mathbf{x}_k \in \mathbb{R}^n$ is the system state, $\mathbf{u}_k \in \mathbb{R}^{n_u}$ is the system input, $\mathbf{y}_k \in \mathbb{R}^{n_y}$ is the measured output, $\mathbf{w}_k \in \mathbb{R}^{n_w}$ is the process noise, and $\mathbf{v}_k \in \mathbb{R}^{n_v}$ is the measurement noise. In each time interval $[k-1, k]$, the system evolves according to one of the possible models in (6.1) indexed by $i \in \mathcal{M} = \{1, \dots, n_i\}$, with $i = 1$ denoting the nominal model. Measurement bias is modeled by $\mathbf{s}^{[i]}$. Uncertainties are assumed to be bounded by convex polytopic sets $\mathbf{w}_k \in W$, $\mathbf{v}_k \in V$, and the system input is bounded by an admissible set given by a convex

polytope $\mathbf{u}_k \in U$. Note that a convex polytope in H-rep can be converted to CG-rep using Proposition 3.2.

The objective of AFD is to find which model describes the process behaviour. In this chapter, the dynamics of such system are assumed to not change during the diagnosis procedure, i.e. the AFD is fast enough to avoid the switching between models. In this sense, an input sequence $(\mathbf{u}_k, \mathbf{u}_{k+1}, \dots, \mathbf{u}_{k+N-1})$ of minimal length N is designed such that the output sequence $(\mathbf{y}_{k+1}^{[i]}, \mathbf{y}_{k+2}^{[i]}, \dots, \mathbf{y}_{k+N}^{[i]})$ is consistent with only one $i \in \mathcal{M}$. If feasible, this problem may admit multiple solutions. For this reason, we introduce a cost function and select among the feasible input sequences the optimal one.

6.2 Passive fault detection

The set-based state estimation methods proposed in Section 4.2 can be effectively employed to perform guaranteed passive fault detection of nonlinear discrete-time systems as in (6.1). Let $\hat{X}_{k-1}^{[i]}$ denote the previous update set described by a constrained zonotope satisfying $\mathbf{x}_{k-1} \in \hat{X}_{k-1}^{[i]}$, and let $\bar{X}_k^{[i]}$ denote the prediction set described by a constrained zonotope satisfying

$$\bar{X}_k^{[i]} \supseteq \{\mathbf{f}^{[i]}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}) : \mathbf{x}_{k-1} \in \hat{X}_{k-1}^{[i]}, \mathbf{w}_{k-1} \in W\}. \quad (6.2)$$

Therefore, define the estimated measurement set $\hat{Y}_k^{[i]}$ by

$$\hat{Y}_k^{[i]} = \mathbf{s}^{[i]} \oplus \mathbf{C}^{[i]} \bar{X}_k^{[i]} \oplus \mathbf{D}_v^{[i]} V. \quad (6.3)$$

Assuming the i -th model to be active at $k = 0$, and $\mathbf{x}_0 \in \hat{X}_0^{[i]}$, a constrained zonotope satisfying (6.2) can be obtained by using any of the methods proposed in Section 4.2, including the CZIB approach if Assumption 4.1 in Section 4.2.1.1 is satisfied. The estimated measurement set in (6.3) can be computed through (3.8)–(3.9). Given the current output \mathbf{y}_k , if the inclusion

$$\mathbf{y}_k \in \hat{Y}_k^{[i]} \quad (6.4)$$

does not hold, then a fault has occurred at a given time $k_f \leq k$, i.e., the i -th model was not active in the time interval $[k_f, k]$. Therefore, passive fault detection can be effectively performed by checking if the inclusion (6.4) holds. This inclusion can be verified by solving an LP (Property 3.1).

6.3 Reachable sets

This section focuses on the computation of reachable sets for the nonlinear discrete-time system (6.1), aiming the design of an optimal input sequence for guaranteed active fault diagnosis.

6.3.1 Guaranteed enclosure

The following theorem is based on the first-order Taylor extension proposed in Chapter 4 (Section 4.2.3) and provides a simplified enclosure of the image of a nonlinear function of class \mathcal{C}^2 with domain given by constrained zonotopes. The obtained enclosure is also a constrained zonotope.

Theorem 6.1. Let $\boldsymbol{\eta} : \mathbb{R}^n \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_w} \rightarrow \mathbb{R}^n$ be of class \mathcal{C}^2 , and let $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{u} \in \mathbb{R}^{n_u}$ and $\mathbf{w} \in \mathbb{R}^{n_w}$ denote its arguments. Let $X \subset \mathbb{R}^n$, $U \subset \mathbb{R}^{n_u}$, and $W \subset \mathbb{R}^{n_w}$ be constrained zonotopes such that $\mathbf{x} \in X$, $\mathbf{u} \in U$, $\mathbf{w} \in W$, and choose any $\boldsymbol{\gamma}_x \in X$, $\boldsymbol{\gamma}_u \in U$, $\boldsymbol{\gamma}_w \in W$. For each $q = 1, 2, \dots, n$, define an interval matrix $\mathbf{Q}^{[q]} \in \mathbb{I}\mathbb{R}^{n+n_u+n_w \times n+n_u+n_w}$ satisfying $\mathbf{Q}^{[q]} \supseteq \mathbf{H}\boldsymbol{\eta}_q(X, U, W)$, and define an interval vector $\boldsymbol{\varphi} \in \mathbb{I}\mathbb{R}^n$ whose q -th component satisfies

$$\boldsymbol{\varphi}_q \supseteq \begin{bmatrix} X - \boldsymbol{\gamma}_x \\ \mathbf{u} - \boldsymbol{\gamma}_u \\ W - \boldsymbol{\gamma}_w \end{bmatrix}^T \mathbf{Q}^{[q]} \begin{bmatrix} X - \boldsymbol{\gamma}_x \\ \mathbf{u} - \boldsymbol{\gamma}_u \\ W - \boldsymbol{\gamma}_w \end{bmatrix}. \quad (6.5)$$

In addition, let $R \triangleq \{\text{diag}(\text{rad}(\boldsymbol{\varphi})), \text{mid}(\boldsymbol{\varphi})\}$. Then, for any $\mathbf{u} \in U$,

$$\begin{aligned} \boldsymbol{\eta}(X, \mathbf{u}, W) \subseteq & \boldsymbol{\eta}(\boldsymbol{\gamma}_x, \boldsymbol{\gamma}_u, \boldsymbol{\gamma}_w) \oplus \nabla_{\mathbf{x}}^T \boldsymbol{\eta}(\boldsymbol{\gamma}_x, \boldsymbol{\gamma}_u, \boldsymbol{\gamma}_w)(X - \boldsymbol{\gamma}_x) \oplus \nabla_{\mathbf{u}}^T \boldsymbol{\eta}(\boldsymbol{\gamma}_x, \boldsymbol{\gamma}_u, \boldsymbol{\gamma}_w)(\mathbf{u} - \boldsymbol{\gamma}_u) \\ & \oplus \nabla_{\mathbf{w}}^T \boldsymbol{\eta}(\boldsymbol{\gamma}_x, \boldsymbol{\gamma}_u, \boldsymbol{\gamma}_w)(W - \boldsymbol{\gamma}_w) \oplus R. \end{aligned} \quad (6.6)$$

Proof. Let $(\mathbf{x}, \mathbf{u}, \mathbf{w}) \in X \times U \times W$, $\mathbf{z} \triangleq (\mathbf{x}, \mathbf{u}, \mathbf{w})$, and $\boldsymbol{\gamma}_z \triangleq (\boldsymbol{\gamma}_x, \boldsymbol{\gamma}_u, \boldsymbol{\gamma}_w)$. From the Taylor's theorem around $\bar{\mathbf{z}}$, the q -th component of $\boldsymbol{\eta}(\mathbf{z})$ is given by (Combastel, 2005)

$$\eta_q(\mathbf{z}) = \eta_q(\bar{\mathbf{z}}) + \nabla_{\mathbf{z}}^T \eta_q(\bar{\mathbf{z}})(\mathbf{z} - \bar{\mathbf{z}}) + (\mathbf{z} - \bar{\mathbf{z}})^T \boldsymbol{\Gamma}^{[q]}(\mathbf{z} - \bar{\mathbf{z}}),$$

for one $\boldsymbol{\Gamma}^{[q]} \in \mathbf{Q}^{[q]}$. Noting that $(\mathbf{z} - \bar{\mathbf{z}})^T \boldsymbol{\Gamma}^{[q]}(\mathbf{z} - \bar{\mathbf{z}}) \in \boldsymbol{\varphi}_q$ for every $(\mathbf{x}, \mathbf{u}, \mathbf{w}) \in X \times U \times W$, $\boldsymbol{\Gamma}^{[q]} \in \mathbf{Q}^{[q]}$, and also $\nabla_{\mathbf{z}}^T \eta_q(\bar{\mathbf{z}})(\mathbf{z} - \bar{\mathbf{z}}) = \nabla_{\mathbf{x}}^T \eta_q(\boldsymbol{\gamma}_x, \boldsymbol{\gamma}_u, \boldsymbol{\gamma}_w)(\mathbf{x} - \boldsymbol{\gamma}_x) + \nabla_{\mathbf{u}}^T \eta_q(\boldsymbol{\gamma}_x, \boldsymbol{\gamma}_u, \boldsymbol{\gamma}_w)(\mathbf{u} - \boldsymbol{\gamma}_u) + \nabla_{\mathbf{w}}^T \eta_q(\boldsymbol{\gamma}_x, \boldsymbol{\gamma}_u, \boldsymbol{\gamma}_w)(\mathbf{w} - \boldsymbol{\gamma}_w)$, it follows that

$$\begin{aligned} \eta_q(\mathbf{x}, \mathbf{u}, \mathbf{w}) \in & \eta_q(\boldsymbol{\gamma}_x, \boldsymbol{\gamma}_u, \boldsymbol{\gamma}_w) \oplus \nabla_{\mathbf{x}}^T \eta_q(\boldsymbol{\gamma}_x, \boldsymbol{\gamma}_u, \boldsymbol{\gamma}_w)(X - \boldsymbol{\gamma}_x) \oplus \nabla_{\mathbf{u}}^T \eta_q(\boldsymbol{\gamma}_x, \boldsymbol{\gamma}_u, \boldsymbol{\gamma}_w)(\mathbf{u} - \boldsymbol{\gamma}_u) \\ & \oplus \nabla_{\mathbf{w}}^T \eta_q(\boldsymbol{\gamma}_x, \boldsymbol{\gamma}_u, \boldsymbol{\gamma}_w)(W - \boldsymbol{\gamma}_w) \oplus \boldsymbol{\varphi}_q, \end{aligned}$$

for every $\mathbf{x} \in X$, $\mathbf{w} \in W$. Therefore, $\boldsymbol{\gamma} = \text{mid}(\boldsymbol{\gamma}) \oplus \frac{1}{2} \text{diag}(\text{diam}(\boldsymbol{\gamma})) B_{\infty}^n$ implies that

$$\begin{aligned} \boldsymbol{\eta}(X, \mathbf{u}, W) \subseteq & \boldsymbol{\eta}(\boldsymbol{\gamma}_x, \boldsymbol{\gamma}_u, \boldsymbol{\gamma}_w) \oplus \nabla_{\mathbf{x}}^T \boldsymbol{\eta}(\boldsymbol{\gamma}_x, \boldsymbol{\gamma}_u, \boldsymbol{\gamma}_w)(X - \boldsymbol{\gamma}_x) \oplus \nabla_{\mathbf{u}}^T \boldsymbol{\eta}(\boldsymbol{\gamma}_x, \boldsymbol{\gamma}_u, \boldsymbol{\gamma}_w)(\mathbf{u} - \boldsymbol{\gamma}_u) \\ & \oplus \nabla_{\mathbf{w}}^T \boldsymbol{\eta}(\boldsymbol{\gamma}_x, \boldsymbol{\gamma}_u, \boldsymbol{\gamma}_w)(W - \boldsymbol{\gamma}_w) \oplus R. \end{aligned}$$

■

Theorem 6.1 provides a simplified constrained zonotope enclosing the trajectories of a nonlinear system. Nevertheless, the computation of the intervals $\boldsymbol{\varphi}_q$ is not trivial in the

general case, and particular solutions are proposed. To obtain an affine parameterization in \mathbf{u}_k of the reachable sets of (6.1), aiming the input design for active fault diagnosis, since the input \mathbf{u}_k is unknown *a priori*, the interval matrices $\mathbf{Q}^{[q]} = \square(\mathbf{H}\eta_q(X, U, W))$ are evaluated conservatively for the entire admissible set U in the reachable set computations. In the following, a new approach is proposed to compute reachable state and output sets for (6.1), yielding an affine parameterization in the input \mathbf{u}_k .

6.3.2 Affine parameterization of the reachable set

Consider constrained zonotopes $X_k \subset \mathbb{R}^n$, $W \subset \mathbb{R}^{n_w}$, $V \subset \mathbb{R}^{n_v}$. It is assumed that $\mathbf{x}_k \in X_k$, and $(\mathbf{w}_{k+m}, \mathbf{v}_{k+m}) \in W \times V$ for $m \in \{0, 1, \dots, N\}$. For each time k , choose $\gamma_{x,k} \in X_k$, $\gamma_{u,k} \in U$, $\gamma_{w,k} \in W$. Then, we consider intervals $\varphi_q^{[i]}$ computed using interval arithmetic, satisfying

$$\varphi_q^{[i]} \supseteq \begin{bmatrix} \square X_k - \gamma_{x,k} \\ \square U - \gamma_{u,k} \\ \square W - \gamma_{w,k} \end{bmatrix}^T \mathbf{Q}^{[q,i]} \begin{bmatrix} \square X_k - \gamma_{x,k} \\ \square U - \gamma_{u,k} \\ \square W - \gamma_{w,k} \end{bmatrix}, \quad (6.7)$$

where $\mathbf{Q}^{[q,i]} = \square \mathbf{H} \mathbf{f}_q^{[i]}(\square X, \square U, \square W)$. Note that $\varphi_q^{[i]}$ computed as in (6.7) is not a function of the input \mathbf{u}_k . Therefore, Theorem 6.1 immediately provides an enclosure $\Phi_{k+1}^{[i]} \supseteq \mathbf{f}^{[i]}(X_k, \mathbf{u}_k, W)$, given by

$$\begin{aligned} \Phi_{k+1}^{[i]} &= \mathbf{f}^{[i]}(\gamma_{x,k}, \gamma_{u,k}, \gamma_{w,k}) \oplus \nabla_x^T \mathbf{f}^{[i]}(\gamma_{x,k}, \gamma_{u,k}, \gamma_{w,k})(X_k - \gamma_{x,k}) \oplus \nabla_u^T \mathbf{f}^{[i]}(\gamma_{x,k}, \gamma_{u,k}, \gamma_{w,k})(\mathbf{u}_k - \gamma_{u,k}) \\ &\oplus \nabla_w^T \mathbf{f}^{[i]}(\gamma_{x,k}, \gamma_{u,k}, \gamma_{w,k})(W - \gamma_{w,k}) \oplus R_k^{[i]}(\gamma_{x,k}, \gamma_{u,k}, \gamma_{w,k}, X_k, U, W). \end{aligned} \quad (6.8)$$

Let $\mathbf{A}_k^{[i]} \triangleq \nabla_x^T \mathbf{f}^{[i]}(\gamma_{x,k}, \gamma_{u,k}, \gamma_{w,k})$, $\mathbf{B}_k^{[i]} \triangleq \nabla_u^T \mathbf{f}^{[i]}(\gamma_{x,k}, \gamma_{u,k}, \gamma_{w,k})$, $\mathbf{F}_k^{[i]} \triangleq \nabla_w^T \mathbf{f}^{[i]}(\gamma_{x,k}, \gamma_{u,k}, \gamma_{w,k})$, and $\mathbf{r}_k^{[i]} \triangleq \mathbf{f}^{[i]}(\gamma_{x,k}, \gamma_{u,k}, \gamma_{w,k}) - \mathbf{A}_k^{[i]} \gamma_{x,k}$, where dependences on $(\gamma_{x,k}, \gamma_{u,k}, \gamma_{w,k})$ were omitted for brevity. For clarity, we rewrite (6.8) as follows:

$$\Phi_{k+1}^{[i]}(\mathbf{u}_k) = \mathbf{r}_k^{[i]} \oplus \mathbf{A}_k^{[i]} X_k \oplus \mathbf{B}_k^{[i]}(\mathbf{u}_k - \gamma_{u,k}) \oplus \mathbf{F}_k^{[i]}(W - \gamma_{w,k}) \oplus R_k^{[i]}(X_k, U, W), \quad (6.9)$$

which is affine in \mathbf{u}_k (the dependency on the inputs is depicted henceforth).

Similarly to the approach in Scott et al. (2014), for each model $i \in \mathcal{M}$ we define the *reachable state and output sets* of (6.1) on the interval $[k+1, k+N]$ as

$$\vec{\Phi}_{k+1:k+N}^{[i]}(\vec{\mathbf{u}}) = \vec{\mathbf{r}}^{[i]} \oplus \vec{\mathbf{A}}^{[i]} X_k \oplus \vec{\mathbf{B}}^{[i]}(\vec{\mathbf{u}} - \vec{\gamma}_u) \oplus \vec{\mathbf{F}}^{[i]}(\vec{W} - \vec{\gamma}_w) \oplus \vec{R}^{[i]}(X_k, \vec{U}, \vec{W}), \quad (6.10)$$

$$\vec{\Psi}_{k+1:k+N}^{[i]}(\vec{\mathbf{u}}) = \vec{\mathbf{s}}^{[i]} \oplus \vec{\mathbf{C}}^{[i]} \vec{\Phi}_{k+1:k+N}^{[i]}(\vec{\mathbf{u}}) \oplus \vec{\mathbf{D}}_v^{[i]} \vec{V}, \quad (6.11)$$

where $\vec{U} = U \times U \times \dots \times U$, $\vec{W} = W \times W \times \dots \times W$, $\vec{V} = V \times V \times \dots \times V$, with $N-1$ Cartesian products, $\vec{\gamma}_u$ and $\vec{\gamma}_w$ are sequences satisfying $\vec{\gamma}_u \in \vec{U}$, $\vec{\gamma}_w \in \vec{W}$, respectively, and $\vec{\mathbf{u}} = (\mathbf{u}_k, \mathbf{u}_{k+1}, \dots, \mathbf{u}_{k+N-1})$. The variables $\vec{\mathbf{r}}^{[i]}$, $\vec{\mathbf{A}}^{[i]}$, $\vec{\mathbf{B}}^{[i]}$, $\vec{\mathbf{F}}^{[i]}$, $\vec{R}^{[i]}$, $\vec{\mathbf{s}}^{[i]}$, $\vec{\mathbf{C}}^{[i]}$, $\vec{\mathbf{D}}_v^{[i]}$, are defined

according to

$$\vec{\mathbf{r}}^{[i]} = \begin{bmatrix} \mathbf{r}_k^{[i]} \\ \mathbf{A}_{k+1}^{[i]} \mathbf{r}_k^{[i]} + \mathbf{r}_{k+1}^{[i]} \\ \vdots \\ \sum_{n=0}^{N-2} \left(\prod_{j=n+1}^{N-1} \mathbf{A}_{k+j}^{[i]} \right) \mathbf{r}_{k+n}^{[i]} + \mathbf{r}_{k+N-1}^{[i]} \end{bmatrix}, \quad \vec{\mathbf{A}}^{[i]} = \begin{bmatrix} \mathbf{A}_k^{[i]} \\ \mathbf{A}_{k+1}^{[i]} \mathbf{A}_k^{[i]} \\ \vdots \\ \prod_{j=0}^{N-1} \mathbf{A}_{k+j}^{[i]} \end{bmatrix}, \quad (6.12)$$

$$\vec{\mathbf{B}}^{[i]} = \begin{bmatrix} \mathbf{B}_k^{[i]} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{A}_{k+1}^{[i]} \mathbf{B}_k^{[i]} & \mathbf{B}_{k+1}^{[i]} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \left(\prod_{j=1}^{N-1} \mathbf{A}_{k+j}^{[i]} \right) \mathbf{B}_k^{[i]} & \left(\prod_{j=2}^{N-1} \mathbf{A}_{k+j}^{[i]} \right) \mathbf{B}_{k+1}^{[i]} & \cdots & \mathbf{B}_{k+N-1}^{[i]} \end{bmatrix}, \quad (6.13)$$

$$\vec{\mathbf{F}}^{[i]} = \begin{bmatrix} \mathbf{F}_k^{[i]} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{A}_{k+1}^{[i]} \mathbf{F}_k^{[i]} & \mathbf{F}_{k+1}^{[i]} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \left(\prod_{j=1}^{N-1} \mathbf{A}_{k+j}^{[i]} \right) \mathbf{F}_k^{[i]} & \left(\prod_{j=2}^{N-1} \mathbf{A}_{k+j}^{[i]} \right) \mathbf{F}_{k+1}^{[i]} & \cdots & \mathbf{F}_{k+N-1}^{[i]} \end{bmatrix}, \quad (6.14)$$

$$\vec{R}^{[i]} = R_k^{[i]}(X_k, U, W) \times R_{k+1}^{[i]}(\Phi_{k+1}^{[i]}(\vec{\mathbf{u}}), U, W) \times \cdots \times R_{k+N-1}^{[i]}(\Phi_{k+N-1}^{[i]}(\vec{\mathbf{u}}), U, W), \quad (6.15)$$

$$\vec{\mathbf{s}}^{[i]} = \begin{bmatrix} \mathbf{s}_k^{[i]} \\ \mathbf{s}_{k+1}^{[i]} \\ \vdots \\ \mathbf{s}_{k+N-1}^{[i]} \end{bmatrix}, \quad \vec{\mathbf{C}}^{[i]} = \begin{bmatrix} \mathbf{C}^{[i]} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{C}^{[i]} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{C}^{[i]} \end{bmatrix}, \quad \vec{\mathbf{D}}_v^{[i]} = \begin{bmatrix} \mathbf{D}_v^{[i]} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_v^{[i]} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{D}_v^{[i]} \end{bmatrix}, \quad (6.16)$$

where the matrix products are defined from the left, i.e. $\prod_{j=1}^3 \mathbf{A}_{k+j}^{[i]} \triangleq \mathbf{A}_{k+3}^{[i]} \mathbf{A}_{k+2}^{[i]} \mathbf{A}_{k+1}^{[i]}$.

Note that the set $\vec{R}^{[i]}$ is a function of $\Phi_{k+j}^{[i]}(\vec{\mathbf{u}})$, $j = 1, 2, \dots, N-1$, and therefore also of the input sequence $\vec{\mathbf{u}}$ (see (6.9)). To circumvent this dependency of $\vec{R}^{[i]}$ on the input sequence, we propose the use of conservative enclosures $\underline{\Phi}_{k+j}^{[i]}$ satisfying $\underline{\Phi}_{k+j}^{[i]} \supseteq \Phi_{k+j}^{[i]}(\vec{\mathbf{u}})$ in (6.15). These are computed recursively by

$$\underline{\Phi}_{k+j}^{[i]} = \mathbf{r}_{k+j-1}^{[i]} \oplus \mathbf{A}_{k+j-1}^{[i]} \underline{\Phi}_{k+j-1}^{[i]} \oplus \mathbf{B}_{k+j-1}^{[i]} (U - \gamma_{u,k+j-1}) \oplus \mathbf{F}_{k+j-1}^{[i]} (W - \gamma_{w,k+j-1}) \oplus R_{k+j-1}^{[i]}(\underline{\Phi}_{k+j-1}^{[i]}, U, W), \quad (6.17)$$

where $\underline{\Phi}_k^{[i]} \triangleq X_k$. Therefore, we define the respective conservative reachable state and output sets satisfying $\vec{\underline{\Phi}}_{k+1:k+N}^{[i]}(\vec{\mathbf{u}}) \supseteq \vec{\Phi}_{k+1:k+N}^{[i]}(\vec{\mathbf{u}})$, and $\vec{\underline{\Psi}}_{k+1:k+N}^{[i]}(\vec{\mathbf{u}}) \supseteq \vec{\Psi}_{k+1:k+N}^{[i]}(\vec{\mathbf{u}})$. These are computed by

$$\vec{\underline{\Phi}}_{k+1:k+N}^{[i]}(\vec{\mathbf{u}}) = \vec{\mathbf{r}}^{[i]} \oplus \vec{\mathbf{A}}^{[i]} X_k \oplus \vec{\mathbf{B}}^{[i]}(\vec{\mathbf{u}} - \vec{\gamma}_u) \oplus \vec{\mathbf{F}}^{[i]}(\vec{W} - \vec{\gamma}_w) \oplus \vec{R}^{[i]}(X_k, \vec{U}, \vec{W}), \quad (6.18)$$

$$\vec{\underline{\Psi}}_{k+1:k+N}^{[i]}(\vec{\mathbf{u}}) = \vec{\mathbf{s}}^{[i]} \oplus \vec{\mathbf{C}}^{[i]} \vec{\underline{\Phi}}_{k+1:k+N}^{[i]}(\vec{\mathbf{u}}) \oplus \vec{\mathbf{D}}_v^{[i]} \vec{V}, \quad (6.19)$$

where

$$\vec{R}^{[i]} \triangleq R_k^{[i]}(X_k, U, W) \times R_{k+1}^{[i]}(\Phi_{k+1}^{[i]}, U, W) \times \dots \times R_{k+N-1}^{[i]}(\Phi_{k+N-1}^{[i]}, U, W).$$

Note that the proposed conservative enclosure (6.17) is not a function of the input \mathbf{u}_{k+j-1} , $j = 1, 2, \dots, N$. Consequently, (6.18)–(6.19) are all affine in the input sequence $\vec{\mathbf{u}}$ as desired.

6.4 Input design for active fault diagnosis

This section proposes a new method for input design aiming at active fault diagnosis of the nonlinear discrete-time system (6.1). The method is based on the open-loop fault diagnosis approach proposed in Scott et al. (2014) for linear systems using zonotopes. This method is then extended using constrained zonotopes based on the approach proposed in Raimondo et al. (2016).

6.4.1 Zonotope method

Let $X_k \subset \mathbb{R}^n$, $W \subset \mathbb{R}^{nw}$, and $V \subset \mathbb{R}^{nv}$ be zonotopes. In this subsection we assume that U is described by a zonotope, but with its H-rep also being available. We first present the required lemma below.

Lemma 6.1. (Dobkin et al., 1993; Scott et al., 2014) Let $Z = \{\mathbf{G}_z, \mathbf{a}_z + \mathbf{b}_z\}$ and $Y = \{\mathbf{G}_y, \mathbf{a}_y + \mathbf{b}_y\}$. Then $Z \cap Y = \emptyset \iff \mathbf{a}_y - \mathbf{a}_z \notin \{\mathbf{G}_z, \mathbf{b}_z\} \oplus \{\mathbf{G}_y, -\mathbf{b}_y\}$.

The following definition is similar to Definition 1 in Scott et al. (2014), but it differs by the fact that the conservative reachable output sets formulated by (6.19) are used in place of the exact enclosures.

Definition 6.1. An input sequence $\vec{\mathbf{u}}$ is said to be a *separating input* on the time interval $[k+1, k+N]$ if

$$\vec{\Psi}_{k+1:k+N}^{[i]}(\vec{\mathbf{u}}) \cap \vec{\Psi}_{k+1:k+N}^{[j]}(\vec{\mathbf{u}}) = \emptyset, \quad (6.20)$$

for every $i, j \in \mathcal{M}$, $i \neq j$.

Let $\vec{\mathbf{u}} \in \vec{U}$ be an input sequence defined in the time interval $[k, k+N-1]$, and consider the output sequence $\vec{\mathbf{y}} = (\mathbf{y}_{k+1}, \mathbf{y}_{k+2}, \dots, \mathbf{y}_{k+N})$ observed by the injection of $\vec{\mathbf{u}}$ into the nonlinear system (6.1)¹. If $\vec{\mathbf{u}}$ is a separating input as in Definition 6.1, then the inclusion

$$\vec{\mathbf{y}} \in \vec{\Psi}_{k+1:k+N}^{[i]}(\vec{\mathbf{u}}) \quad (6.21)$$

is satisfied only for the model $i \in \mathcal{M}$ that is active in $[k+1, k+N]$. Therefore, the injection of a separating input in (6.1) allows guaranteed active fault diagnosis in the time interval $[k+1, k+N]$ by verifying the inclusion (6.21) for every $i \in \mathcal{M}$. Since $\vec{\Psi}_{k+1:k+N}^{[i]}(\vec{\mathbf{u}})$ is a

¹The input and output sequences are displaced by one time step since system (6.1) is causal.

zonotope, this inclusion can be easily verified by solving an LP (similarly to Property 3.1). The following theorem is an extension of Theorem 3 in Scott et al. (2014) to the approach presented in this section.

Theorem 6.2. Let $\vec{\Psi}_{k+1:k+N}^{[i]}(\vec{\mathbf{u}}) = \{\mathbf{G}_{k+1:k+N}^{\Psi}(i), \mathbf{c}_{k+1:k+N}^{\Psi}(\vec{\mathbf{u}}, i)\}$. An input sequence $\vec{\mathbf{u}}$ is a separating input iff

$$\vec{\mathbf{N}}(i, j)\vec{\mathbf{u}} \notin \vec{Z}(i, j), \quad (6.22)$$

for all $i, j \in \mathcal{M}, i \neq j$, where $\vec{\mathbf{N}}(i, j) = \vec{\mathbf{C}}^{[j]}\vec{\mathbf{B}}^{[j]} - \vec{\mathbf{C}}^{[i]}\vec{\mathbf{B}}^{[i]}$, and $\vec{Z}(i, j) = \{\mathbf{G}_{k+1:k+N}^{\Psi}(i) \mathbf{G}_{k+1:k+N}^{\Psi}(j), \mathbf{c}_{k+1:k+N}^{\Psi}(\vec{\gamma}_u, i) - \mathbf{c}_{k+1:k+N}^{\Psi}(\vec{\gamma}_u, j) + \vec{\mathbf{N}}(i, j)\vec{\gamma}_u\}$.

Proof. The relation below follows immediately from (6.18):

$$\vec{\Phi}_{k+1:k+N}^{[i]}(\vec{\mathbf{u}}) = \vec{\Phi}_{k+1:k+N}^{[i]}(\vec{\gamma}_u) \oplus \vec{\mathbf{B}}_u^{[i]}(\vec{\mathbf{u}} - \vec{\gamma}_u),$$

which means that the center of $\vec{\Phi}_{k+1:k+N}^{[i]}(\vec{\mathbf{u}})$ is displaced from the center of $\vec{\Phi}_{k+1:k+N}^{[i]}(\vec{\gamma}_u)$ by $\vec{\mathbf{B}}^{[i]}(\vec{\mathbf{u}} - \vec{\gamma}_u)$, and the respective generator matrix is unaffected. Consequently, it follows from (6.19) that

$$\vec{\Psi}_{k+1:k+N}^{[i]}(\vec{\mathbf{u}}) = \vec{\Psi}_{k+1:k+N}^{[i]}(\vec{\gamma}_u) \oplus \vec{\mathbf{C}}^{[i]}\vec{\mathbf{B}}^{[i]}(\vec{\mathbf{u}} - \vec{\gamma}_u).$$

Therefore, $\mathbf{c}_{k+1:k+N}^{\Psi}(\vec{\mathbf{u}}, i) = \mathbf{c}_{k+1:k+N}^{\Psi}(\vec{\gamma}_u, i) + \vec{\mathbf{C}}^{[i]}\vec{\mathbf{B}}^{[i]}(\vec{\mathbf{u}} - \vec{\gamma}_u)$. Finally, if $\vec{\mathbf{N}}(i, j)$ and $\vec{Z}(i, j)$ are defined as in the statement of the theorem, then Lemma 6.1 implies that (6.20) holds for every $i, j \in \mathcal{M}, i \neq j$, iff (6.22) holds for every $i, j \in \mathcal{M}, i \neq j$. ■

Let n_q denote the number of all possible combinations between $i, j \in \mathcal{M}, i \neq j$, and define $\mathbf{N}^{[q]} \triangleq \mathbf{N}(i, j)$, $\vec{Z}^{[q]} \triangleq \vec{Z}(i, j)$, $q \in \{1, 2, \dots, n_q\}$. Lemma 6.2 is proposed in Scott et al. (2014) and can be used to verify if a given input sequence $\vec{\mathbf{u}}$ is a separating input according to Theorem 6.2 by means of the solution of n_q LPs.

Lemma 6.2. (Scott et al., 2014) Let $\vec{Z}^{[q]} = \{\mathbf{G}^{[q]}, \mathbf{c}^{[q]}\}$. For each $\vec{\mathbf{u}} \in \vec{U}$ and $q \in \{1, 2, \dots, n_q\}$, define

$$\begin{aligned} \hat{\delta}^{[q]}(\vec{\mathbf{u}}) &= \min_{\delta^{[q]}, \xi^{[q]}} \delta^{[q]} \\ \text{s.t. } \vec{\mathbf{N}}^{[q]}\vec{\mathbf{u}} &= \mathbf{G}^{[q]}\xi^{[q]} + \mathbf{c}^{[q]} \\ \|\xi\|_{\infty} &\leq 1 + \delta^{[q]}. \end{aligned}$$

Then $\vec{\mathbf{N}}^{[q]}\vec{\mathbf{u}} \notin \vec{Z}^{[q]} \iff \hat{\delta}^{[q]}(\vec{\mathbf{u}}) > 0$.

Proof. See Lemma 4 in Scott et al. (2014). ■

Based on Theorem 6.2, we therefore consider the design of an optimal separating input by solving the optimization problem

$$\min_{\vec{\mathbf{u}}} J(\vec{\mathbf{u}}) \quad (6.23)$$

$$\begin{aligned} \text{s.t. } \vec{\mathbf{u}} &\in \vec{U}, \\ \vec{\mathbf{N}}^{[q]} \vec{\mathbf{u}} &\notin \vec{Z}^{[q]}, \quad \forall q = 1, 2, \dots, n_q, \end{aligned}$$

where $J(\vec{\mathbf{u}})$ is a cost function chosen to minimize any harmful effects caused to the system when injecting the separating input. For simplicity we define $J(\vec{\mathbf{u}}) = \|\vec{\mathbf{u}}\|_{\mathbf{R}}^2 = \sum_{j=0}^{N-1} \mathbf{u}_{k+j}^T \mathbf{R} \mathbf{u}_{k+j}$, with $\vec{\mathbf{R}} \triangleq \text{blkdiag}(\mathbf{R}, \mathbf{R}, \dots, \mathbf{R})$.

As in [Scott et al. \(2014\)](#), the optimization problem (6.23) is a bilevel program and can be rewritten as a mixed-integer quadratic program (MIQP) as follows. Define the *minimum separation threshold* $\varepsilon > 0$, then the optimization problem (6.23) can be written as

$$\begin{aligned} \min_{\vec{\mathbf{u}}} J(\vec{\mathbf{u}}) & \tag{6.24} \\ \text{s.t. } \vec{\mathbf{u}} &\in \vec{U}, \\ \varepsilon &\leq \hat{\delta}^{[q]} \leq \hat{\delta}_m^{[q]}, \quad \forall q = 1, 2, \dots, n_q, \end{aligned}$$

with $\hat{\delta}^{[q]}$ defined as in [Lemma 6.2](#), and $\hat{\delta}_m^{[q]}$ is given by $\hat{\delta}_m^{[q]} = \max_{\vec{\mathbf{u}} \in \vec{U}} \hat{\delta}^{[q]}(\vec{\mathbf{u}})$ (see [Section 5.1](#) in [Scott et al. \(2016\)](#) for details in the computation of $\hat{\delta}_m^{[q]}$). By replacing the inner optimization programs in (6.24) by their necessary and sufficient conditions for optimality ([Proposition 3.4.1](#) in [Bertsekas \(1999\)](#)), and by introducing binary variables to avoid non-convex constraints (see [Section 5](#) in [Scott et al. \(2016\)](#)), the optimization problem (6.24) can be rewritten as the single level program

$$\begin{aligned} \min_{\vec{\mathbf{u}}, \delta^{[q]}, \boldsymbol{\xi}^{[q]}, \boldsymbol{\lambda}^{[q]}, \boldsymbol{\mu}_1^{[q]}, \boldsymbol{\mu}_2^{[q]}, \mathbf{p}_1^{[q]}, \mathbf{p}_2^{[q]}} J(\vec{\mathbf{u}}) & \tag{6.25} \\ \text{s.t. } \vec{\mathbf{u}} &\in \vec{U}, \\ \varepsilon &\leq \hat{\delta}^{[q]} \leq \hat{\delta}_m^{[q]}, \\ \vec{\mathbf{N}}^{[q]} \vec{\mathbf{u}} &= \mathbf{G}^{[q]} \boldsymbol{\xi}^{[q]} + \mathbf{c}^{[q]}, \\ \|\boldsymbol{\xi}^{[q]}\|_{\infty} &\leq 1 + \delta^{[q]}, \\ (\mathbf{G}^{[q]})^T \boldsymbol{\lambda}^{[q]} &= \boldsymbol{\mu}_1^{[q]} - \boldsymbol{\mu}_2^{[q]}, \\ 1 &= (\boldsymbol{\mu}_1^{[q]} + \boldsymbol{\mu}_2^{[q]})^T \mathbf{1}, \\ \mathbf{0} &\leq \boldsymbol{\mu}_1^{[q]}, \boldsymbol{\mu}_2^{[q]}, \\ \mathbf{p}_1^{[q]}, \mathbf{p}_2^{[q]} &\in \{0, 1\}^{n_g}, \\ \boldsymbol{\mu}_1^{[q]} &\leq \mathbf{p}_1^{[q]}, \\ \boldsymbol{\mu}_2^{[q]} &\leq \mathbf{p}_2^{[q]}, \\ \xi_j^{[q]} - 1 - \delta^{[q]} &\in [-2(1 + \hat{\delta}_m^{[q]})(1 - p_{1,j}^{[q]}), 0], \\ \xi_j^{[q]} + 1 + \delta^{[q]} &\in [0, 2(1 + \hat{\delta}_m^{[q]})(1 - p_{2,j}^{[q]})], \end{aligned} \left. \vphantom{\begin{aligned} \min_{\vec{\mathbf{u}}, \delta^{[q]}, \boldsymbol{\xi}^{[q]}, \boldsymbol{\lambda}^{[q]}, \boldsymbol{\mu}_1^{[q]}, \boldsymbol{\mu}_2^{[q]}, \mathbf{p}_1^{[q]}, \mathbf{p}_2^{[q]}} J(\vec{\mathbf{u}}) \right.} \right\} \forall q = 1, 2, \dots, n_q,$$

which is a MIQP, with $j = 1, 2, \dots, n_g^{[q]}$. The constraints $\vec{\mathbf{N}}^{[q]} \vec{\mathbf{u}} = \mathbf{G}^{[q]} \boldsymbol{\xi}^{[q]} + \mathbf{c}^{[q]}$ and $\|\boldsymbol{\xi}^{[q]}\|_\infty \leq 1 + \delta^{[q]}$ come from Lemma 6.2; $\varepsilon \leq \hat{\delta}^{[q]} \leq \hat{\delta}_m^{[q]}$ guarantees the boundedness of (6.25); $(\mathbf{G}^{[q]})^T \boldsymbol{\lambda}^{[q]} = \boldsymbol{\mu}_1^{[q]} - \boldsymbol{\mu}_2^{[q]}$, $1 = (\boldsymbol{\mu}_1^{[q]} + \boldsymbol{\mu}_2^{[q]})^T \mathbf{1}$, and $\mathbf{0} \leq \boldsymbol{\mu}_1^{[q]}, \boldsymbol{\mu}_2^{[q]}$ are a result from Karush-Kuhn-Tucker conditions; and $\mathbf{p}_1^{[q]}, \mathbf{p}_2^{[q]} \in \{0, 1\}^{n_g^{[q]}}$, $\boldsymbol{\mu}_1^{[q]} \leq \mathbf{p}_1^{[q]}$, $\boldsymbol{\mu}_2^{[q]} \leq \mathbf{p}_2^{[q]}$, $\xi_j^{[q]} - 1 - \delta^{[q]} \in [-2(1 + \hat{\delta}_m^{[q]})(1 - p_{1,j}^{[q]}), 0]$, $\xi_j^{[q]} + 1 + \delta^{[q]} \in [0, 2(1 + \hat{\delta}_m^{[q]})(1 - p_{2,j}^{[q]})]$ are mixed-integer reformulations of non-convex complementarity constraints (Scott et al., 2014).

6.4.2 Constrained zonotope method

Let $X_k \subset \mathbb{R}^n$, $W \subset \mathbb{R}^{n_w}$, and $V \subset \mathbb{R}^{n_v}$ now be constrained zonotopes. Similarly to the open-loop method in Raimondo et al. (2016), the approach presented in the previous subsection can be extended to constrained zonotopes as follows.

Theorem 6.3. Let $\vec{\Psi}_{k+1:k+N}^{[i]}(\vec{\mathbf{u}}) = \{\mathbf{G}_{k+1:k+N}^\Psi(i), \mathbf{c}_{k+1:k+N}^\Psi(\vec{\mathbf{u}}, i), \mathbf{A}_{k+1:k+N}^\Psi(i), \mathbf{b}_{k+1:k+N}^\Psi(i)\}$. An input sequence $\vec{\mathbf{u}}$ is a separating input iff

$$\vec{\mathbf{N}}(i, j) \vec{\mathbf{u}} \notin \vec{Z}(i, j), \quad (6.26)$$

for all $i, j \in \mathcal{M}$, $i \neq j$, where $\vec{\mathbf{N}}(i, j) = \vec{\mathbf{C}}^{[j]} \vec{\mathbf{B}}^{[j]} - \vec{\mathbf{C}}^{[i]} \vec{\mathbf{B}}^{[i]}$, and

$$\vec{Z}(i, j) = \left\{ \left[\mathbf{G}_{k+1:k+N}^\Psi(i) - \mathbf{G}_{k+1:k+N}^\Psi(j), \mathbf{c}_{k+1:k+N}^\Psi(\vec{\gamma}_u, i) - \mathbf{c}_{k+1:k+N}^\Psi(\vec{\gamma}_u, j) + \vec{\mathbf{N}}(i, j) \vec{\gamma}_u, \right. \right. \\ \left. \left. \begin{bmatrix} \mathbf{A}_{k+1:k+N}^\Psi(i) & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_{k+1:k+N}^\Psi(j) \end{bmatrix}, \begin{bmatrix} \mathbf{b}_{k+1:k+N}^\Psi(i) \\ \mathbf{b}_{k+1:k+N}^\Psi(j) \end{bmatrix} \right\}.$$

Proof. The following is similar to the proof of Theorem 1 in Raimondo et al. (2016). From the generalized intersection (3.10) and recalling that $\mathbf{c}_{k+1:k+N}^\Psi(\vec{\mathbf{u}}, i) = \mathbf{c}_{k+1:k+N}^\Psi(\vec{\gamma}_u, i) + \mathbf{C}^{[i]} \mathbf{B}^{[i]}(\vec{\mathbf{u}} - \vec{\gamma}_u)$, it holds that

$$\vec{\Psi}_{k+1:k+N}^{[i]}(\vec{\mathbf{u}}) \cap \vec{\Psi}_{k+1:k+N}^{[j]}(\vec{\mathbf{u}}) = \emptyset \iff \nexists \boldsymbol{\xi} : \|\boldsymbol{\xi}\|_\infty \leq 1, \\ \begin{bmatrix} \mathbf{A}_{k+1:k+N}^\Psi(i) & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_{k+1:k+N}^\Psi(j) \\ \mathbf{G}_{k+1:k+N}^\Psi(i) & -\mathbf{G}_{k+1:k+N}^\Psi(j) \end{bmatrix} \boldsymbol{\xi} = \begin{bmatrix} \mathbf{b}_{k+1:k+N}^\Psi(i) \\ \mathbf{b}_{k+1:k+N}^\Psi(j) \\ \mathbf{c}_{k+1:k+N}^\Psi(\vec{\gamma}_u, j) - \mathbf{c}_{k+1:k+N}^\Psi(\vec{\gamma}_u, i) + \vec{\mathbf{N}}(i, j)(\vec{\mathbf{u}} - \vec{\gamma}_u) \end{bmatrix}.$$

The relation above is equivalent to

$$\vec{\Psi}_{k+1:k+N}^{[i]}(\vec{\mathbf{u}}) \cap \vec{\Psi}_{k+1:k+N}^{[j]}(\vec{\mathbf{u}}) = \emptyset \iff \\ \vec{\mathbf{N}}(i, j) \vec{\mathbf{u}} \notin \left\{ \left[\mathbf{G}_{k+1:k+N}^\Psi(i) - \mathbf{G}_{k+1:k+N}^\Psi(j), \mathbf{c}_{k+1:k+N}^\Psi(\vec{\gamma}_u, i) - \mathbf{c}_{k+1:k+N}^\Psi(\vec{\gamma}_u, j) + \vec{\mathbf{N}}(i, j) \vec{\gamma}_u, \right. \right. \\ \left. \left. \begin{bmatrix} \mathbf{A}_{k+1:k+N}^\Psi(i) & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_{k+1:k+N}^\Psi(j) \end{bmatrix}, \begin{bmatrix} \mathbf{b}_{k+1:k+N}^\Psi(i) \\ \mathbf{b}_{k+1:k+N}^\Psi(j) \end{bmatrix} \right\},$$

which right hand side is (6.26). ■

We now reformulate Theorem 6.3 in terms of lifted zonotopes.

Corollary 6.1. Let $\vec{Z}(i, j) = \{\mathbf{G}^Z(i, j), \mathbf{c}^Z(i, j), \mathbf{A}^Z(i, j), \mathbf{b}^Z(i, j)\}$ be a constrained zonotope as defined in Theorem 6.3. An input sequence $\vec{\mathbf{u}}$ is a separating input iff

$$\begin{bmatrix} \vec{\mathbf{N}}(i, j) \\ \mathbf{0} \end{bmatrix} \vec{\mathbf{u}} \notin \vec{Z}^+(i, j) = \left\{ \begin{bmatrix} \mathbf{G}^Z(i, j) \\ \mathbf{A}^Z(i, j) \end{bmatrix}, \begin{bmatrix} \mathbf{c}^Z(i, j) \\ -\mathbf{b}^Z(i, j) \end{bmatrix} \right\}, \quad (6.27)$$

for all $i, j \in \mathcal{M}$, $i \neq j$.

Proof. See Corollary 2 in Raimondo et al. (2016). ■

Let $\vec{\mathbf{M}}^{[q]} = [\vec{\mathbf{N}}^T(i, j) \ \mathbf{0}]^T$, $\vec{Z}^{+[q]} = \{\mathbf{G}^{+[q]}, \mathbf{c}^{+[q]}\} = \vec{Z}^+(i, j)$, $q \in \{1, 2, \dots, n_q\}$. Since $\vec{Z}^{+[q]}$ is a zonotope, the relation (6.27) can be verified by solving an LP similar to Lemma 6.2. We then consider the design of the optimal separating input by solving

$$\begin{aligned} \min_{\vec{\mathbf{u}}} J(\vec{\mathbf{u}}) & \quad (6.28) \\ \text{s.t. } \vec{\mathbf{u}} & \in \vec{U}, \\ \vec{\mathbf{M}}^{[q]} \vec{\mathbf{u}} & \notin \vec{Z}^{+[q]}, \quad \forall q = 1, 2, \dots, n_q, \end{aligned}$$

where for simplicity $J(\vec{\mathbf{u}}) = \|\vec{\mathbf{u}}\|_{\mathbf{R}}^2 = \sum_{j=0}^{N-1} \mathbf{u}_{k+j}^T \mathbf{R} \mathbf{u}_{k+j}$, $\vec{\mathbf{R}} = \text{blkdiag}(\mathbf{R}, \mathbf{R}, \dots, \mathbf{R})$. Analogously to the previous subsection, the optimization problem (6.28) is a bilevel program, but can be reformulated as an MIQP given by

$$\begin{aligned} \min_{\vec{\mathbf{u}}, \delta^{[q]}, \boldsymbol{\xi}^{[q]}, \boldsymbol{\lambda}^{[q]}, \boldsymbol{\mu}_1^{[q]}, \boldsymbol{\mu}_2^{[q]}, \mathbf{p}_1^{[q]}, \mathbf{p}_2^{[q]}} J(\vec{\mathbf{u}}) & \quad (6.29) \\ \text{s.t. } \vec{\mathbf{u}} & \in \vec{U}, \\ \left. \begin{aligned} \varepsilon & \leq \hat{\delta}^{[q]} \leq \hat{\delta}_m^{[q]}, \\ \vec{\mathbf{M}}^{[q]} \vec{\mathbf{u}} & = \mathbf{G}^{+[q]} \boldsymbol{\xi}^{[q]} + \mathbf{c}^{+[q]}, \\ \|\boldsymbol{\xi}^{[q]}\|_{\infty} & \leq 1 + \delta^{[q]}, \\ (\mathbf{G}^{+[q]})^T \boldsymbol{\lambda}^{[q]} & = \boldsymbol{\mu}_1^{[q]} - \boldsymbol{\mu}_2^{[q]}, \\ 1 & = (\boldsymbol{\mu}_1^{[q]} + \boldsymbol{\mu}_2^{[q]})^T \mathbf{1}, \\ \mathbf{0} & \leq \boldsymbol{\mu}_1^{[q]}, \boldsymbol{\mu}_2^{[q]}, \\ \mathbf{p}_1^{[q]}, \mathbf{p}_2^{[q]} & \in \{0, 1\}^{n_g^{[q]}}, \\ \boldsymbol{\mu}_1^{[q]} & \leq \mathbf{p}_1^{[q]}, \\ \boldsymbol{\mu}_2^{[q]} & \leq \mathbf{p}_2^{[q]}, \\ \xi_j^{[q]} - 1 - \delta^{[q]} & \in [-2(1 + \hat{\delta}_m^{[q]})(1 - p_{1,j}^{[q]}), 0], \\ \xi_j^{[q]} + 1 + \delta^{[q]} & \in [0, 2(1 + \hat{\delta}_m^{[q]})(1 - p_{2,j}^{[q]})], \end{aligned} \right\} \quad \forall q = 1, 2, \dots, n_q, \end{aligned}$$

with $j = 1, 2, \dots, n_g^{[q]}$. The meaning of each constraint in (6.29) is similar to the ones from the zonotope method presented in the previous section.

Remark 6.1. The MIQPs (6.25) and (6.29) may admit multiple optimal solutions due to the non-convex nature of mixed-integer problems. In such cases, any of the resulting optimal input sequences is considered valid for active fault diagnosis.

6.5 Numerical example

This section demonstrates the performance of the proposed active-fault diagnosis approach for nonlinear discrete-time systems described by (6.1) in a numerical example. The experiment consists of fault diagnosis of a planar robotic manipulator composed of two revolute joints (Figure 6.1) which is subject to actuator faults.

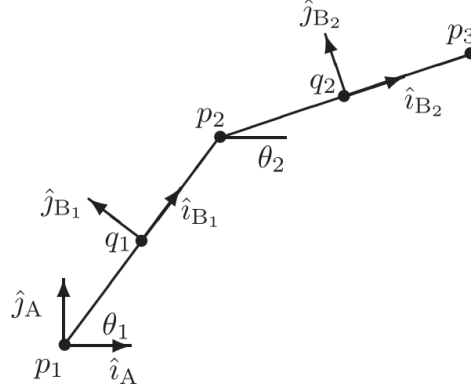


Figure 6.1: The planar robotic manipulator (D'Amato et al., 2011).

Let θ_1 and θ_2 denote the angles of the first and second revolute joint, respectively, and let u denote the torque applied to the first joint. The second joint is passive (i.e., it is not actuated). The equations of motion that describe the dynamics of the robotic manipulator are given by (D'Amato et al., 2011)

$$\begin{aligned} \kappa^{[q]}u &= \left(\frac{1}{3}m_1l_1^2 + m_2l_1^2\right)\ddot{\theta}_1 + \frac{1}{2}m_2l_1l_2\sin(\theta_1 - \theta_2)\ddot{\theta}_2 + \frac{1}{2}m_2l_1l_2\cos(\theta_1 - \theta_2)\dot{\theta}_2 \\ &\quad + (k_1 + k_2)\theta_1 - k_2\theta_2 + (c_1 + c_2)\dot{\theta}_1 - c_2\dot{\theta}_2, \\ 0 &= \left(\frac{1}{3}m_2l_2^2\right)\ddot{\theta}_2 - \frac{1}{2}m_2l_1l_2\sin(\theta_1 - \theta_2)\dot{\theta}_1^2 + \frac{1}{2}m_2l_1l_2\cos(\theta_1 - \theta_2)\ddot{\theta}_1 \\ &\quad - k_2\theta_1 + k_2\theta_2 - c_2\dot{\theta}_1 + c_2\dot{\theta}_2, \end{aligned} \tag{6.30}$$

where $m_1 = 2$ kg, $m_2 = 1$ kg, $l_1 = 3$ m, $l_2 = 2$ m, $k_1 = 7$ N·m/rad, $k_2 = 5$ N·m/rad, $c_1 = 10$ kg·m²/rad, $c_2 = 1$ kg·m²/rad. The actuator faults are modeled by the parameter $\kappa^{[q]}$. Two scenarios are considered, the nominal scenario in which $\kappa^{[1]} = 1$, and the faulty one given

by $\kappa^{[2]} = 0$. By defining the variables

$$\begin{aligned} \mathbf{q} &\triangleq \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix}, \\ \mathbf{R}(\mathbf{q}) &\triangleq \begin{bmatrix} \frac{1}{3}m_1l_1^2 + m_2l_1^2 & \frac{1}{2}m_2l_1l_2 \cos(\theta_1 - \theta_2) \\ \frac{1}{2}m_2l_1l_2 \cos(\theta_1 - \theta_2) & \frac{1}{3}m_2l_2^2 \end{bmatrix}, \quad \mathbf{P}^{[i]} \triangleq \begin{bmatrix} \kappa^{[i]} \\ 0 \end{bmatrix}, \\ \mathbf{S}(\mathbf{q}, \dot{\mathbf{q}}) &\triangleq \begin{bmatrix} \frac{1}{2}m_2l_1l_2 \sin(\theta_1 - \theta_2)\dot{\theta}_2^2 + (k_1 + k_2)\theta_1 - k_2\theta_2 + (c_1 + c_2)\dot{\theta}_1 - c_2\dot{\theta}_2 \\ -\frac{1}{2}m_2l_1l_2 \sin(\theta_1 - \theta_2)\dot{\theta}_1^2 - k_2\theta_1 + k_2\theta_2 - c_2\dot{\theta}_1 + c_2\dot{\theta}_2 \end{bmatrix}, \end{aligned}$$

the dynamic equations (6.30) can be written in compact form as $\mathbf{R}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{S}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{P}^{[i]}u$. In addition, by defining the system state $\mathbf{x} = [\theta_1 \ \theta_2 \ \dot{\theta}_1 \ \dot{\theta}_2]^T$, it leads to the following state-space representation of the robotic manipulator dynamics:

$$\dot{\mathbf{x}} = \begin{bmatrix} x_3 \\ x_4 \\ \mathbf{R}(x_1, x_2)^{-1}(-\mathbf{S}(x_1, x_2, x_3, x_4) + \mathbf{P}^{[i]}u) \end{bmatrix}. \quad (6.31)$$

We consider that θ_1 and θ_2 are measured by sensors. Then, the corresponding measurement equation is given by

$$\begin{bmatrix} y_{1,k} \\ y_{2,k} \end{bmatrix} = \begin{bmatrix} x_{1,k} \\ x_{2,k} \end{bmatrix} + \begin{bmatrix} v_{1,k} \\ v_{2,k} \end{bmatrix}, \quad (6.32)$$

with $\|\mathbf{v}_k\|_\infty \leq 0.1$. In this example, the nonlinear equations (6.31) are discretized through Euler approximation with sampling time 0.01 s. Moreover, the initial state \mathbf{x}_0 is bounded by the zonotope

$$X_0 = \{0.01 \cdot \mathbf{I}_4, [\pi/2 \ \pi/2 \ 0 \ 0]^T\}.$$

For fault diagnosis of the system (6.31)–(6.32), we consider the design of an input sequence with length $N = 5$, and choose $\vec{\gamma}_u = \mathbf{0}$, $\vec{\gamma}_w = \mathbf{0}$. The cost function in (6.25) is chosen as $J(\vec{\mathbf{u}}) = \sum_{j=0}^{N-1} ru_j^2$, with $r = 1$. The admissible inputs must satisfy $|u_k| \leq 100$ N·m for $k \in [0, N - 1]$. Moreover, $\gamma_{x,0} = [\pi/2 \ \pi/2 \ 0 \ 0]^T$, and $\gamma_{x,k}$ is obtained by propagating $(\gamma_{x,k-1}, \gamma_{u,k-1}, \gamma_{w,k-1})$ through the discretized version of the nonlinear dynamics (6.31) for each model, with $k = 1, 2, \dots, N - 1$. The number of generators of the zonotopes $\vec{Z}^{[q]}$ is limited to 20 using the generator reduction algorithm described by Method 3.2 (Section 3.3). The minimum separation threshold was chosen as $\varepsilon = 1.0 \cdot 10^{-8}$.

The MIQP (6.25) was solved using MATLAB 9.1 and GUROBI 8.0.1², which resulted in the optimal separating input

$$\vec{\mathbf{u}} = (-100, -91.52, -73.88, -45.12, 0) \text{ N}\cdot\text{m}. \quad (6.33)$$

²<http://www.gurobi.com>

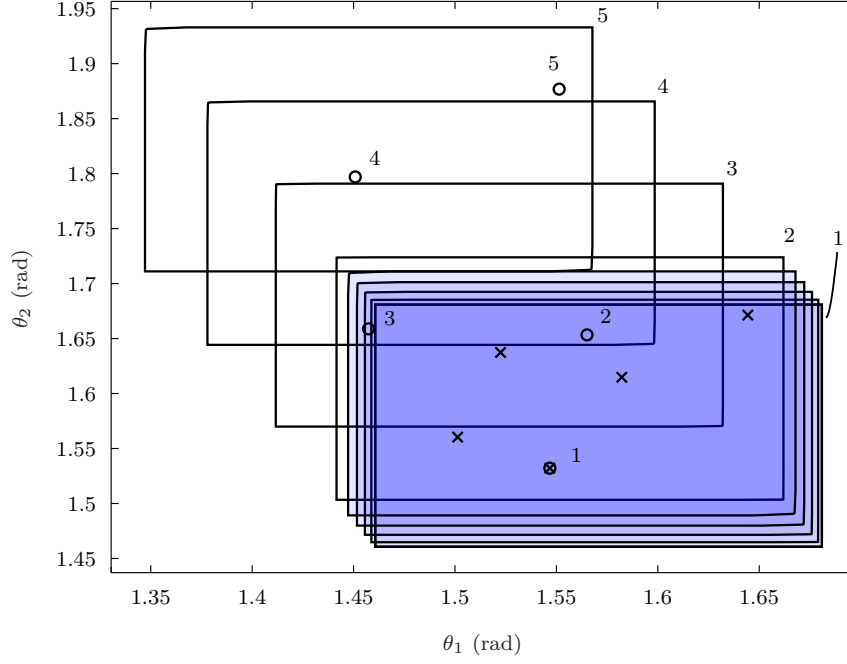


Figure 6.2: The reachable output sets for models 1 (colorless sets) and 2 (blue sets) along with the output \vec{y} observed by the injection of \vec{u} into models 1 (o) and 2 (x) for $k = 1, 2, \dots, 5$. For clarity, numbers note the corresponding time step k for the output sequence $\vec{y}^{[1]}$ observed from model 1 and its output reachable sets.

This input sequence was obtained with execution time³ 0.035 s. Figure 6.2 shows the output reachable sets for models 1 and 2 (nominal and faulty, respectively), along with the respective output sequences observed by the injection of (6.33) into (6.31) for $k = 1, 2, \dots, 5$, which are denoted here by $\vec{y}^{[1]}$ and $\vec{y}^{[2]}$. Note that, as expected the following relations were satisfied:

$$\vec{y}^{[1]} \in \vec{\Psi}_{1:5}^{[1]}(\vec{u}), \vec{y}^{[2]} \in \vec{\Psi}_{1:5}^{[2]}(\vec{u}), \vec{y}^{[1]} \notin \vec{\Psi}_{1:5}^{[2]}(\vec{u}), \vec{y}^{[2]} \notin \vec{\Psi}_{1:5}^{[1]}(\vec{u}).$$

Therefore, the active model in (6.31) was guaranteed to be identified by the injection of the separating input (6.33), allowing robust fault diagnosis of the robotic manipulator subject to actuator faults using the proposed active fault diagnosis method.

A second experiment, consisting of fault diagnosis of quadrotor UAV which is subject to actuator and sensor faults, can be found in Chapter 9. In this experiment, the method developed in Section 6.4.2 based on constrained zonotopes is also demonstrated.

6.6 Final remarks

This chapter introduced a new approach for set-based active fault diagnosis of a class of nonlinear discrete-time systems with nonlinear dynamics and linear measurement. A simple and effective method for passive fault detection was developed based on the state estimation methods proposed in Chapter 4. A modified version of the first-order Taylor

³Laptop with 8GB RAM and Intel Core i7 4510U 3.1 GHz processor.

extension (Theorem 6.1) was proposed to compute affine parametrizations of the reachable state and output sets of the nonlinear discrete-time systems as a function of the system input. By being affine in the system input, this allowed the development of a nonlinear set-based active fault diagnosis method based on the design of optimal separating inputs, by extending the linear open-loop approach in Scott et al. (2014) using zonotopes, and the method in Raimondo et al. (2016) using constrained zonotopes. The resulting problem consisted in solving an MIQP to find the optimal input sequence that allows guaranteed fault diagnosis in a given time horizon.

The performance of the proposed active fault diagnosis method was demonstrated in a numerical experiment with a robotic manipulator subject to actuator faults. The diagnosis of the current possible faults was guaranteed in the experiment by injecting the designed optimal input sequence into the system.

The next chapter introduces new methods based on constrained zonotopes for set-based state estimation and active fault diagnosis of discrete-time linear descriptor systems.

7

Descriptor systems

This chapter presents new methods based on constrained zonotopes for set-valued state estimation and active fault diagnosis of linear descriptor systems (LDS). In contrast to the other sets representations mentioned in Chapter 3, linear static constraints on the state variables, typical of descriptor systems, can be directly incorporated in the mathematical description of constrained zonotopes. Thanks to this feature, set-based methods based on CZs can provide less conservative enclosures. In addition, this chapter proposes a new representation for unbounded sets based on zonotopes, which allows to develop methods for state estimation and AFD of linear descriptor systems without assuming the knowledge of an admissible set that encloses all the possible trajectories of the system. Therefore, the proposed methods lead to more accurate results in state estimation in comparison to existing methods based on zonotopes, without requiring rank assumptions on the structure of the descriptor system and with a fair trade-off between accuracy and efficiency. The superiority of the proposed approaches with respect to zonotope-based methods is highlighted in numerical examples. The content concerning the new methods based on constrained zonotopes proposed in this chapter was published in [Rego et al. \(2020b\)](#).

7.1 Problem formulation

7.1.1 State estimation

Consider a linear discrete-time descriptor system with time k , state $\mathbf{x}_k \in \mathbb{R}^n$, input $\mathbf{u}_k \in \mathbb{R}^{n_u}$, process uncertainty $\mathbf{w}_k \in \mathbb{R}^{n_w}$, measured output $\mathbf{y}_k \in \mathbb{R}^{n_y}$, and measurement uncertainty $\mathbf{v}_k \in \mathbb{R}^{n_v}$. In each time interval $[k-1, k]$, $k = 1, 2, \dots$, the system evolves according to the model

$$\begin{aligned} \mathbf{E}\mathbf{x}_k &= \mathbf{A}\mathbf{x}_{k-1} + \mathbf{B}\mathbf{u}_{k-1} + \mathbf{B}_w\mathbf{w}_{k-1}, \\ \mathbf{y}_k &= \mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k + \mathbf{D}_v\mathbf{v}_k, \end{aligned} \quad (7.1)$$

with $\mathbf{E} \in \mathbb{R}^{n \times n}$, $\mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{B} \in \mathbb{R}^{n \times n_u}$, $\mathbf{B}_w \in \mathbb{R}^{n \times n_w}$, $\mathbf{C} \in \mathbb{R}^{n_y \times n}$, $\mathbf{D} \in \mathbb{R}^{n_y \times n_u}$, and $\mathbf{D}_v \in \mathbb{R}^{n_y \times n_v}$. In the following, it is assumed that $\text{rank}(\mathbf{E}) \leq n$. Note that when \mathbf{E} is singular, one has $n - \text{rank}(\mathbf{E})$ purely static constraints. It is assumed that the initial state $\mathbf{x}_0 \in X_0$ and $(\mathbf{w}_k, \mathbf{v}_k) \in W \times V$ for all $k \geq 0$, where X_0 , W and V are known convex polytopic sets. Moreover, the initial condition $(\mathbf{x}_0, \mathbf{u}_0, \mathbf{w}_0, \mathbf{v}_0)$ is assumed to be feasible, i.e., consistent with the static relations in (7.1), and the output \mathbf{y}_0 is computed as $\mathbf{y}_0 = \mathbf{C}\mathbf{x}_0 + \mathbf{D}\mathbf{u}_0 + \mathbf{D}_v\mathbf{v}_0$. For any $k \geq 0$, the objective of state estimation is to approximate the feasible trajectories of (7.1) as accurately as possible by a guaranteed enclosure \tilde{X}_k .

7.1.2 Fault diagnosis

Consider a linear discrete-time descriptor system whose dynamics obeys one of possible n_m known models

$$\begin{aligned} \mathbf{E}^{[i]}\mathbf{x}_k^{[i]} &= \mathbf{A}^{[i]}\mathbf{x}_{k-1}^{[i]} + \mathbf{B}^{[i]}\mathbf{u}_{k-1} + \mathbf{B}_w^{[i]}\mathbf{w}_{k-1}, \\ \mathbf{y}_k^{[i]} &= \mathbf{C}^{[i]}\mathbf{x}_k^{[i]} + \mathbf{D}^{[i]}\mathbf{u}_k + \mathbf{D}_v^{[i]}\mathbf{v}_k, \end{aligned} \quad (7.2)$$

for $k \geq 1$, with $\mathbf{E}^{[i]} \in \mathbb{R}^{n \times n}$, $\mathbf{A}^{[i]} \in \mathbb{R}^{n \times n}$, $\mathbf{B}^{[i]} \in \mathbb{R}^{n \times n_u}$, $\mathbf{B}_w^{[i]} \in \mathbb{R}^{n \times n_w}$, $\mathbf{C}^{[i]} \in \mathbb{R}^{n_y \times n}$, $\mathbf{D}^{[i]} \in \mathbb{R}^{n_y \times n_u}$, and $\mathbf{D}_v^{[i]} \in \mathbb{R}^{n_y \times n_v}$, $i \in \mathcal{M} \triangleq \{1, 2, \dots, n_m\}$. Also, $\text{rank}(\mathbf{E}^{[i]}) \leq n$, and let $\mathbf{x}_0^{[i]} \in X_0$, $(\mathbf{w}_k, \mathbf{v}_k) \in W \times V$, and $\mathbf{u}_k \in U$, with X_0 , W , V and U being known convex polytopic sets.

Similar to chapter 6, in this chapter the goal of AFD is to find which model describes the process behaviour. The dynamics of the system are assumed to not change during the diagnosis procedure, i.e. the AFD is fast enough to avoid the switching between models. In this sense, a sequence $(\mathbf{u}_0, \mathbf{u}_1, \dots, \mathbf{u}_N)$ of minimal length N is designed such that any output $\mathbf{y}_N^{[i]}$ is consistent with only one $i \in \mathcal{M}$. If feasible, this problem may admit multiple solutions. For this reason, we introduce a cost function and select among the feasible input sequences the optimal one.

7.1.3 Motivational example

In this subsection, we present a motivational example illustrating that, in the general case, the feasible sets of the linear discrete-time descriptor system (7.1) cannot be described

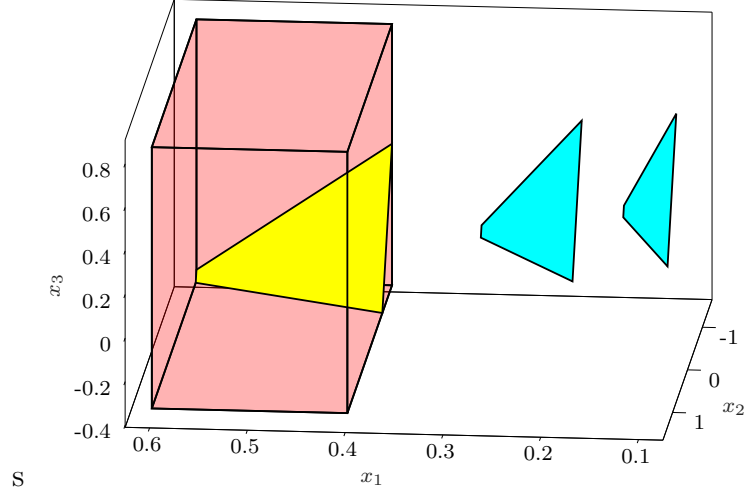


Figure 7.1: The feasible sets for the system (7.3): the initial set X_0 (red), the set S_0 (yellow), and S_k (cyan) for $k = 1, 2$.

exactly by zonotopes even when X_0 , W , and V are all zonotopes. For the sake of simplicity, consider

$$\mathbf{E}\mathbf{x}_k = \mathbf{A}\mathbf{x}_{k-1}, \quad (7.3)$$

with

$$\mathbf{E} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{A} = \begin{bmatrix} 0.5 & 0 & 0 \\ 0.8 & 0.95 & 0 \\ 0.3 & 0.1 & 0.1 \end{bmatrix}.$$

Let $\mathbf{x}_0 \in X_0$, where X_0 is a zonotope given by

$$X_0 = \left\{ \left[\begin{array}{ccc} 0.1 & 0 & 0 \\ 0 & 1.5 & 0 \\ 0 & 0 & 0.6 \end{array} \right], \left[\begin{array}{c} 0.5 \\ 0 \\ 0.25 \end{array} \right] \right\}. \quad (7.4)$$

Since the last row of \mathbf{E} is zero, the dynamics (7.3) further constrains \mathbf{x}_{k-1} for $k \geq 1$. In particular, \mathbf{x}_0 should be consistent with both (7.4) and the constraints in (7.3). Therefore, $\mathbf{x}_0 \in S_0$ with $S_0 = \{\mathbf{x} \in X_0 : (7.3) \text{ holds for } k = 1\}$. Similarly, we define $S_k = \{\mathbf{x} \in \mathbb{R}^n : (7.3) \text{ holds, } \mathbf{E}\mathbf{x} = \mathbf{A}\mathbf{x}', \mathbf{x}' \in S_{k-1}\}$, with $k \geq 1$. Figure 7.1 shows the zonotope X_0 and the convex polytopes S_k for $k = 0, 1, 2$. These latter are computed exactly using the MPT toolbox (Herceg et al., 2013). As it can be noticed, the sets S_k are not symmetric, and therefore cannot be described accurately by zonotopes. Consequently, in the general case the use of zonotopes for reachability analysis, set-based estimation, and fault diagnosis of descriptor systems could result very conservative. The next sections present methods to address these problems using constrained zonotopes, as well as a new proposal for representing unbounded sets.

7.2 Set-based state estimation of LDS using constrained zonotopes

This section presents a new method for set-based state estimation of system (7.1). Consider an initial condition X_0 and an input \mathbf{u}_k with $k \geq 0$, let

$$\hat{X}_0 = \{\mathbf{x} \in X_0 : \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u}_0 + \mathbf{D}_v\mathbf{v} = \mathbf{y}_0, \mathbf{v} \in V\}. \quad (7.5)$$

The linear prediction step (3.17) must be reformulated when \mathbf{E} is not invertible. We first consider the following assumption.

Assumption 7.1. There exists a known CZ $X_a = \{\mathbf{G}_a, \mathbf{c}_a, \mathbf{A}_a, \mathbf{b}_a\} \subset \mathbb{R}^n$ such that $\mathbf{x}_k \in X_a$ for all $k \geq 0$.

Remark 7.1. Even though X_a can be arbitrarily large, Assumption 7.1 is restrictive in the sense that the descriptor system (7.1) must be stable. Unstable systems are out of the scope of this section and are considered later in Section 7.5.

The set estimation in Section 3.5 relied on (3.8)–(3.10) to compute the steps (3.17)–(3.18). In order to extend this method to the case of descriptor systems, when computing \bar{X}_k it is necessary to take into account the static constraints arising from the possible singularity of \mathbf{E} . The proposed method is based on singular value decomposition (SVD). Let $\mathbf{E} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$, where \mathbf{U} and \mathbf{V} are invertible by construction. Since \mathbf{E} is square, then $\mathbf{\Sigma}$ is also square. Without loss of generality, let $\mathbf{\Sigma}$ be arranged as $\mathbf{\Sigma} = \text{blkdiag}(\tilde{\mathbf{\Sigma}}, \mathbf{0})$, where $\tilde{\mathbf{\Sigma}} \in \mathbb{R}^{n_z \times n_z}$ is diagonal with all the $n_z = \text{rank}(\mathbf{E})$ nonzero singular values of \mathbf{E} . Moreover, let $\mathbf{z}_k = (\tilde{\mathbf{z}}_k, \check{\mathbf{z}}_k) = \mathbf{T}^{-1}\mathbf{x}_k$, $\tilde{\mathbf{z}}_k \in \mathbb{R}^{n_z}$, $\check{\mathbf{z}}_k \in \mathbb{R}^{n-n_z}$,

$$\begin{aligned} \begin{bmatrix} \tilde{\mathbf{A}} \\ \check{\mathbf{A}} \end{bmatrix} &= \begin{bmatrix} \tilde{\mathbf{\Sigma}}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \mathbf{U}^{-1}\mathbf{A}\mathbf{T}, \\ \begin{bmatrix} \tilde{\mathbf{B}} \\ \check{\mathbf{B}} \end{bmatrix} &= \begin{bmatrix} \tilde{\mathbf{\Sigma}}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \mathbf{U}^{-1}\mathbf{B}, \quad \begin{bmatrix} \tilde{\mathbf{B}}_w \\ \check{\mathbf{B}}_w \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{\Sigma}}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \mathbf{U}^{-1}\mathbf{B}_w, \end{aligned} \quad (7.6)$$

with $\mathbf{T} = (\mathbf{V}^T)^{-1}$, $\tilde{\mathbf{A}} \in \mathbb{R}^{n_z \times n}$, $\tilde{\mathbf{B}} \in \mathbb{R}^{n_z \times n_u}$, and $\tilde{\mathbf{B}}_w \in \mathbb{R}^{n_z \times n_w}$. Then, system (7.1) can be rewritten with decoupled dynamics given by (Jonckheere, 1988)

$$\tilde{\mathbf{z}}_k = \tilde{\mathbf{A}}\mathbf{z}_{k-1} + \tilde{\mathbf{B}}\mathbf{u}_{k-1} + \tilde{\mathbf{B}}_w\mathbf{w}_{k-1}, \quad (7.7a)$$

$$\mathbf{0} = \check{\mathbf{A}}\mathbf{z}_{k-1} + \check{\mathbf{B}}\mathbf{u}_{k-1} + \check{\mathbf{B}}_w\mathbf{w}_{k-1}, \quad (7.7b)$$

$$\mathbf{y}_k = \mathbf{C}\mathbf{T}\mathbf{z}_k + \mathbf{D}\mathbf{u}_k + \mathbf{D}_v\mathbf{v}_k. \quad (7.7c)$$

Consider $W = \{\mathbf{G}_w, \mathbf{c}_w, \mathbf{A}_w, \mathbf{b}_w\}$, and \hat{X}_0 given by (7.5). From (7.7b), the state \mathbf{z}_0 must satisfy $\check{\mathbf{A}}\mathbf{z}_0 + \check{\mathbf{B}}\mathbf{u}_0 + \check{\mathbf{B}}_w\mathbf{w}_0 = \mathbf{0}$. This constraint can be incorporated in the CG-rep of

the initial set \hat{Z}_0 as follows. Let $\mathbf{T}^{-1}\hat{X}_0 \triangleq \{\mathbf{G}_0, \mathbf{c}_0, \mathbf{A}_0, \mathbf{b}_0\}$. Then, with a slight abuse of notation¹, define $\hat{Z}_0 \triangleq \{\hat{\mathbf{G}}_0, \hat{\mathbf{c}}_0, \hat{\mathbf{A}}_0, \hat{\mathbf{b}}_0\}$, with $\hat{\mathbf{G}}_0 = [\mathbf{G}_0 \ \mathbf{0}]$, $\hat{\mathbf{c}}_0 = \mathbf{c}_0$,

$$\hat{\mathbf{A}}_0 = \begin{bmatrix} \mathbf{A}_0 & \mathbf{0} \\ \check{\mathbf{A}}\mathbf{G}_0 & \check{\mathbf{B}}_w\mathbf{G}_w \end{bmatrix}, \quad \hat{\mathbf{b}}_0 = \begin{bmatrix} \mathbf{b}_0 \\ -\check{\mathbf{A}}\mathbf{c}_0 - \check{\mathbf{B}}_w\mathbf{c}_w - \check{\mathbf{B}}\mathbf{u}_0 \end{bmatrix}.$$

Note that the extra columns in $\hat{\mathbf{G}}_0$ and $\hat{\mathbf{A}}_0$ come from $\mathbf{w}_0 \in W$. Having defined \hat{Z}_0 , for state estimation purposes, the static relation (7.7b) can be shifted forward to time k without loss of information. By doing so, from (7.7), the variables $\check{\mathbf{z}}_k$ are fully determined by (7.7a), while $\check{\mathbf{z}}_k$ are obtained a posteriori by $\check{\mathbf{A}}\mathbf{z}_k + \check{\mathbf{B}}\mathbf{u}_k + \check{\mathbf{B}}_w\mathbf{w}_k = \mathbf{0}$.

Consider the set $Z_a = \mathbf{T}^{-1}X_a = \{\mathbf{T}^{-1}\mathbf{G}_a, \mathbf{T}^{-1}\mathbf{c}_a, \mathbf{A}_a, \mathbf{b}_a\}$, and let $\mathbf{T}^{-1}\mathbf{c}_a = [\check{\mathbf{c}}_a^T \ \check{\mathbf{c}}_a^T]^T$, $\mathbf{T}^{-1}\mathbf{G}_a = [\check{\mathbf{G}}_a^T \ \check{\mathbf{G}}_a^T]^T$. An effective enclosure of the prediction step for the descriptor system (7.7) can be obtained in CG-rep as follows.

Lemma 7.1. For $k \geq 1$, consider $\mathbf{z}_{k-1} \in \hat{Z}_{k-1} = \{\hat{\mathbf{G}}_{k-1}, \hat{\mathbf{c}}_{k-1}, \hat{\mathbf{A}}_{k-1}, \hat{\mathbf{b}}_{k-1}\}$, $\mathbf{w}_{k-1}, \mathbf{w}_k \in W = \{\mathbf{G}_w, \mathbf{c}_w, \mathbf{A}_w, \mathbf{b}_w\}$, and system (7.7). If Assumption 7.1 holds, then $\mathbf{z}_k \in \bar{Z}_k = \{\bar{\mathbf{G}}_k, \bar{\mathbf{c}}_k, \bar{\mathbf{A}}_k, \bar{\mathbf{b}}_k\}$, with

$$\begin{aligned} \bar{\mathbf{G}}_k &= \begin{bmatrix} \check{\mathbf{A}}\hat{\mathbf{G}}_{k-1} & \check{\mathbf{B}}_w\mathbf{G}_w & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \check{\mathbf{G}}_a & \mathbf{0} \end{bmatrix}, \quad \bar{\mathbf{c}}_k = \begin{bmatrix} \check{\mathbf{A}}\hat{\mathbf{c}}_{k-1} + \check{\mathbf{B}}\mathbf{u}_{k-1} + \check{\mathbf{B}}_w\mathbf{c}_w \\ \check{\mathbf{c}}_a \end{bmatrix}, \\ \bar{\mathbf{A}}_k &= \begin{bmatrix} \text{blkdiag}(\hat{\mathbf{A}}_{k-1}, \mathbf{A}_w, \mathbf{A}_a, \mathbf{A}_w) \\ \check{\mathbf{A}} \begin{bmatrix} \check{\mathbf{A}}\hat{\mathbf{G}}_{k-1} \\ \mathbf{0} \end{bmatrix} & \check{\mathbf{A}} \begin{bmatrix} \check{\mathbf{B}}_w\mathbf{G}_w \\ \mathbf{0} \end{bmatrix} & \check{\mathbf{A}} \begin{bmatrix} \mathbf{0} \\ \check{\mathbf{G}}_a \end{bmatrix} & \check{\mathbf{B}}_w\mathbf{G}_w \end{bmatrix}, \\ \bar{\mathbf{b}}_k &= \begin{bmatrix} [\hat{\mathbf{b}}_{k-1}^T \ \mathbf{b}_w^T \ \mathbf{b}_a^T \ \mathbf{b}_w^T]^T \\ -\check{\mathbf{A}} \begin{bmatrix} \check{\mathbf{A}}\hat{\mathbf{c}}_{k-1} + \check{\mathbf{B}}\mathbf{u}_{k-1} + \check{\mathbf{B}}_w\mathbf{c}_w \\ \check{\mathbf{c}}_a \end{bmatrix} & -\check{\mathbf{B}}\mathbf{u}_k - \check{\mathbf{B}}_w\mathbf{c}_w \end{bmatrix}. \end{aligned}$$

Proof. Since by assumption $(\mathbf{z}_{k-1}, \mathbf{w}_{k-1}, \mathbf{w}_k) \in \hat{Z}_{k-1} \times W \times W$, there exists $(\boldsymbol{\xi}_{k-1}, \boldsymbol{\delta}_{k-1}, \boldsymbol{\delta}_k) \in B_\infty(\hat{\mathbf{A}}_{k-1}, \hat{\mathbf{b}}_{k-1}) \times B_\infty(\mathbf{A}_w, \mathbf{b}_w) \times B_\infty(\mathbf{A}_a, \mathbf{b}_a)$ such that $\mathbf{z}_{k-1} = \hat{\mathbf{c}}_{k-1} + \hat{\mathbf{G}}_{k-1}\boldsymbol{\xi}_{k-1}$, $\mathbf{w}_{k-1} = \mathbf{c}_w + \mathbf{G}_w\boldsymbol{\delta}_{k-1}$, and $\mathbf{w}_k = \mathbf{c}_w + \mathbf{G}_w\boldsymbol{\delta}_k$. Besides, Assumption 7.1 implies that $\mathbf{z}_k \in Z_a$. Thus, there must exist $\boldsymbol{\xi}_a \in B_\infty(\mathbf{A}_a, \mathbf{b}_a)$ such that $\check{\mathbf{z}}_k = \check{\mathbf{c}}_a + \check{\mathbf{G}}_a\boldsymbol{\xi}_a$. Therefore, substituting these equalities in (7.7a) leads to

$$(\check{\mathbf{z}}_k, \check{\mathbf{z}}_k) = (\check{\mathbf{A}}\hat{\mathbf{c}}_{k-1} + \check{\mathbf{B}}\mathbf{u}_{k-1} + \check{\mathbf{B}}_w\mathbf{c}_w + \check{\mathbf{A}}\hat{\mathbf{G}}_{k-1}\boldsymbol{\xi}_{k-1} + \check{\mathbf{B}}_w\mathbf{G}_w\boldsymbol{\delta}_{k-1}, \check{\mathbf{c}}_a + \check{\mathbf{G}}_a\boldsymbol{\xi}_a). \quad (7.8)$$

¹This is an abuse of notation since $\hat{Z}_0 \neq \mathbf{T}^{-1}\hat{X}_0$. The constraints $\check{\mathbf{A}}\mathbf{z}_0 + \check{\mathbf{B}}\mathbf{u}_0 + \check{\mathbf{B}}_w\mathbf{w}_0 = \mathbf{0}$ are incorporated in \hat{Z}_0 , while not in $\mathbf{T}^{-1}\hat{X}_0$.

From the constraint (7.7b) shifted to time k , we have that

$$\begin{aligned} & \check{\mathbf{B}}_w \mathbf{c}_w + \check{\mathbf{B}}_w \mathbf{G}_w \boldsymbol{\delta}_k + \check{\mathbf{A}} \begin{bmatrix} \check{\mathbf{A}} \hat{\mathbf{c}}_{k-1} + \check{\mathbf{B}} \mathbf{u}_{k-1} + \check{\mathbf{B}}_w \mathbf{c}_w \\ \check{\mathbf{c}}_a \end{bmatrix} \\ & + \check{\mathbf{A}} \begin{bmatrix} \check{\mathbf{A}} \hat{\mathbf{G}}_{k-1} & \check{\mathbf{B}}_w \mathbf{G}_w & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \check{\mathbf{G}}_a \end{bmatrix} \begin{bmatrix} \boldsymbol{\xi}_{k-1} \\ \boldsymbol{\delta}_{k-1} \\ \boldsymbol{\xi}_a \end{bmatrix} + \check{\mathbf{B}} \mathbf{u}_k = \mathbf{0}. \end{aligned} \quad (7.9)$$

Rearranging (7.8) and (7.9), grouping $(\boldsymbol{\xi}_{k-1}, \boldsymbol{\delta}_{k-1}, \boldsymbol{\xi}_a, \boldsymbol{\delta}_k)$, and writing in the CG-rep (3.7) proves the lemma. \blacksquare

Lemma 7.1 provides a predicted enclosure of the state \mathbf{z}_k in which the equality constraints (7.7b), shifted to time k , are directly taken into account. This is possible thanks to the fact that CZs incorporate equality constraints (see (3.7)). Finally, the prediction-update algorithm proposed for descriptor systems consists in the computation of CZs \bar{Z}_k , \hat{Z}_k , and \hat{X}_k , such that

$$\bar{Z}_k = \{\bar{\mathbf{G}}_k, \bar{\mathbf{c}}_k, \bar{\mathbf{A}}_k, \bar{\mathbf{b}}_k\}, \quad (7.10)$$

$$\hat{Z}_k = \bar{Z}_k \cap_{\text{CT}} ((\mathbf{y}_k - \mathbf{D}_u \mathbf{u}_k) \oplus (-\mathbf{D}_v V_k)), \quad (7.11)$$

$$\hat{X}_k = \mathbf{T} \hat{Z}_k. \quad (7.12)$$

For this algorithm, the initial set is \hat{Z}_0 . The algorithm (7.10)–(7.12) operates recursively with \bar{Z}_k and \hat{Z}_k for $k \geq 1$ in the transformed state-space (7.7), while the estimated enclosure in the original state-space (7.1) is given by \hat{X}_k .

Remark 7.2. The set Z_a is used only to predict an enclosure for the components $\check{\mathbf{z}}_k$. This way, the static relation (7.7b) is incorporated as constraints to the variables $\boldsymbol{\xi}_a$ in Z_a .

Remark 7.3. By construction, the CG-rep (7.10) corresponds to the exact feasible state set of (7.7) at k for the known state and uncertainty bounds. In addition, (7.11)–(7.12) can be computed exactly. However, in practice, in order to limit the complexity of the resulting sets these are outer approximated by using order reduction algorithms. In this case, equalities (7.10)–(7.12) are replaced by the relation \supset .

7.3 Active fault diagnosis of LDS using constrained zonotopes

The previous section presented a method to address the problem of set-based estimation of descriptor systems using constrained zonotopes. In the following, this tool is used in the design of a CZ-based AFD method accounting for a finite number of possible abrupt faults.

7.3.1 State and output reachable sets

Let $\vec{\mathbf{u}} = (\mathbf{u}_0, \dots, \mathbf{u}_N) \in \mathbb{R}^{(N+1)nu}$, $\vec{\mathbf{w}} = (\mathbf{w}_0, \dots, \mathbf{w}_N) \in \mathbb{R}^{(N+1)nw}$, and $\vec{W} = W \times \dots \times W$. Consider a variable transformation similar to the one used in the previous section. With a slight abuse of notation, let $\mathbf{z}_k = (\tilde{\mathbf{z}}_k, \check{\mathbf{z}}_k) = (\mathbf{T}^{-1}\mathbf{x}_k, \mathbf{w}_k)$, $\tilde{\mathbf{z}}_k \in \mathbb{R}^{n_z}$, $\check{\mathbf{z}}_k \in \mathbb{R}^{n+n_w-n_z}$, with $\mathbf{T}^{[i]} = ((\mathbf{V}^{[i]})^T)^{-1}$, $\mathbf{V}^{[i]}$ being obtained from the SVD $\mathbf{E}^{[i]} = \mathbf{U}^{[i]}\mathbf{\Sigma}^{[i]}(\mathbf{V}^{[i]})^T$. Then, (7.2) can be rewritten as

$$\tilde{\mathbf{z}}_k^{[i]} = \tilde{\mathbf{A}}_z^{[i]}\mathbf{z}_{k-1}^{[i]} + \tilde{\mathbf{B}}^{[i]}\mathbf{u}_{k-1}, \quad (7.13a)$$

$$\mathbf{0} = \check{\mathbf{A}}_z^{[i]}\mathbf{z}_k^{[i]} + \check{\mathbf{B}}^{[i]}\mathbf{u}_k, \quad (7.13b)$$

$$\mathbf{y}_k^{[i]} = \mathbf{F}^{[i]}\mathbf{z}_k^{[i]} + \mathbf{D}^{[i]}\mathbf{u}_k + \mathbf{D}_v^{[i]}\mathbf{v}_k, \quad (7.13c)$$

with $\mathbf{F}^{[i]} = \mathbf{C}^{[i]}\mathbf{T}^{[i]}\mathbf{P}$, where $\mathbf{P} = [\mathbf{I}_n \ \mathbf{0}_{n \times n_w}]$, $\tilde{\mathbf{A}}_z^{[i]} = [\tilde{\mathbf{A}}^{[i]} \ \tilde{\mathbf{B}}_w^{[i]}]$, and $\check{\mathbf{A}}_z^{[i]} = [\check{\mathbf{A}}^{[i]} \ \check{\mathbf{B}}_w^{[i]}]$. Note that the $(\tilde{\cdot})$ and $(\check{\cdot})$ variables are defined according to (7.6) for each i , and equation (7.13b) has been already shifted to time k .

For each model i , consider the CZ $Z_a^{[i]} = (\mathbf{T}^{[i]})^{-1}X_a \times W = \{\mathbf{G}_a^{[i]}, \mathbf{c}_a^{[i]}, \mathbf{A}_a^{[i]}, \mathbf{b}_a^{[i]}\}$, where X_a satisfies Assumption 1, the set $\{\mathbf{G}_z^{[i]}, \mathbf{c}_z^{[i]}, \mathbf{A}_z^{[i]}, \mathbf{b}_z^{[i]}\} = (\mathbf{T}^{[i]})^{-1}X_0 \times W$, and define the initial feasible set $Z_0^{[i]}(\mathbf{u}_0) = \{\mathbf{z} \in (\mathbf{T}^{[i]})^{-1}X_0 \times W : (7.13b) \text{ holds for } k = 0\}$. This set is given by $Z_0^{[i]}(\mathbf{u}_0) = \{\mathbf{G}_0^{[i]}, \mathbf{c}_0^{[i]}, \mathbf{A}_0^{[i]}, \mathbf{b}_0^{[i]}(\mathbf{u}_0)\}$, where $\mathbf{G}_0^{[i]} = \mathbf{G}_z^{[i]}$, $\mathbf{c}_0^{[i]} = \mathbf{c}_z^{[i]}$,

$$\mathbf{A}_0^{[i]} = \begin{bmatrix} \mathbf{A}_z^{[i]} \\ \check{\mathbf{A}}_z^{[i]}\mathbf{G}_0^{[i]} \end{bmatrix}, \quad \mathbf{b}_0^{[i]}(\mathbf{u}_0) = \begin{bmatrix} \mathbf{b}_z^{[i]} \\ -\check{\mathbf{A}}_z^{[i]}\mathbf{c}_0^{[i]} - \check{\mathbf{B}}^{[i]}\mathbf{u}_0 \end{bmatrix}. \quad (7.14)$$

In addition, define the solution mappings $(\phi_k^{[i]}, \psi_k^{[i]}) : \mathbb{R}^{(k+1)nu} \times \mathbb{R}^n \times \mathbb{R}^{(k+1)nw} \times \mathbb{R}^{nv} \rightarrow \mathbb{R}^{n+n_w} \times \mathbb{R}^{ny}$ such that $\phi_k^{[i]}(\vec{\mathbf{u}}, \mathbf{z}_0, \vec{\mathbf{w}})$ and $\psi_k^{[i]}(\vec{\mathbf{u}}, \mathbf{z}_0, \vec{\mathbf{w}}, \mathbf{v}_k)$ are the state and output of (7.13) at k , respectively. Then, for each $i \in \mathcal{M}$, define *state and output reachable sets* at time k as

$$Z_k^{[i]}(\vec{\mathbf{u}}) = \{\phi_k^{[i]}(\vec{\mathbf{u}}, \mathbf{z}_0, \vec{\mathbf{w}}) : (\mathbf{z}_0^{[i]}, \vec{\mathbf{w}}) \in Z_0^{[i]}(\mathbf{u}_0) \times \vec{W}\},$$

$$Y_k^{[i]}(\vec{\mathbf{u}}) = \{\psi_k^{[i]}(\vec{\mathbf{u}}, \mathbf{z}_0, \vec{\mathbf{w}}, \mathbf{v}_k) : (\mathbf{z}_0, \vec{\mathbf{w}}, \mathbf{v}_k) \in Z_0^{[i]}(\mathbf{u}_0) \times \vec{W} \times V\}.$$

Using (3.8)–(3.9), and taking note that by assumption $\mathbf{z}_k^{[i]} \in Z_a^{[i]} \subset \mathbb{R}^n \times W$ for every $k \geq 0$, the set $Z_N^{[i]}(\vec{\mathbf{u}})$ is given by the CZ $\{\mathbf{G}_N^{[i]}, \mathbf{c}_N^{[i]}(\vec{\mathbf{u}}), \mathbf{A}_N^{[i]}, \mathbf{b}_N^{[i]}(\vec{\mathbf{u}})\}$, where $\mathbf{G}_N^{[i]}$, $\mathbf{c}_N^{[i]}(\vec{\mathbf{u}})$, $\mathbf{A}_N^{[i]}$, and $\mathbf{b}_N^{[i]}(\vec{\mathbf{u}})$ are obtained by the recursive relations

$$\mathbf{c}_k^{[i]}(\vec{\mathbf{u}}) = \begin{bmatrix} \tilde{\mathbf{A}}_z^{[i]}\mathbf{c}_{k-1}^{[i]}(\vec{\mathbf{u}}) + \tilde{\mathbf{B}}^{[i]}\mathbf{u}_{k-1} \\ \check{\mathbf{c}}_a^{[i]} \end{bmatrix}, \quad \mathbf{G}_k^{[i]} = \begin{bmatrix} \tilde{\mathbf{A}}_z^{[i]}\mathbf{G}_{k-1}^{[i]} & \mathbf{0} \\ \mathbf{0} & \check{\mathbf{G}}_a^{[i]} \end{bmatrix},$$

$$\mathbf{A}_k^{[i]} = \begin{bmatrix} \mathbf{A}_{k-1}^{[i]} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_a^{[i]} \\ \check{\mathbf{A}}_z^{[i]}\mathbf{G}_k^{[i]} \end{bmatrix}, \quad \mathbf{b}_k^{[i]}(\vec{\mathbf{u}}) = \begin{bmatrix} \mathbf{b}_{k-1}^{[i]}(\vec{\mathbf{u}}) \\ \mathbf{b}_a^{[i]} \\ -\check{\mathbf{A}}_z^{[i]}\mathbf{c}_k^{[i]}(\vec{\mathbf{u}}) - \check{\mathbf{B}}^{[i]}\mathbf{u}_k \end{bmatrix},$$

for $k = 1, 2, \dots, N$. Note that the third constraint in $(\mathbf{A}_k^{[i]}, \mathbf{b}_k^{[i]}(\vec{\mathbf{u}}))$ comes from the fact that

(7.13b) must hold. Using the initial values (7.14), the variables $\mathbf{c}_N^{[i]}(\vec{\mathbf{u}})$ and $\mathbf{b}_N^{[i]}(\vec{\mathbf{u}})$ can be written as explicit functions of the input sequence $\vec{\mathbf{u}}$ as

$$\mathbf{c}_N^{[i]}(\vec{\mathbf{u}}) = \begin{bmatrix} \tilde{\mathbf{A}}_z^{[i]} \\ \mathbf{0} \end{bmatrix}^N \mathbf{c}_z^{[i]} + \sum_{m=1}^N \left(\begin{bmatrix} \tilde{\mathbf{A}}_z^{[i]} \\ \mathbf{0} \end{bmatrix}^{m-1} \begin{bmatrix} \mathbf{0} \\ \check{\mathbf{c}}_a^{[i]} \end{bmatrix} \right) + \mathbf{H}_N^{[i]} \vec{\mathbf{u}}, \quad (7.15)$$

$$\mathbf{b}_N^{[i]}(\vec{\mathbf{u}}) = \boldsymbol{\alpha}_N^{[i]} + \boldsymbol{\Lambda}_N^{[i]} \mathbf{c}_z^{[i]} + \boldsymbol{\Omega}_N^{[i]} \vec{\mathbf{u}}, \quad (7.16)$$

where $\boldsymbol{\alpha}_N^{[i]} = \boldsymbol{\beta}_N^{[i]} + \boldsymbol{\Upsilon}_N^{[i]} \mathbf{p}_N^{[i]}$, $\boldsymbol{\Lambda}_N^{[i]} = \boldsymbol{\Upsilon}_N^{[i]} \mathbf{Q}_N^{[i]}$, $\boldsymbol{\Omega}_N^{[i]} = \boldsymbol{\Gamma}_N^{[i]} + \boldsymbol{\Upsilon}_N^{[i]} \vec{\mathbf{H}}^{[i]}$,

$$\begin{aligned} \boldsymbol{\beta}_N^{[i]} &= \left[[(\mathbf{b}_z^{[i]})^T \ \mathbf{0}] \quad [(\mathbf{b}_a^{[i]})^T \ \mathbf{0}] \quad \dots \quad [(\mathbf{b}_a^{[i]})^T \ \mathbf{0}] \right]^T, \\ \mathbf{p}_N^{[i]} &= \left[\begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}^T \quad \begin{bmatrix} \mathbf{0} \\ \check{\mathbf{c}}_a^{[i]} \end{bmatrix}^T \quad \dots \quad \sum_{m=1}^N \left(\begin{bmatrix} \tilde{\mathbf{A}}_z^{[i]} \\ \mathbf{0} \end{bmatrix}^{m-1} \begin{bmatrix} \mathbf{0} \\ \check{\mathbf{c}}_a^{[i]} \end{bmatrix} \right)^T \right]^T, \\ \boldsymbol{\Upsilon}_N^{[i]} &= \text{blkdiag}([\mathbf{0} \ (-\check{\mathbf{A}}_z^{[i]})^T]^T, \dots, [\mathbf{0} \ (-\check{\mathbf{A}}_z^{[i]})^T]^T), \\ \boldsymbol{\Gamma}_N^{[i]} &= \text{blkdiag}([\mathbf{0} \ (-\check{\mathbf{B}}^{[i]})^T]^T, \dots, [\mathbf{0} \ (-\check{\mathbf{B}}^{[i]})^T]^T), \\ \mathbf{Q}_N^{[i]} &= \left[\dots \quad \left(\begin{bmatrix} \tilde{\mathbf{A}}_z^{[i]} \\ \mathbf{0} \end{bmatrix}^\ell \right)^T \quad \dots \right]^T, \quad \vec{\mathbf{H}}^{[i]} = [\dots \quad (\mathbf{H}_\ell^{[i]})^T \quad \dots]^T, \\ \mathbf{H}_h^{[i]} &= \left[\dots \quad \underbrace{\begin{bmatrix} \tilde{\mathbf{A}}_z^{[i]} \\ \mathbf{0} \end{bmatrix}^{h-m} \begin{bmatrix} \tilde{\mathbf{B}}^{[i]} \\ \mathbf{0} \end{bmatrix}}_{m=1,2,\dots,h} \quad \dots \quad \underbrace{\begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}}_{N-h+1 \text{ terms}} \quad \dots \right], \end{aligned}$$

with $\ell = 0, \dots, N$. The variables $\boldsymbol{\beta}_N$, \mathbf{p}_N , $\boldsymbol{\Upsilon}_N$, and $\boldsymbol{\Gamma}_N$ have $N + 1$ block matrices. In addition, the expression $\mathbf{H}_h^{[i]}$ holds for $h = 1, \dots, N$, while $\mathbf{H}_0^{[i]} = \mathbf{0}_{n \times (N+1)n_u}$.

Since $Z_N^{[i]}(\vec{\mathbf{u}})$ is a CZ, the output reachable set $Y_N^{[i]}(\vec{\mathbf{u}})$ is then a CZ obtained in accordance with (7.13c) as $Y_N^{[i]}(\vec{\mathbf{u}}) = \mathbf{F}^{[i]} Z_N^{[i]}(\vec{\mathbf{u}}) \oplus \mathbf{D}^{[i]} \mathbf{u}_N \oplus \mathbf{D}_v^{[i]} V$. Using properties (3.8) and (3.9), and letting $V = \{\mathbf{G}_v, \mathbf{c}_v, \mathbf{A}_v, \mathbf{b}_v\}$, this set is $Y_N^{[i]}(\vec{\mathbf{u}}) = \{\mathbf{G}_N^{Y[i]}, \mathbf{c}_N^{Y[i]}(\vec{\mathbf{u}}), \mathbf{A}_N^{Y[i]}, \mathbf{b}_N^{Y[i]}(\vec{\mathbf{u}})\}$, with

$$\mathbf{c}_N^{Y[i]}(\vec{\mathbf{u}}) = \mathbf{F}^{[i]} \mathbf{c}_N^{[i]}(\vec{\mathbf{u}}) + \mathbf{D}^{[i]} \mathbf{u}_N + \mathbf{D}_v^{[i]} \mathbf{c}_v, \quad (7.17a)$$

$$\mathbf{G}_N^{Y[i]} = [\mathbf{F}^{[i]} \mathbf{G}_N^{[i]} \ \mathbf{G}_v], \quad \mathbf{A}_N^{Y[i]} = \text{blkdiag}(\mathbf{A}_N^{[i]}, \mathbf{A}_v), \quad (7.17b)$$

$$\mathbf{b}_N^{Y[i]}(\vec{\mathbf{u}}) = [(\mathbf{b}_N^{[i]}(\vec{\mathbf{u}}))^T \ \mathbf{b}_v^T]^T. \quad (7.17c)$$

7.3.2 Input design

Consider an input sequence $\vec{\mathbf{u}} \in \vec{U}$ to be injected into the set of models (7.13), and let $\mathbf{y}_N^{[i]}$ denote the observed output. We are interested in the design of an input sequence such that the relation $\mathbf{y}_N^{[i]} \in Y_N^{[i]}(\vec{\mathbf{u}})$ is valid for only one $i \in \mathcal{M}$.

Definition 7.1. An input sequence $\vec{\mathbf{u}}$ is said to be a *separating input* on $k \in [0, N]$ if, for every $i, j \in \mathcal{M}$, $i \neq j$,

$$Y_N^{[i]}(\vec{\mathbf{u}}) \cap Y_N^{[j]}(\vec{\mathbf{u}}) = \emptyset. \quad (7.18)$$

Clearly, if (7.18) holds for all $i, j \in \mathcal{M}$, $i \neq j$, then $\mathbf{y}_N^{[i]} \in Y_N^{[i]}(\vec{\mathbf{u}})$ must hold only for one i . In the case that this is not valid for any $i \in \mathcal{M}$, one concludes that the real dynamics does not belong to the set of models (7.13). The following theorem is based on the computation of $Y_N^{[i]}(\vec{\mathbf{u}})$ expressed by (7.17) and the results in Raimondo et al. (2016).

Theorem 7.1. An input $\vec{\mathbf{u}} \in \vec{U}$ is a separating input iff

$$\left[\begin{array}{c} \mathbf{N}(i, j) \\ \boldsymbol{\Omega}(i, j) \end{array} \right] \vec{\mathbf{u}} \notin \mathcal{Y}(i, j) = \left\{ \left[\begin{array}{c} \mathbf{G}_N^Y(i, j) \\ \mathbf{A}_N^Y(i, j) \end{array} \right], \left[\begin{array}{c} \mathbf{c}_N^Y(i, j) \\ -\mathbf{b}_N^Y(i, j) \end{array} \right] \right\}, \quad (7.19)$$

$\forall i, j \in \mathcal{M}$, $i \neq j$, where $\mathbf{N}(i, j) = \mathbf{F}^{[j]} \vec{\mathbf{H}}^{[j]} - \mathbf{F}^{[i]} \vec{\mathbf{H}}^{[i]} + [\mathbf{0} \ (\mathbf{D}^{[j]} - \mathbf{D}^{[i]})]$, $\boldsymbol{\Omega}(i, j) = [(\boldsymbol{\Omega}_N^{[i]})^T \ \mathbf{0} \ (\boldsymbol{\Omega}_N^{[j]})^T \ \mathbf{0}]^T$, and

$$\begin{aligned} \mathbf{G}_N^Y(i, j) &= [\mathbf{G}_N^{Y[i]} - \mathbf{G}_N^{Y[j]}], \quad \mathbf{c}_N^Y(i, j) = \mathbf{c}_N^{Y[i]}(\vec{\mathbf{0}}) - \mathbf{c}_N^{Y[j]}(\vec{\mathbf{0}}), \\ \mathbf{A}_N^Y(i, j) &= \left[\begin{array}{cc} \mathbf{A}_N^{Y[i]} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_N^{Y[j]} \end{array} \right], \quad \mathbf{b}_N^Y(i, j) = \left[\begin{array}{c} \mathbf{b}_N^{Y[i]}(\vec{\mathbf{0}}) \\ \mathbf{b}_N^{Y[j]}(\vec{\mathbf{0}}) \end{array} \right], \end{aligned}$$

where $\vec{\mathbf{0}}$ denotes the zero input sequence.

Proof. The relations below follow from (7.15)–(7.17a) and (7.17c):

$$\mathbf{c}_N^{Y[i]}(\vec{\mathbf{u}}) = \mathbf{c}_N^{Y[i]}(\vec{\mathbf{0}}) + \mathbf{F}^{[i]} \mathbf{H}_N^{[i]} \vec{\mathbf{u}} + \mathbf{D}^{[i]} \mathbf{u}_N, \quad (7.20)$$

$$\mathbf{b}_N^{Y[i]}(\vec{\mathbf{u}}) = \mathbf{b}_N^{Y[i]}(\vec{\mathbf{0}}) + [(\boldsymbol{\Omega}_N^{[i]})^T \ \mathbf{0}]^T \vec{\mathbf{u}}. \quad (7.21)$$

From (3.10) with $\mathbf{R} = \mathbf{I}$, (7.18) is true iff $\# \boldsymbol{\xi} \in B_\infty$ such that

$$\left[\begin{array}{cc} \mathbf{A}_N^{Y[i]} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_N^{Y[j]} \\ \mathbf{G}_N^{Y[i]} & -\mathbf{G}_N^{Y[j]} \end{array} \right] \boldsymbol{\xi} = \left[\begin{array}{c} \mathbf{b}_N^{Y[i]}(\vec{\mathbf{u}}) \\ \mathbf{b}_N^{Y[j]}(\vec{\mathbf{u}}) \\ \mathbf{c}_N^{Y[j]}(\vec{\mathbf{u}}) - \mathbf{c}_N^{Y[i]}(\vec{\mathbf{u}}) \end{array} \right].$$

According to (7.20)–(7.21), one has $\mathbf{c}_N^{Y[j]}(\vec{\mathbf{u}}) - \mathbf{c}_N^{Y[i]}(\vec{\mathbf{u}}) = -\mathbf{c}_N^Y(i, j) + \mathbf{N}(i, j) \vec{\mathbf{u}}$, $[(\mathbf{b}_N^{Y[i]}(\vec{\mathbf{u}}))^T \ (\mathbf{b}_N^{Y[j]}(\vec{\mathbf{u}}))^T]^T = \mathbf{b}_N^Y(i, j) + \boldsymbol{\Omega}(i, j) \vec{\mathbf{u}}$, with $\mathbf{c}_N^Y(i, j)$, $\mathbf{b}_N^Y(i, j)$, $\mathbf{N}(i, j)$, and $\boldsymbol{\Omega}(i, j)$ defined as in the statement of the theorem. Then, (7.18) holds iff $\# \boldsymbol{\xi} \in B_\infty$ such that $\mathbf{G}_N^Y(i, j) \boldsymbol{\xi} = -\mathbf{c}_N^Y(i, j) + \mathbf{N}(i, j) \vec{\mathbf{u}}$, and $\mathbf{A}_N^Y(i, j) \boldsymbol{\xi} = \mathbf{b}_N^Y(i, j) + \boldsymbol{\Omega}(i, j) \vec{\mathbf{u}}$. This is equivalent to

$$\# \boldsymbol{\xi} \in B_\infty : \left[\begin{array}{c} \mathbf{G}_N^Y(i, j) \\ \mathbf{A}_N^Y(i, j) \end{array} \right] \boldsymbol{\xi} + \left[\begin{array}{c} \mathbf{c}_N^Y(i, j) \\ -\mathbf{b}_N^Y(i, j) \end{array} \right] = \left[\begin{array}{c} \mathbf{N}(i, j) \\ \boldsymbol{\Omega}(i, j) \end{array} \right] \vec{\mathbf{u}},$$

which in turn holds iff (7.19) is satisfied with $\mathbf{G}_N^Y(i, j)$ and $\mathbf{A}_N^Y(i, j)$ defined as in the statement of the theorem. \blacksquare

Let n_q denote the number of all possible combinations of $i, j \in \mathcal{M}$, $i \neq j$, and define $\mathbf{N}^{\mathcal{Y}[q]} = [\mathbf{N}^T(i, j) \ \boldsymbol{\Omega}^T(i, j)]^T$, $\mathcal{Y}^{[q]} = \mathcal{Y}(i, j) = \{\mathbf{G}^{\mathcal{Y}[q]}, \mathbf{c}^{\mathcal{Y}[q]}\}$, for each $q \in \{1, 2, \dots, n_q\}$, with

$\{\mathbf{G}^{\mathcal{Y}^{[q]}}, \mathbf{c}^{\mathcal{Y}^{[q]}}\}$ being the right hand side of (7.19). As it can be noticed, $\mathcal{Y}^{[q]}$ is a zonotope. Then, the relation $\mathbf{N}^{\mathcal{Y}^{[q]}}\vec{\mathbf{u}} \notin \mathcal{Y}^{[q]}$ can be verified by solving a linear program (LP) similar to what was proposed in Scott et al. (2014). In this sense, the following lemma provides an effective way to verify if a given input sequence is a separating input according to Theorem 7.1, consequently satisfying (7.18).

Lemma 7.2. Let $\mathcal{Y}^{[q]} = \{\mathbf{G}^{\mathcal{Y}^{[q]}}, \mathbf{c}^{\mathcal{Y}^{[q]}}\}$. For each $\vec{\mathbf{u}} \in \vec{U}$ and $q \in \{1, 2, \dots, n_q\}$, define $\hat{\delta}^{[q]}(\vec{\mathbf{u}}) = \min_{\delta^{[q]}, \xi^{[q]}} \delta^{[q]}$, subject to

$$\mathbf{N}^{\mathcal{Y}^{[q]}}\vec{\mathbf{u}} = \mathbf{G}^{\mathcal{Y}^{[q]}}\xi^{[q]} + \mathbf{c}^{\mathcal{Y}^{[q]}}, \quad \|\xi\|_{\infty} \leq 1 + \delta^{[q]}.$$

Then $\mathbf{N}^{\mathcal{Y}^{[q]}}\vec{\mathbf{u}} \notin \mathcal{Y}^{[q]} \iff \hat{\delta}^{[q]}(\vec{\mathbf{u}}) > 0$.

Proof. See Lemma 4 in Scott et al. (2014). ■

For the AFD of the n_m models in (7.13) of the descriptor system, we consider the design of a separating input of minimum length according to the optimization problem

$$\min_{\vec{\mathbf{u}} \in \vec{U}} \{J(\vec{\mathbf{u}}) : \mathbf{N}^{\mathcal{Y}^{[q]}}\vec{\mathbf{u}} \notin \mathcal{Y}^{[q]}, \forall q = 1, 2, \dots, n_q\}, \quad (7.22)$$

with $J(\vec{\mathbf{u}})$ chosen to minimize any harmful effects caused by injecting $\vec{\mathbf{u}}$ into (7.13). For simplicity, we may choose $J(\vec{\mathbf{u}}) = \sum_{j=0}^N \mathbf{u}_j^T \mathbf{R} \mathbf{u}_j$, where \mathbf{R} is a weighting matrix. As in Scott et al. (2014), this is a bilevel optimization problem and can be rewritten as a mixed-integer quadratic program by defining a *minimum separation threshold* $\varepsilon > 0$ such that $\varepsilon \leq \hat{\delta}^{[q]}(\vec{\mathbf{u}})$, for all $q = 1, 2, \dots, n_q$, similar to the one presented in Chapter 6.

7.4 Mixed zonotopes

In this section, we propose a new set representation referred to as *mixed zonotope*, which is an extension of constrained zonotopes that describes a larger class of sets. Mixed zonotopes inherit most of the properties of constrained zonotopes, while having the additional ability to describe unbounded sets.

Definition 7.2. A set $Z \subseteq \mathbb{R}^n$ is a *mixed zonotope* if there exists $(\mathbf{M}_z, \mathbf{G}_z, \mathbf{c}_z, \mathbf{S}_z, \mathbf{A}_z, \mathbf{b}_z) \in \mathbb{R}^{n \times n_{\delta}} \times \mathbb{R}^{n \times n_g} \times \mathbb{R}^n \times \mathbb{R}^{n_c \times n_{\delta}} \times \mathbb{R}^{n_c \times n_g} \times \mathbb{R}^{n_c}$ such that

$$Z = \{\mathbf{c}_z + \mathbf{M}_z \boldsymbol{\delta} + \mathbf{G}_z \boldsymbol{\xi} : \boldsymbol{\delta} \in \mathbb{R}^{n_{\delta}}, \|\boldsymbol{\xi}\|_{\infty} \leq 1, \mathbf{S}_z \boldsymbol{\delta} + \mathbf{A}_z \boldsymbol{\xi} = \mathbf{b}_z\}. \quad (7.23)$$

We refer to (7.23) as *mixed generator representation* (MG-rep). Each column of \mathbf{M}_z is an *unbounded generator*, each column of \mathbf{G}_z is a *bounded generator*, \mathbf{c}_z is the *center*, and $\mathbf{S}_z \boldsymbol{\delta} + \mathbf{A}_z \boldsymbol{\xi} = \mathbf{b}_z$ are the *constraints*. Note that, by the definition of the MG-rep (7.23), if \mathbf{S}_z is a matrix of zeros, then the mixed zonotope Z is *unbounded* in the directions given by

the columns of \mathbf{M}_z . This allows the MG-rep (7.23) to describe symmetrically unbounded sets, such as strips (Proposition 7.2) and the entire space \mathbb{R}^n (Proposition 7.1). We use the shorthand notation $Z = \{\mathbf{M}_z, \mathbf{G}_z, \mathbf{c}_z, \mathbf{S}_z, \mathbf{A}_z, \mathbf{b}_z\}$ for mixed zonotopes. Moreover, we denote by ‘ $_$ ’ an empty argument to this shorthand notation. For instance, zonotopes are expressed in MG-rep as $\{\mathbf{G}_z, \mathbf{c}_z\} = \{_, \mathbf{G}_z, \mathbf{c}_z, _, _, _\}$, and constrained zonotopes as $\{\mathbf{G}_z, \mathbf{c}_z, \mathbf{A}_z, \mathbf{b}_z\} = \{_, \mathbf{G}_z, \mathbf{c}_z, _, \mathbf{A}_z, \mathbf{b}_z\}$.

Consider sets $Z, W \subseteq \mathbb{R}^n$, $Y \subseteq \mathbb{R}^m$, and a real matrix $\mathbf{R} \in \mathbb{R}^{m \times n}$. Similarly to constrained zonotopes, if Z, W, Y are in MG-rep, i.e., $Z \triangleq \{\mathbf{M}_z, \mathbf{G}_z, \mathbf{c}_z, \mathbf{S}_z, \mathbf{A}_z, \mathbf{b}_z\}$, $W \triangleq \{\mathbf{M}_w, \mathbf{G}_w, \mathbf{c}_w, \mathbf{S}_w, \mathbf{A}_w, \mathbf{b}_w\}$, and $Y \triangleq \{\mathbf{M}_y, \mathbf{G}_y, \mathbf{c}_y, \mathbf{S}_y, \mathbf{A}_y, \mathbf{b}_y\}$, then the elementary set operations (3.1)–(3.3) are computed trivially as

$$\mathbf{R}Z = \{\mathbf{R}\mathbf{M}_z, \mathbf{R}\mathbf{G}_z, \mathbf{R}\mathbf{c}_z, \mathbf{S}_z, \mathbf{A}_z, \mathbf{b}_z\}, \quad (7.24)$$

$$Z \oplus W = \left\{ \begin{bmatrix} \mathbf{M}_z & \mathbf{M}_w \end{bmatrix}, \begin{bmatrix} \mathbf{G}_z & \mathbf{G}_w \end{bmatrix}, \mathbf{c}_z + \mathbf{c}_w, \begin{bmatrix} \mathbf{S}_z & \mathbf{0} \\ \mathbf{0} & \mathbf{S}_w \end{bmatrix}, \begin{bmatrix} \mathbf{A}_z & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_w \end{bmatrix}, \begin{bmatrix} \mathbf{b}_z \\ \mathbf{b}_w \end{bmatrix} \right\}, \quad (7.25)$$

$$Z \cap_{\mathbf{R}} Y = \left\{ \begin{bmatrix} \mathbf{M}_z & \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{G}_z & \mathbf{0} \end{bmatrix}, \mathbf{c}_z, \begin{bmatrix} \mathbf{S}_z & \mathbf{0} \\ \mathbf{0} & \mathbf{S}_y \\ \mathbf{R}\mathbf{M}_z & -\mathbf{M}_y \end{bmatrix}, \begin{bmatrix} \mathbf{A}_z & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_y \\ \mathbf{R}\mathbf{G}_z & -\mathbf{G}_y \end{bmatrix}, \begin{bmatrix} \mathbf{b}_z \\ \mathbf{b}_y \\ \mathbf{c}_y - \mathbf{R}\mathbf{c}_z \end{bmatrix} \right\}. \quad (7.26)$$

The demonstration of (7.24)–(7.26) is analogous to (3.8)–(3.10), which is presented in Scott et al. (2016). Moreover, note that every zonotope is a mixed zonotope, and also every constrained zonotope is a mixed zonotope. In the following, we demonstrate that other classes of sets are also mixed zonotopes.

Proposition 7.1. The Euclidean space \mathbb{R}^n is a mixed zonotope.

Proof. Let $\mathbf{M}_R \in \mathbb{R}^{n \times n\delta}$ be a full row rank matrix, and let $\mathbf{c}_R \in \mathbb{R}^n$. Since $\text{rank}(\mathbf{M}_R) = n$, for every $\mathbf{z} \in \mathbb{R}^n$ there exist at least one $\boldsymbol{\delta} \in \mathbb{R}^{n\delta}$ such that $\mathbf{z} = \mathbf{c}_R + \mathbf{M}_R \boldsymbol{\delta}$. Therefore, $\mathbb{R}^n \subseteq \{\mathbf{M}_R, _, \mathbf{c}_R, _, _, _\}$. On the other side, choose one $\boldsymbol{\delta} \in \mathbb{R}^{n\delta}$, and define $\mathbf{r} \triangleq \mathbf{c}_R + \mathbf{M}_R \boldsymbol{\delta}$. Since $\text{rank}(\mathbf{M}_R) = n$, there must exist $\mathbf{z} \in \mathbb{R}^n$ such that $\mathbf{z} = \mathbf{r}$. Therefore, $\{\mathbf{M}_R, _, \mathbf{c}_R, _, _, _\} \subseteq \mathbb{R}^n$, which implies that

$$\mathbb{R}^n = \{\mathbf{M}_R, _, \mathbf{c}_R, _, _, _\}. \quad (7.27)$$

■

Proposition 7.2. Every strip is a mixed zonotope.

Proof. Consider a strip $S = \{\mathbf{x} \in \mathbb{R}^n : |\boldsymbol{\rho}_s^T \mathbf{x} - d_s| \leq \sigma_s\}$, with $\boldsymbol{\rho}_s \in \mathbb{R}^n$, $d_s, \sigma_s \in \mathbb{R}$, $\sigma_s \geq 0$. This strip can be described in the MG-rep (7.23) with one bounded generator and n unbounded generators as follows. Note that an equivalent definition of the strip S is $\{\mathbf{x} \in \mathbb{R}^n : \boldsymbol{\rho}_s^T \mathbf{x} \in [-\sigma_s + d_s, \sigma_s + d_s]\}$. Therefore, by writing the interval $[-\sigma_s + d_s, \sigma_s + d_s]$ in G-rep as $\{\sigma_s, d_s\}$, or equivalently in MG-rep as $[-\sigma_s + d_s, \sigma_s + d_s] = \{_, \sigma_s, d_s, _, _, _\}$, and

noting that the set of real vectors \mathbb{R}^n can be described in MG-rep as $\mathbb{R}^n = \{\mathbf{I}_n, _, \mathbf{0}, _, _, _ \}$, therefore we have that, by definition of S and the generalized intersection (7.26), the strip S can be described in MG-rep as

$$S = \mathbb{R}^n \cap_{\rho^T} \{\sigma_s, d_s\} = \{\mathbf{I}_n, \mathbf{0}_{n \times 1}, \mathbf{0}_{n \times 1}, \rho^T, -\sigma_s, d_s\}. \quad (7.28)$$

■

Proposition 7.3. Every hyperplane is a mixed zonotope.

Proof. Consider a hyperplane $H \triangleq \{\mathbf{x} \in \mathbb{R}^n : \rho_s^T \mathbf{x} = d_s\}$, with $\rho_s \in \mathbb{R}^n$, $d_s \in \mathbb{R}$. Note that H is a particular case of a strip given by $\{\mathbf{x} \in \mathbb{R}^n : |\rho_s^T \mathbf{x} - d_s| \leq 0\}$ (i.e., a degenerated strip). Therefore, by Proposition 7.2, H can be described in MG-rep as

$$H = \mathbb{R}^n \cap_{\rho^T} \{0, d_s\} = \{\mathbf{I}_n, \mathbf{0}_{n \times 1}, \mathbf{0}_{n \times 1}, \rho^T, 0, d_s\} = \{\mathbf{I}_n, _, \mathbf{0}_{n \times 1}, \rho^T, _, d_s\}. \quad (7.29)$$

■

Figure 7.2 shows examples of unbounded sets described by mixed zonotopes. The ability of describing unbounded sets allows the set-based state estimation of the linear descriptor system (7.1) to be performed using MG-rep (see Section 7.5): (i) without the necessity of knowing a bounded initial set satisfying $\mathbf{x}_0 \in X_0$; and (ii) without requiring the existence of an admissible set X_a satisfying Assumption 7.1. A few other applications of mixed zonotopes are discussed in Section 7.4.2. In the following, we propose methods for complexity reduction of the MG-rep (7.23).

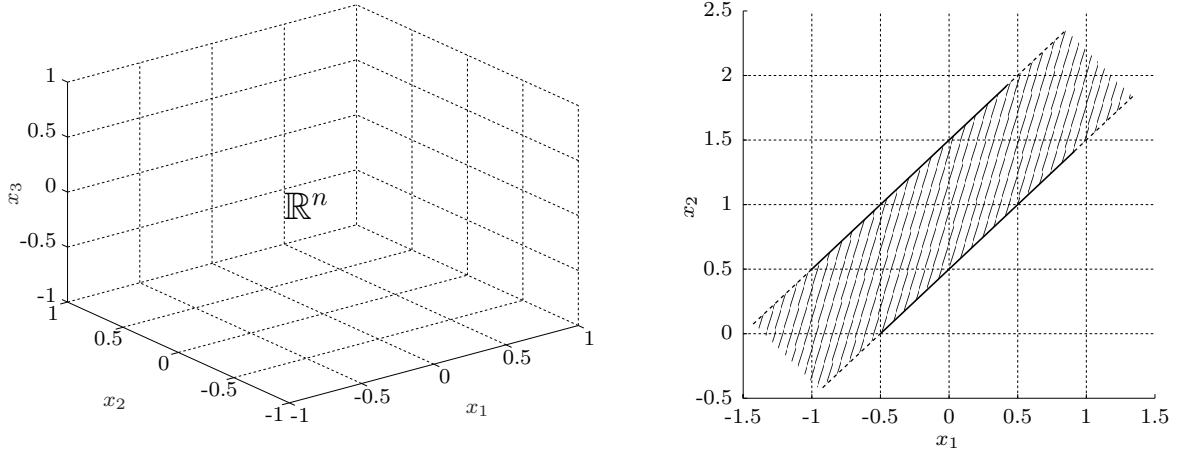


Figure 7.2: Examples of mixed zonotopes: the Euclidean space $\mathbb{R}^n = \{\mathbf{M}_R, _, \mathbf{c}_R, _, _, _ \}$ (left), where \mathbf{M}_R and \mathbf{c}_R can be chosen as \mathbf{I}_n and $\mathbf{0}_{n \times 1}$, respectively, and the strip $S = \{\mathbf{x} \in \mathbb{R}^2 : |[-1 \ 1]\mathbf{x} - 1| \leq 0.5\} = \{\mathbf{I}_2, \mathbf{0}_{2 \times 1}, \mathbf{0}_{2 \times 1}, [-1 \ 1], -0.5, 1\}$ (right).

7.4.1 Complexity reduction of mixed zonotopes

Similarly to zonotopes and constrained zonotopes, the set operations (7.24)–(7.26) result in a linear increase of number of bounded and unbounded generators, and constraints of the MG-rep (7.23). If these operations are performed iteratively, which is the case in set-based state estimation, the complexity of the set may increase indefinitely. This section presents methods for complexity reduction of mixed zonotopes, which outer-approximate a mixed zonotope by another one with fewer bounded and unbounded generators, and constraints.

7.4.1.1 Elimination of unbounded generators

The proposed procedure to eliminate unbounded generators is similar to the constraint elimination of constrained zonotopes described by Method 3.4, as explained below.

Let $Z = \{\mathbf{M}_z, \mathbf{G}_z, \mathbf{c}_z, \mathbf{S}_z, \mathbf{A}_z, \mathbf{b}_z\} \subseteq \mathbb{R}^n$ be a mixed zonotope with n_g bounded generators, n_δ unbounded generators, and n_c constraints. It will be shown that a number of unbounded generators equal to the row rank of \mathbf{S}_z can always be eliminated, and additionally, this procedure does not introduce conservativeness, i.e., the resulting set is equivalent to the previous one. The procedure is similar to Method 3.4, with the difference that the chosen generator variable to be eliminated is an unbounded one. This is done by solving the constraints $\mathbf{S}_z \boldsymbol{\delta} + \mathbf{A}_z \boldsymbol{\xi} = \mathbf{b}_z$ in the variables $\boldsymbol{\delta}$, and substituting these in $\mathbf{c}_z + \mathbf{M}_z \boldsymbol{\delta} + \mathbf{G}_z \boldsymbol{\xi}$ in (7.23). Note that this simultaneously eliminates the same number of constraints from Z . However, in contrast to eliminating a bounded generator, since $\boldsymbol{\delta}$ is unbounded, hence no information is lost in the process. Moreover, note that in the case that all the unbounded generators of Z can be eliminated through this procedure, then Z is a constrained zonotope. Algorithm 7.1 summarizes the proposed procedure.

Algorithm 7.1 Elimination of an unbounded generator from a mixed zonotope

- 1: Let $Z \triangleq \{\mathbf{M}, \mathbf{G}, \mathbf{c}, \mathbf{S}, \mathbf{A}, \mathbf{b}\}$. Choose one unbounded generator δ_j to eliminate from Z and one $i \in \{1, \dots, n_c\}$, such that $S_{i,j} \neq 0$.
 - 2: Solve the constraint $S_{i,:} \boldsymbol{\delta} + \mathbf{A}_{i,:} \boldsymbol{\xi} = \mathbf{b}_{i,:}$ for δ_j .
 - 3: Replace the result above in $\mathbf{S} \boldsymbol{\delta} + \mathbf{A} \boldsymbol{\xi} = \mathbf{b}$ and $\mathbf{c} + \mathbf{M} \boldsymbol{\delta} + \mathbf{G} \boldsymbol{\xi}$.
-

7.4.1.2 Constraint elimination

Let $Z = \{\mathbf{M}_z, \mathbf{G}_z, \mathbf{c}_z, \mathbf{S}_z, \mathbf{A}_z, \mathbf{b}_z\} \subseteq \mathbb{R}^n$. Constraint elimination of mixed zonotopes can be done exactly as with constrained zonotopes by using Method 3.4. However, since the removal of an unbounded generator is not conservative, if $\text{rank}(\mathbf{S}_z) > 0$ then $k_\delta = \text{rank}(\mathbf{S}_z)$ unbounded generators chosen to be eliminated together with k_δ constraints prior to removing any bounded generator from Z . Constraint elimination is done in this order for reduced conservativeness.

7.4.1.3 Elimination of bounded generators

To eliminate a bounded generator from $Z = \{\mathbf{M}_z, \mathbf{G}_z, \mathbf{c}_z, \mathbf{S}_z, \mathbf{A}_z, \mathbf{b}_z\} \subseteq \mathbb{R}^n$, we consider the case in which \mathbf{S}_z can be reduced to a matrix of zeros through the unbounded generator elimination procedure. In other words, let $Z^- = \{\mathbf{M}_z^-, \mathbf{G}_z^-, \mathbf{c}_z^-, \mathbf{S}_z^-, \mathbf{A}_z^-, \mathbf{b}_z^-\} \subseteq \mathbb{R}^n$ be the equivalent mixed zonotope obtained after removing $k_\delta = \text{rank}(\mathbf{S}_z)$ constraints and unbounded generators from Z . We assume that $\mathbf{S}_z^- = \mathbf{0}$. Then, the resulting mixed zonotope Z^- can be decoupled as $Z^- = \mathbf{c}_z^- \oplus \mathbf{M}_z^- \mathbb{R}^{n_\delta} \oplus \mathbf{G}_z^- B_\infty(\mathbf{A}_z^-, \mathbf{b}_z^-)$.

Therefore, bounded generator elimination can be done exactly as in Method 3.5, i.e., by first lifting $\mathbf{G}_z^- B_\infty(\mathbf{A}_z^-, \mathbf{b}_z^-)$ and then using a zonotope order reduction method. The resulting mixed zonotope is given by $\bar{Z} = \mathbf{c}_z^- \oplus \mathbf{M}_z^- \mathbb{R}^{n_\delta} \oplus \bar{\mathbf{G}}_z B_\infty(\bar{\mathbf{A}}_z, \mathbf{b}_z^-) \supseteq Z^-$, with $\bar{\mathbf{G}}_z$ and $\bar{\mathbf{A}}_z$ being the resulting matrices provided by the zonotope generator reduction method.

7.4.2 Concept examples

7.4.2.1 Exact intersection of a zonotope and a strip

Since both zonotope and strip are mixed zonotopes, then the intersection of a zonotope and a strip can be computed trivially and efficiently in MG-rep, using the generalized intersection (3.10) as illustrated below.

Consider the zonotope (Bravo et al., 2006)

$$Z = \left\{ \left[\begin{array}{ccc} 0.2812 & 0.1968 & 0.4235 \\ 0.0186 & -0.2063 & -0.2267 \end{array} \right], \left[\begin{array}{c} 0 \\ 0 \end{array} \right] \right\},$$

and the strip $S = \{\mathbf{x} \in \mathbb{R}^2 : |\boldsymbol{\rho}_s^T \mathbf{x} - d_s| \leq \sigma_s\}$, with $\boldsymbol{\rho}_s = [1 \ -1]^T$, $d_s = 0$, and $\sigma_s = 0.1$. The intersection $Z \cap S$ is computed by writing S in the MG-rep (7.28) and then using (3.10). The resulting set is

$$Z \cap S = \left\{ \left[\begin{array}{cc} 0 & 0 \\ 0 & 0 \end{array} \right], \left[\begin{array}{cccc} 0.2812 & 0.1968 & 0.4235 & 0 \\ 0.0186 & -0.2063 & -0.2267 & 0 \end{array} \right], \left[\begin{array}{c} 0 \\ 0 \end{array} \right], \right. \\ \left. \left[\begin{array}{cc} 1 & -1 \\ -1 & 1 \\ 0 & -1 \end{array} \right], \left[\begin{array}{cccc} 0 & 0 & 0 & -0.1 \\ 0.2812 & 0.1968 & 0.4235 & 0 \\ 0.0186 & -0.2063 & -0.2267 & 0 \end{array} \right], \left[\begin{array}{c} 0 \\ 0 \\ 0 \end{array} \right] \right\}.$$

Figure 7.3 shows the zonotope Z (gray) and strip S (blue), as well as the zonotope obtained using the intersection method (Bravo et al., 2006) (yellow) and the mixed zonotope $Z \cap S$ computed in MG-rep (red). The latter is obtained using Proposition 7.2 and (7.26), and as it can be noticed, it corresponds to the exact intersection of the zonotope Z and the strip S .

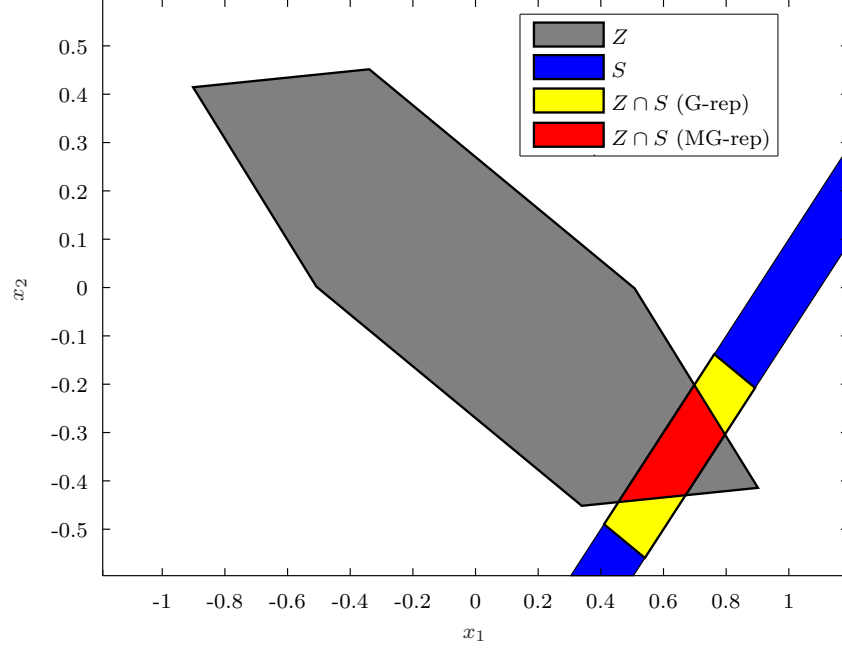


Figure 7.3: The zonotope Z (gray), the strip S (blue), the zonotope obtained using the intersection method (Bravo et al., 2006) (yellow), and the mixed zonotope $Z \cap S$ computed in MG-rep (red)

7.4.2.2 Unbounded intersection between strips

Let $S_1 \subset \mathbb{R}^n$ and $S_2 \subset \mathbb{R}^n$ be strips. The intersection $S_1 \cap S_2$ for $n > 2$ is not bounded in general case, and therefore cannot be computed using zonotopes or constrained zonotopes. On the other side, since S_1 and S_2 can be written in MG-rep using Proposition 7.2, the set $S_1 \cap S_2$ is then a mixed zonotope and can be computed using (7.26).

Let $S_1 \triangleq \{\mathbf{x} \in \mathbb{R}^3 : |\boldsymbol{\rho}_1^T \mathbf{x} - d_1| \leq \sigma_1\}$ and $S_2 \triangleq \{\mathbf{x} \in \mathbb{R}^3 : |\boldsymbol{\rho}_2^T \mathbf{x} - d_2| \leq \sigma_2\}$, with $\boldsymbol{\rho}_1 = [1 \ -1 \ 1]^T$, $\boldsymbol{\rho}_2 = [1 \ 1 \ 1]^T$, $d_1 = 1$, $d_2 = 1$, $\sigma_1 = 0.1$, and $\sigma_2 = 0.1$. Figure 7.4 shows the strips S_1 (blue) and S_2 (red), as well as the intersection $S_1 \cap S_2$ computed in MG-rep (magenta). Note that the intersection is unbounded, and therefore cannot be expressed as a constrained zonotope or a strip, while it can be described exactly using the MG-rep (7.23).

7.4.2.3 Two-side generalized intersection

Consider sets $Z \subset \mathbb{R}^{n_z}$, $Y \subset \mathbb{R}^{n_y}$ and real matrices $\mathbf{R}_z \in \mathbb{R}^{n_z \times n}$, $\mathbf{R}_y \in \mathbb{R}^{n_y \times n}$. Define the two-side generalized intersection between Z and Y as

$$Z \cap_{\mathbf{R}_z, \mathbf{R}_y} Y \triangleq \{\mathbf{x} \in \mathbb{R}^n : \mathbf{R}_z \mathbf{x} \in Z, \mathbf{R}_y \mathbf{x} \in Y\}. \quad (7.30)$$

Note that the generalized intersection (3.3) is a particular case of (7.30) in which $\mathbf{R}_z = \mathbf{I}$ and $\mathbf{R}_y = \mathbf{R}$. When Z and Y are in MG-rep, the two-side generalized intersection

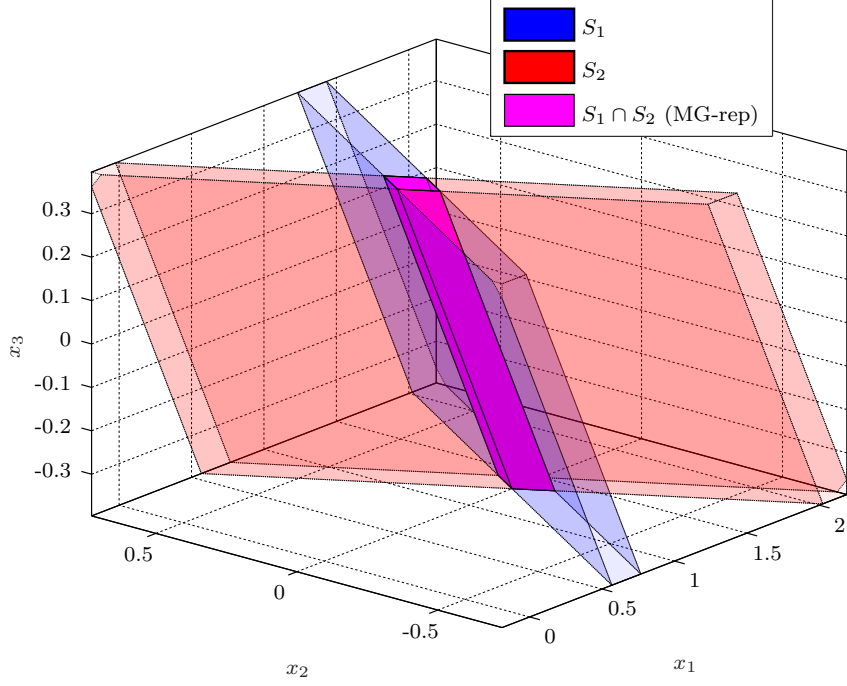


Figure 7.4: The strips S_1 (blue) and S_2 (red), and the intersection $S_1 \cap S_2$ computed in MG-rep (magenta).

(7.30) can be computed as follows. Let $Z \cap_{\mathbb{R}_z, \mathbb{R}_y} Y$ be rewritten as

$$Z \cap_{\mathbb{R}_z, \mathbb{R}_y} Y = (\mathbb{R}^n \cap_{\mathbb{R}_z} Z) \cap_{\mathbb{R}_y} Y.$$

Let $Z = \{\mathbf{M}_z, \mathbf{G}_z, \mathbf{c}_z, \mathbf{S}_z, \mathbf{A}_z, \mathbf{b}_z\}$, $Y = \{\mathbf{M}_y, \mathbf{G}_y, \mathbf{c}_y, \mathbf{S}_y, \mathbf{A}_y, \mathbf{b}_y\}$, and let \mathbb{R}^n be given in MG-rep as $\mathbb{R}^n = \{\mathbf{I}_n, _, \mathbf{0}, _, _, _ \}$. Then, using (3.3) twice leads to

$$Z \cap_{\mathbb{R}_z, \mathbb{R}_y} Y = \left\{ \left[\mathbf{I}_n \ \mathbf{0} \ \mathbf{0} \right], \left[\mathbf{0} \ \mathbf{0} \right], \mathbf{0}, \begin{bmatrix} \mathbf{0} & \mathbf{S}_z & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{S}_y \\ \mathbf{R}_z & -\mathbf{M}_z & \mathbf{0} \\ \mathbf{R}_y & \mathbf{0} & -\mathbf{M}_y \end{bmatrix}, \begin{bmatrix} \mathbf{A}_z & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_y \\ -\mathbf{G}_z & \mathbf{0} \\ \mathbf{0} & -\mathbf{G}_y \end{bmatrix}, \begin{bmatrix} \mathbf{b}_z \\ \mathbf{b}_y \\ \mathbf{c}_z \\ \mathbf{c}_y \end{bmatrix} \right\}.$$

7.5 Set-based state estimation of LDS using mixed zonotopes

This section presents a new method for set-based state estimation of system (7.1) using mixed zonotopes. Given an initial condition X_0 and an input \mathbf{u}_k with $k \geq 0$, let

$$\hat{X}_0 = \{\mathbf{x} \in X_0 : \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u}_0 + \mathbf{D}_v \mathbf{v} = \mathbf{y}_0, \mathbf{v} \in V\}. \quad (7.31)$$

The method proposed in this section is also based on singular value decomposition. Let $\mathbf{E} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$, where \mathbf{U} and \mathbf{V} are invertible by construction. Since \mathbf{E} is square, then $\mathbf{\Sigma}$

is also square. Without loss of generality, let Σ be arranged as $\Sigma = \text{blkdiag}(\tilde{\Sigma}, \mathbf{0})$, where $\tilde{\Sigma} \in \mathbb{R}^{n_z \times n_z}$ is diagonal with all the $n_z = \text{rank}(\mathbf{E})$ nonzero singular values of \mathbf{E} . Moreover, let $\mathbf{z}_k = (\tilde{\mathbf{z}}_k, \check{\mathbf{z}}_k) = \mathbf{T}^{-1}\mathbf{x}_k$, $\tilde{\mathbf{z}}_k \in \mathbb{R}^{n_z}$, $\check{\mathbf{z}}_k \in \mathbb{R}^{n-n_z}$,

$$\begin{aligned} \begin{bmatrix} \tilde{\mathbf{A}} \\ \check{\mathbf{A}} \end{bmatrix} &= \begin{bmatrix} \tilde{\Sigma}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \mathbf{U}^{-1}\mathbf{A}\mathbf{T}, \\ \begin{bmatrix} \tilde{\mathbf{B}} \\ \check{\mathbf{B}} \end{bmatrix} &= \begin{bmatrix} \tilde{\Sigma}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \mathbf{U}^{-1}\mathbf{B}, \quad \begin{bmatrix} \tilde{\mathbf{B}}_w \\ \check{\mathbf{B}}_w \end{bmatrix} = \begin{bmatrix} \tilde{\Sigma}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \mathbf{U}^{-1}\mathbf{B}_w, \end{aligned} \quad (7.32)$$

with $\mathbf{T} = (\mathbf{V}^T)^{-1}$, $\tilde{\mathbf{A}} \in \mathbb{R}^{n_z \times n}$, $\check{\mathbf{A}} \in \mathbb{R}^{(n-n_z) \times n}$, $\tilde{\mathbf{B}} \in \mathbb{R}^{n_z \times n_u}$, $\check{\mathbf{B}} \in \mathbb{R}^{(n-n_z) \times n_u}$. Then, as in Section 7.2, system (7.1) can be rewritten as

$$\tilde{\mathbf{z}}_k = \tilde{\mathbf{A}}\mathbf{z}_{k-1} + \tilde{\mathbf{B}}\mathbf{u}_{k-1} + \tilde{\mathbf{B}}_w\mathbf{w}_{k-1}, \quad (7.33a)$$

$$\mathbf{0} = \check{\mathbf{A}}\mathbf{z}_k + \check{\mathbf{B}}\mathbf{u}_k + \check{\mathbf{B}}_w\mathbf{w}_k, \quad (7.33b)$$

$$\mathbf{y}_k = \mathbf{C}\mathbf{T}\mathbf{z}_k + \mathbf{D}\mathbf{u}_k + \mathbf{D}_v\mathbf{v}_k. \quad (7.33c)$$

Let $W = \{\mathbf{M}_w, \mathbf{G}_w, \mathbf{c}_w, \mathbf{S}_w, \mathbf{A}_w, \mathbf{b}_w\}$, and \hat{X}_0 given by (7.31). From (7.33b), the state \mathbf{z}_0 must satisfy $\check{\mathbf{A}}\mathbf{z}_0 + \check{\mathbf{B}}\mathbf{u}_0 + \check{\mathbf{B}}_w\mathbf{w}_0 = \mathbf{0}$. This constraint can be incorporated in the MG-rep of the initial set \hat{Z}_0 as follows. Let $\mathbf{T}^{-1}\hat{X}_0 \triangleq \{\mathbf{M}_0, \mathbf{G}_0, \mathbf{c}_0, \mathbf{S}_0, \mathbf{A}_0, \mathbf{b}_0\}$. Then, $\hat{Z}_0 \triangleq \{\hat{\mathbf{M}}_0, \hat{\mathbf{G}}_0, \hat{\mathbf{c}}_0, \hat{\mathbf{S}}_0, \hat{\mathbf{A}}_0, \hat{\mathbf{b}}_0\}$, with $\hat{\mathbf{M}}_0 = [\mathbf{M}_0 \ \mathbf{0}]$, $\hat{\mathbf{G}}_0 = [\mathbf{G}_0 \ \mathbf{0}]$, $\hat{\mathbf{c}}_0 = \mathbf{c}_0$,

$$\hat{\mathbf{S}}_0 = \begin{bmatrix} \mathbf{S}_0 & \mathbf{0} \\ \mathbf{0} & \mathbf{S}_w \\ \check{\mathbf{A}}\mathbf{M}_0 & \check{\mathbf{B}}_w\mathbf{M}_w \end{bmatrix}, \quad \hat{\mathbf{A}}_0 = \begin{bmatrix} \mathbf{A}_0 & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_w \\ \check{\mathbf{A}}\mathbf{G}_0 & \check{\mathbf{B}}_w\mathbf{G}_w \end{bmatrix}, \quad \hat{\mathbf{b}}_0 = \begin{bmatrix} \mathbf{b}_0 \\ \mathbf{b}_w \\ -\check{\mathbf{A}}\mathbf{c}_0 - \check{\mathbf{B}}_w\mathbf{c}_w - \check{\mathbf{B}}\mathbf{u}_0 \end{bmatrix}.$$

Note that the extra columns in $\hat{\mathbf{G}}_0$ and $\hat{\mathbf{A}}_0$ come from $\mathbf{w}_0 \in W$. Moreover, from (7.33), the variables $\tilde{\mathbf{z}}_k$ are fully determined by (7.33a), while $\check{\mathbf{z}}_k$ are obtained a posteriori by $\check{\mathbf{A}}\mathbf{z}_k + \check{\mathbf{B}}\mathbf{u}_k + \check{\mathbf{B}}_w\mathbf{w}_k = \mathbf{0}$.

An effective enclosure of the prediction step for the descriptor system (7.33) can be obtained in MG-rep as follows.

Lemma 7.3. Consider the transformed state-space (7.33). Let $\mathbf{z}_{k-1} \in \hat{Z}_{k-1} \triangleq \{\hat{\mathbf{M}}_{k-1}, \hat{\mathbf{G}}_{k-1}, \hat{\mathbf{c}}_{k-1}, \hat{\mathbf{S}}_{k-1}, \hat{\mathbf{A}}_{k-1}, \hat{\mathbf{b}}_{k-1}\}$, and $\mathbf{w}_{k-1}, \mathbf{w}_k \in W \triangleq \{\mathbf{M}_w, \mathbf{G}_w, \mathbf{c}_w, \mathbf{S}_w, \mathbf{A}_w, \mathbf{b}_w\}$, for all $k \geq 1$. Moreover, let $\mathbf{c}_R \in \mathbb{R}^{n-n_z}$, and let $\mathbf{M}_R \in \mathbb{R}^{(n-n_z) \times n_{\delta R}}$ be a full row rank matrix with $n_{\delta} \geq n - n_z$. Then $\mathbf{z}_k \in \bar{Z}_k \triangleq \{\bar{\mathbf{M}}_k, \bar{\mathbf{G}}_k, \bar{\mathbf{c}}_k, \bar{\mathbf{S}}_k, \bar{\mathbf{A}}_k, \bar{\mathbf{b}}_k\}$, with

$$\bar{\mathbf{M}}_k = \begin{bmatrix} \tilde{\mathbf{A}}\hat{\mathbf{M}}_{k-1} & \tilde{\mathbf{B}}_w\mathbf{M}_w & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{M}_R \end{bmatrix}, \quad \bar{\mathbf{G}}_k = \begin{bmatrix} \tilde{\mathbf{A}}\hat{\mathbf{G}}_{k-1} & \tilde{\mathbf{B}}_w\mathbf{G}_w & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}, \quad \bar{\mathbf{c}}_k = \begin{bmatrix} \tilde{\mathbf{A}}\hat{\mathbf{c}}_{k-1} + \tilde{\mathbf{B}}\mathbf{u}_{k-1} + \tilde{\mathbf{B}}_w\mathbf{c}_w \\ \mathbf{c}_R \end{bmatrix},$$

$$\bar{\mathbf{S}}_k = \begin{bmatrix} \hat{\mathbf{S}}_{k-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{S}_w & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{S}_w & \mathbf{0} \\ \check{\mathbf{A}} \begin{bmatrix} \tilde{\mathbf{A}}\hat{\mathbf{M}}_{k-1} \\ \mathbf{0} \end{bmatrix} & \check{\mathbf{A}} \begin{bmatrix} \tilde{\mathbf{B}}_w\mathbf{M}_w \\ \mathbf{0} \end{bmatrix} & \mathbf{0} & \check{\mathbf{A}} \begin{bmatrix} \mathbf{0} \\ \mathbf{M}_R \end{bmatrix} \end{bmatrix}, \bar{\mathbf{A}}_k = \begin{bmatrix} \hat{\mathbf{A}}_{k-1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_w & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{A}_w \\ \check{\mathbf{A}} \begin{bmatrix} \tilde{\mathbf{A}}\hat{\mathbf{G}}_{k-1} \\ \mathbf{0} \end{bmatrix} & \check{\mathbf{A}} \begin{bmatrix} \tilde{\mathbf{B}}_w\mathbf{G}_w \\ \mathbf{0} \end{bmatrix} & \check{\mathbf{B}}_w\mathbf{G}_w \end{bmatrix},$$

$$\bar{\mathbf{b}}_k = \begin{bmatrix} \hat{\mathbf{b}}_{k-1} \\ \mathbf{b}_w \\ \mathbf{b}_w \\ -\check{\mathbf{A}} \begin{bmatrix} \tilde{\mathbf{A}}\hat{\mathbf{c}}_{k-1} + \tilde{\mathbf{B}}\mathbf{u}_{k-1} + \tilde{\mathbf{B}}_w\mathbf{c}_w \\ \mathbf{c}_R \end{bmatrix} - \check{\mathbf{B}}\mathbf{u}_k - \check{\mathbf{B}}_w\mathbf{c}_w \end{bmatrix}.$$

Proof. Since by assumption $(\mathbf{z}_{k-1}, \mathbf{w}_{k-1}, \mathbf{w}_k) \in \hat{Z}_{k-1} \times W \times W$, there exists $((\boldsymbol{\xi}_{k-1}, \boldsymbol{\delta}_{k-1}), (\boldsymbol{\varphi}_{k-1}, \boldsymbol{\vartheta}_{k-1}), (\boldsymbol{\varphi}_k, \boldsymbol{\vartheta}_k)) \in \{(\boldsymbol{\xi}, \boldsymbol{\delta}) \in B_\infty^{n_g} \times \mathbb{R}^{n_\delta} : \hat{\mathbf{S}}_{k-1}\boldsymbol{\delta} + \hat{\mathbf{A}}_{k-1}\boldsymbol{\xi} = \hat{\mathbf{b}}_{k-1}\} \times \{(\boldsymbol{\varphi}, \boldsymbol{\vartheta}) \in B_\infty^{n_{gw}} \times \mathbb{R}^{n_{\delta w}} : \mathbf{S}_w\boldsymbol{\vartheta} + \mathbf{A}_w\boldsymbol{\varphi} = \mathbf{b}_w\} \times \{(\boldsymbol{\varphi}, \boldsymbol{\vartheta}) \in B_\infty^{n_{gw}} \times \mathbb{R}^{n_{\delta w}} : \mathbf{S}_w\boldsymbol{\vartheta} + \mathbf{A}_w\boldsymbol{\varphi} = \mathbf{b}_w\}$ such that $\mathbf{z}_{k-1} = \hat{\mathbf{c}}_{k-1} + \hat{\mathbf{G}}_{k-1}\boldsymbol{\xi}_{k-1} + \hat{\mathbf{M}}_{k-1}\boldsymbol{\delta}_{k-1}$, $\mathbf{w}_{k-1} = \mathbf{c}_w + \mathbf{S}_w\boldsymbol{\vartheta}_{k-1} + \mathbf{G}_w\boldsymbol{\varphi}_{k-1}$, and $\mathbf{w}_k = \mathbf{c}_w + \mathbf{S}_w\boldsymbol{\vartheta}_k + \mathbf{G}_w\boldsymbol{\varphi}_k$. Moreover, since $\check{\mathbf{z}}_k \in \mathbb{R}^{n-n_z}$, then there must exist $\boldsymbol{\delta}_R \in \mathbb{R}^{n_{\delta_R}}$ such that $\check{\mathbf{z}}_k = \mathbf{c}_R + \mathbf{M}_R\boldsymbol{\delta}_R$ for any $\mathbf{c}_R \in \mathbb{R}^{n-n_z}$, and any full row rank $\mathbf{M}_R \in \mathbb{R}^{(n-n_z) \times n_{\delta_R}}$ with $n_{\delta_R} \geq n - n_z$. Then, substituting these equalities in (7.33a) leads to

$$\begin{bmatrix} \check{\mathbf{z}}_k \\ \check{\mathbf{z}}_k \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{A}}\hat{\mathbf{c}}_{k-1} + \tilde{\mathbf{B}}\mathbf{u}_{k-1} + \tilde{\mathbf{B}}_w\mathbf{c}_w \\ \mathbf{c}_R \end{bmatrix} + \begin{bmatrix} \tilde{\mathbf{A}}\hat{\mathbf{G}}_{k-1} & \tilde{\mathbf{B}}_w\mathbf{G}_w \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{\xi}_{k-1} \\ \boldsymbol{\varphi}_{k-1} \end{bmatrix} + \begin{bmatrix} \tilde{\mathbf{A}}\hat{\mathbf{M}}_{k-1} & \tilde{\mathbf{B}}_w\mathbf{M}_w & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{M}_R \end{bmatrix} \begin{bmatrix} \boldsymbol{\delta}_{k-1} \\ \boldsymbol{\vartheta}_{k-1} \\ \boldsymbol{\delta}_R \end{bmatrix}. \quad (7.34)$$

From the constraint (7.33b), we have that

$$\begin{aligned} & \check{\mathbf{B}}_w\mathbf{c}_w + \check{\mathbf{B}}_w\mathbf{G}_w\boldsymbol{\varphi}_k + \check{\mathbf{B}}_w\mathbf{M}_w\boldsymbol{\vartheta}_k + \check{\mathbf{B}}\mathbf{u}_k + \check{\mathbf{A}} \begin{bmatrix} \tilde{\mathbf{A}}\hat{\mathbf{c}}_{k-1} + \tilde{\mathbf{B}}\mathbf{u}_{k-1} + \tilde{\mathbf{B}}_w\mathbf{c}_w \\ \mathbf{c}_R \end{bmatrix} \\ & + \check{\mathbf{A}} \begin{bmatrix} \tilde{\mathbf{A}}\hat{\mathbf{G}}_{k-1} & \tilde{\mathbf{B}}_w\mathbf{G}_w \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{\xi}_{k-1} \\ \boldsymbol{\varphi}_{k-1} \end{bmatrix} + \check{\mathbf{A}} \begin{bmatrix} \tilde{\mathbf{A}}\hat{\mathbf{M}}_{k-1} & \tilde{\mathbf{B}}_w\mathbf{M}_w & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{M}_R \end{bmatrix} \begin{bmatrix} \boldsymbol{\delta}_{k-1} \\ \boldsymbol{\vartheta}_{k-1} \\ \boldsymbol{\delta}_R \end{bmatrix} = \mathbf{0}. \end{aligned} \quad (7.35)$$

Rearranging (7.34) and (7.35), grouping the variables $(\boldsymbol{\xi}_{k-1}, \boldsymbol{\varphi}_{k-1}, \boldsymbol{\varphi}_k)$ and $(\boldsymbol{\delta}_{k-1}, \boldsymbol{\vartheta}_{k-1}, \boldsymbol{\vartheta}_k, \boldsymbol{\delta}_R)$, and writing in the MG-rep (7.23), proves the lemma. \blacksquare

Remark 7.4. The enclosure \bar{Z}_k provided by Lemma 7.3 has $n_\delta + 2n_{\delta_w} + n_{\delta_R}$ unbounded generators, $n_g + 2n_{g_w}$ bounded generators, and $n_c + 2n_{c_w} + n - n_z$ constraints.

Lemma 7.3 provides a predicted enclosure of the state \mathbf{z}_k in which the equality constraints (7.7b) are directly incorporated. As in the case of CZs, this is possible thanks

to the fact that MZs incorporate equality constraints. The prediction-update algorithm proposed for descriptor systems consists in the computation of MZs \bar{Z}_k , \hat{Z}_k , and \hat{X}_k , such that

$$\bar{Z}_k = \{\bar{\mathbf{M}}_k, \bar{\mathbf{G}}_k, \bar{\mathbf{c}}_k, \bar{\mathbf{S}}_k, \bar{\mathbf{A}}_k, \bar{\mathbf{b}}_k\}, \quad (7.36)$$

$$\hat{Z}_k = \bar{Z}_k \cap_{\text{CT}} ((\mathbf{y}_k - \mathbf{D}_u \mathbf{u}_k) \oplus (-\mathbf{D}_v V_k)), \quad (7.37)$$

$$\hat{X}_k = \mathbf{T} \hat{Z}_k. \quad (7.38)$$

For this algorithm, the initial set is \hat{Z}_0 . The algorithm (7.36)–(7.38) operates recursively with \bar{Z}_k and \hat{Z}_k for $k \geq 1$ in the transformed state-space (7.33), while the estimated enclosure in the original state-space (7.1) is given by \hat{X}_k .

Remark 7.5. By construction, the MG-rep (7.36) corresponds to the exact feasible state set of (7.33) at k for the known state and uncertainty bounds. In addition, (7.37)–(7.38) can be computed exactly. However, in practice, in order to limit the complexity of the resulting sets these are outer approximated by using order reduction algorithms. In this case, equalities (7.36)–(7.38) are replaced by the relation \supset .

Remark 7.6. In contrast to constrained zonotopes, state estimation of the descriptor system (7.1) using mixed zonotopes does not require knowledge of a bounded set enclosing all the states (Assumption 7.1). In other words, this means that the method proposed in this section can be applied also to unstable descriptor systems.

Remark 7.7. Since X_0 is described in MG-rep (7.23), the method proposed in this section then does not require the knowledge of a bounded initial set. The enclosures \hat{X}_k will be bounded depending on observability properties of the system.

7.6 Active fault diagnosis of LDS using mixed zonotopes

The previous section presented a method to address the problem of the set-based estimation of descriptor systems using mixed zonotopes. In the following, this tool is used in the design of an MZ-based AFD method accounting for a finite number of possible abrupt faults.

Let $\vec{\mathbf{u}} = (\mathbf{u}_0, \dots, \mathbf{u}_N) \in \mathbb{R}^{(N+1)n_u}$, $\vec{\mathbf{w}} = (\mathbf{w}_0, \dots, \mathbf{w}_N) \in \mathbb{R}^{(N+1)n_w}$, and $\vec{W} = W \times \dots \times W$. Let $\mathbf{z}_k = (\tilde{\mathbf{z}}_k, \check{\mathbf{z}}_k) = (\mathbf{T}^{-1} \mathbf{x}_k, \mathbf{w}_k)$, $\tilde{\mathbf{z}}_k \in \mathbb{R}^{n_z}$, $\check{\mathbf{z}}_k \in \mathbb{R}^{n+n_w-n_z}$, with $\mathbf{T}^{[i]} = ((\mathbf{V}^{[i]})^T)^{-1}$, $\mathbf{V}^{[i]}$ being obtained from the SVD $\mathbf{E}^{[i]} = \mathbf{U}^{[i]} \boldsymbol{\Sigma}^{[i]} (\mathbf{V}^{[i]})^T$. Then, (7.2) can be rewritten as

$$\tilde{\mathbf{z}}_k^{[i]} = \tilde{\mathbf{A}}_z^{[i]} \mathbf{z}_{k-1}^{[i]} + \tilde{\mathbf{B}}^{[i]} \mathbf{u}_{k-1}, \quad (7.39a)$$

$$\mathbf{0} = \check{\mathbf{A}}_z^{[i]} \mathbf{z}_k^{[i]} + \check{\mathbf{B}}^{[i]} \mathbf{u}_k, \quad (7.39b)$$

$$\mathbf{y}_k^{[i]} = \mathbf{F}^{[i]} \mathbf{z}_k^{[i]} + \mathbf{D}^{[i]} \mathbf{u}_k + \mathbf{D}_v^{[i]} \mathbf{v}_k, \quad (7.39c)$$

with $\mathbf{F}^{[i]} = \mathbf{C}^{[i]} \mathbf{T}^{[i]} \mathbf{L}$, where $\mathbf{L} = [\mathbf{I}_n \ \mathbf{0}_{n \times n_w}]$, $\tilde{\mathbf{A}}_z^{[i]} = [\tilde{\mathbf{A}}^{[i]} \ \tilde{\mathbf{B}}_w^{[i]}]$, and $\check{\mathbf{A}}_z^{[i]} = [\check{\mathbf{A}}^{[i]} \ \check{\mathbf{B}}_w^{[i]}]$. Note that the $(\tilde{\cdot})$ and $(\check{\cdot})$ variables are defined according to (7.32) for each $i \in \mathcal{M}$.

For each model $i \in \mathcal{M}$, consider the set $\{\mathbf{M}_z^{[i]}, \mathbf{G}_z^{[i]}, \mathbf{c}_z^{[i]}, \mathbf{S}_z^{[i]}, \mathbf{A}_z^{[i]}, \mathbf{b}_z^{[i]}\} = (\mathbf{T}^{[i]})^{-1} X_0 \times W$, and define the initial feasible set $Z_0^{[i]}(\mathbf{u}_0) = \{\mathbf{z} \in (\mathbf{T}^{[i]})^{-1} X_0 \times W : (7.39b) \text{ holds for } k = 0\}$. This set is given by $Z_0^{[i]}(\mathbf{u}_0) = \{\mathbf{M}_0^{[i]}, \mathbf{G}_0^{[i]}, \mathbf{c}_0^{[i]}, \mathbf{S}_0^{[i]}, \mathbf{A}_0^{[i]}, \mathbf{b}_0^{[i]}(\mathbf{u}_0)\}$, where $\mathbf{M}_0^{[i]} = \mathbf{M}_z^{[i]}$, $\mathbf{G}_0^{[i]} = \mathbf{G}_z^{[i]}$, $\mathbf{c}_0^{[i]} = \mathbf{c}_z^{[i]}$,

$$\mathbf{S}_0^{[i]} = \begin{bmatrix} \mathbf{S}_z^{[i]} \\ \check{\mathbf{A}}_z^{[i]} \mathbf{M}_0^{[i]} \end{bmatrix}, \quad \mathbf{A}_0^{[i]} = \begin{bmatrix} \mathbf{A}_z^{[i]} \\ \check{\mathbf{A}}_z^{[i]} \mathbf{G}_0^{[i]} \end{bmatrix}, \quad \mathbf{b}_0^{[i]}(\mathbf{u}_0) = \begin{bmatrix} \mathbf{b}_z^{[i]} \\ -\check{\mathbf{A}}_z^{[i]} \mathbf{c}_0^{[i]} - \check{\mathbf{B}}^{[i]} \mathbf{u}_0 \end{bmatrix}. \quad (7.40)$$

Let $\mathbb{R}^{n-n_z} = \{\mathbf{M}_R, _, \mathbf{c}_R, _, _, _ \}$, with \mathbf{M}_R full row rank, let $W = \{\mathbf{M}_w, \mathbf{G}_w, \mathbf{c}_w, \mathbf{S}_w, \mathbf{A}_w, \mathbf{b}_w\}$, and $Z_A = \{\mathbf{M}_A, \mathbf{G}_A, \mathbf{c}_A, \mathbf{S}_A, \mathbf{A}_A, \mathbf{b}_A\} \triangleq \mathbb{R}^{n-n_z} \times W$. In addition, define the solution mappings $(\phi_k^{[i]}, \psi_k^{[i]}) : \mathbb{R}^{(k+1)n_u} \times \mathbb{R}^n \times \mathbb{R}^{(k+1)n_w} \times \mathbb{R}^{n_v} \rightarrow \mathbb{R}^{n+n_w} \times \mathbb{R}^{n_y}$ such that $\phi_k^{[i]}(\vec{\mathbf{u}}, \mathbf{z}_0, \vec{\mathbf{w}})$ and $\psi_k^{[i]}(\vec{\mathbf{u}}, \mathbf{z}_0, \vec{\mathbf{w}}, \mathbf{v}_k)$ are the state and output of (7.13) at k , respectively. Then, for each $i \in \mathcal{M}$, define state and output reachable sets at time k as

$$\begin{aligned} Z_k^{[i]}(\vec{\mathbf{u}}) &\triangleq \{\phi_k^{[i]}(\vec{\mathbf{u}}, \mathbf{z}_0, \vec{\mathbf{w}}) : (\mathbf{z}_0^{[i]}, \vec{\mathbf{w}}) \in Z_0^{[i]}(\mathbf{u}_0) \times \vec{W}\}, \\ Y_k^{[i]}(\vec{\mathbf{u}}) &\triangleq \{\psi_k^{[i]}(\vec{\mathbf{u}}, \mathbf{z}_0, \vec{\mathbf{w}}, \mathbf{v}_k) : (\mathbf{z}_0, \vec{\mathbf{w}}, \mathbf{v}_k) \in Z_0^{[i]}(\mathbf{u}_0) \times \vec{W} \times V\}. \end{aligned}$$

Let $N \in \mathbb{N}$, $N \geq 0$, and define $\vec{\phi}^{[i]}(\vec{\mathbf{u}}, \mathbf{z}_0, \vec{\mathbf{w}}) \triangleq (\phi_0^{[i]}, \phi_1^{[i]}, \dots, \phi_N^{[i]})(\vec{\mathbf{u}}, \mathbf{z}_0, \vec{\mathbf{w}})$, and $\vec{\psi}^{[i]}(\vec{\mathbf{u}}, \mathbf{z}_0, \vec{\mathbf{w}}, \vec{\mathbf{v}}) \triangleq (\psi_0^{[i]}, \psi_1^{[i]}, \dots, \psi_N^{[i]})(\vec{\mathbf{u}}, \mathbf{z}_0, \vec{\mathbf{w}}, \vec{\mathbf{v}})$. Given the time horizon $k \in [0, N]$, we define the *state and output reachable tubes* as

$$\begin{aligned} \vec{Z}^{[i]}(\vec{\mathbf{u}}) &\triangleq \{\vec{\phi}^{[i]}(\vec{\mathbf{u}}, \mathbf{z}_0, \vec{\mathbf{w}}) : (\mathbf{z}_0^{[i]}, \vec{\mathbf{w}}) \in Z_0^{[i]}(\mathbf{u}_0) \times \vec{W}\}, \\ \vec{Y}^{[i]}(\vec{\mathbf{u}}) &\triangleq \{\vec{\psi}^{[i]}(\vec{\mathbf{u}}, \mathbf{z}_0, \vec{\mathbf{w}}, \vec{\mathbf{v}}) : (\mathbf{z}_0, \vec{\mathbf{w}}, \vec{\mathbf{v}}) \in Z_0^{[i]}(\mathbf{u}_0) \times \vec{W} \times \vec{V}\}. \end{aligned}$$

7.6.1 State reachable tube

Let $Z_z = \{\mathbf{M}_z, \mathbf{G}_z, \mathbf{c}_z, \mathbf{S}_z, \mathbf{A}_z, \mathbf{b}_z\} \triangleq ((\mathbf{T}^{[i]})^{-1} X_0) \times W$, and $Z_0 \triangleq \{\mathbf{z} \in Z_z : \mathbf{0} = \check{\mathbf{A}}_z \mathbf{z} + \check{\mathbf{B}} \mathbf{u}_0\}$. Moreover, let $\mathbb{R}^{n-n_z} = \{\mathbf{M}_R, _, \mathbf{c}_R, _, _, _ \}$, with \mathbf{M}_R full row rank, let $W = \{\mathbf{M}_w, \mathbf{G}_w, \mathbf{c}_w, \mathbf{S}_w, \mathbf{A}_w, \mathbf{b}_w\}$, and $Z_A = \{\mathbf{M}_A, \mathbf{G}_A, \mathbf{c}_A, \mathbf{S}_A, \mathbf{A}_A, \mathbf{b}_A\} \triangleq \mathbb{R}^{n-n_z} \times W$.

Using (7.24)–(7.25), the set $Z_k^{[i]}(\vec{\mathbf{u}})$ is given by the MG-rep $\{\mathbf{M}_k^{[i]}, \mathbf{G}_k^{[i]}, \mathbf{c}_k^{[i]}(\vec{\mathbf{u}}), \mathbf{S}_k^{[i]}, \mathbf{A}_k^{[i]}, \mathbf{b}_k^{[i]}(\vec{\mathbf{u}})\}$, where $\mathbf{M}_k^{[i]}, \mathbf{G}_k^{[i]}, \mathbf{c}_k^{[i]}(\vec{\mathbf{u}}), \mathbf{S}_k^{[i]}, \mathbf{A}_k^{[i]}$, and $\mathbf{b}_k^{[i]}(\vec{\mathbf{u}})$ are given by the recursive relations

$$\begin{aligned} \mathbf{c}_k^{[i]}(\vec{\mathbf{u}}) &= \begin{bmatrix} \check{\mathbf{A}}_z^{[i]} \mathbf{c}_{k-1}^{[i]}(\vec{\mathbf{u}}) + \check{\mathbf{B}}^{[i]} \mathbf{u}_{k-1} \\ \mathbf{c}_A \end{bmatrix}, \quad \mathbf{M}_k^{[i]} = \begin{bmatrix} \check{\mathbf{A}}_z^{[i]} \mathbf{M}_{k-1}^{[i]} & \mathbf{0} \\ \mathbf{0} & \mathbf{M}_A \end{bmatrix}, \quad \mathbf{G}_k^{[i]} = \begin{bmatrix} \check{\mathbf{A}}_z^{[i]} \mathbf{G}_{k-1}^{[i]} & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_A \end{bmatrix}, \\ \mathbf{S}_k^{[i]} &= \begin{bmatrix} \mathbf{S}_{k-1}^{[i]} & \mathbf{0} \\ \mathbf{0} & \mathbf{S}_A \\ \check{\mathbf{A}}_z^{[i]} \mathbf{M}_k^{[i]} \end{bmatrix}, \quad \mathbf{A}_k^{[i]} = \begin{bmatrix} \mathbf{A}_{k-1}^{[i]} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_A \\ \check{\mathbf{A}}_z^{[i]} \mathbf{G}_k^{[i]} \end{bmatrix}, \quad \mathbf{b}_k^{[i]}(\vec{\mathbf{u}}) = \begin{bmatrix} \mathbf{b}_{k-1}^{[i]}(\vec{\mathbf{u}}) \\ \mathbf{b}_A \\ -\check{\mathbf{A}}_z^{[i]} \mathbf{c}_k^{[i]}(\vec{\mathbf{u}}) - \check{\mathbf{B}}^{[i]} \mathbf{u}_k \end{bmatrix}, \end{aligned}$$

for $k = 1, 2, \dots, N$. Note that the third constraint in $(\mathbf{S}_k^{[i]}, \mathbf{A}_k^{[i]}, \mathbf{b}_k^{[i]}(\vec{\mathbf{u}}))$, comes from the fact that (7.39b) must hold. Using the initial values (7.40), the state reachable tube $\vec{Z}^{[i]}(\vec{\mathbf{u}})$ can be written as an explicit function of the input sequence $\vec{\mathbf{u}}$ as $\vec{Z}^{[i]}(\vec{\mathbf{u}}) = \{\vec{\mathbf{M}}^{[i]}, \vec{\mathbf{G}}^{[i]}, \vec{\mathbf{c}}^{[i]}(\vec{\mathbf{u}}), \vec{\mathbf{S}}^{[i]}, \vec{\mathbf{A}}^{[i]}, \vec{\mathbf{b}}^{[i]}(\vec{\mathbf{u}})\}$, where

$$\vec{\mathbf{c}}^{[i]}(\vec{\mathbf{u}}) = \mathbf{Q}_N^{[i]} \mathbf{c}_z + \mathbf{p}_N^{[i]} + \vec{\mathbf{H}}^{[i]} \vec{\mathbf{u}}, \quad \vec{\mathbf{M}}^{[i]} = [\mathbf{Q}_N^{[i]} \mathbf{M}_z^{[i]} \vec{\mathbf{P}}_M^{[i]}], \quad \vec{\mathbf{G}}^{[i]} = [\mathbf{Q}_N^{[i]} \mathbf{G}_z^{[i]} \vec{\mathbf{P}}_G^{[i]}], \quad (7.41)$$

$$\vec{\mathbf{A}}^{[i]} = \begin{bmatrix} \begin{bmatrix} \mathbf{A}_z \\ \check{\mathbf{A}}_z^{[i]} \mathbf{G}_z^{[i]} \end{bmatrix} & \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix} & \cdots & \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix} \\ \check{\mathbf{A}}_z^{[i]} \begin{bmatrix} \check{\mathbf{A}}_z^{[i]} \\ \mathbf{0} \end{bmatrix} \mathbf{G}_z^{[i]} & \check{\mathbf{A}}_z^{[i]} \begin{bmatrix} \mathbf{0} \\ \mathbf{G}_A \end{bmatrix} & \cdots & \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix} \\ \check{\mathbf{A}}_z^{[i]} \begin{bmatrix} \check{\mathbf{A}}_z^{[i]} \\ \mathbf{0} \end{bmatrix}^2 \mathbf{G}_z^{[i]} & \check{\mathbf{A}}_z^{[i]} \begin{bmatrix} \check{\mathbf{A}}_z^{[i]} \\ \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{0} \\ \mathbf{G}_A \end{bmatrix} & \cdots & \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix} \\ \vdots & \vdots & \ddots & \vdots \\ \check{\mathbf{A}}_z^{[i]} \begin{bmatrix} \check{\mathbf{A}}_z^{[i]} \\ \mathbf{0} \end{bmatrix}^N \mathbf{G}_z^{[i]} & \check{\mathbf{A}}_z^{[i]} \begin{bmatrix} \check{\mathbf{A}}_z^{[i]} \\ \mathbf{0} \end{bmatrix}^{N-1} \begin{bmatrix} \mathbf{0} \\ \mathbf{G}_A \end{bmatrix} & \cdots & \check{\mathbf{A}}_z^{[i]} \begin{bmatrix} \mathbf{0} \\ \mathbf{G}_A \end{bmatrix} \\ \mathbf{0} & \mathbf{A}_A & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{A}_A \end{bmatrix}, \quad (7.42)$$

$$\vec{\mathbf{S}}^{[i]} = \begin{bmatrix} \begin{bmatrix} \mathbf{S}_z \\ \check{\mathbf{A}}_z^{[i]} \mathbf{M}_z^{[i]} \end{bmatrix} & \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix} & \cdots & \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix} \\ \check{\mathbf{A}}_z^{[i]} \begin{bmatrix} \check{\mathbf{A}}_z^{[i]} \\ \mathbf{0} \end{bmatrix} \mathbf{M}_z^{[i]} & \check{\mathbf{A}}_z^{[i]} \begin{bmatrix} \mathbf{0} \\ \mathbf{M}_A \end{bmatrix} & \cdots & \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix} \\ \check{\mathbf{A}}_z^{[i]} \begin{bmatrix} \check{\mathbf{A}}_z^{[i]} \\ \mathbf{0} \end{bmatrix}^2 \mathbf{M}_z^{[i]} & \check{\mathbf{A}}_z^{[i]} \begin{bmatrix} \check{\mathbf{A}}_z^{[i]} \\ \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{0} \\ \mathbf{M}_A \end{bmatrix} & \cdots & \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix} \\ \vdots & \vdots & \ddots & \vdots \\ \check{\mathbf{A}}_z^{[i]} \begin{bmatrix} \check{\mathbf{A}}_z^{[i]} \\ \mathbf{0} \end{bmatrix}^N \mathbf{M}_z^{[i]} & \check{\mathbf{A}}_z^{[i]} \begin{bmatrix} \check{\mathbf{A}}_z^{[i]} \\ \mathbf{0} \end{bmatrix}^{N-1} \begin{bmatrix} \mathbf{0} \\ \mathbf{M}_A \end{bmatrix} & \cdots & \check{\mathbf{A}}_z^{[i]} \begin{bmatrix} \mathbf{0} \\ \mathbf{M}_A \end{bmatrix} \\ \mathbf{0} & \mathbf{S}_A & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{S}_A \end{bmatrix} \quad (7.43)$$

$$\vec{\mathbf{b}}^{[i]} = \vec{\boldsymbol{\alpha}}^{[i]} + \vec{\boldsymbol{\Lambda}}^{[i]} \mathbf{c}_z + \vec{\boldsymbol{\Omega}}^{[i]} \vec{\mathbf{u}}, \quad (7.44)$$

with

$$\mathbf{Q}_N^{[i]} = \begin{bmatrix} \begin{bmatrix} \tilde{\mathbf{A}}_z^{[i]} \\ \mathbf{0} \end{bmatrix}^0 \\ \begin{bmatrix} \tilde{\mathbf{A}}_z^{[i]} \\ \mathbf{0} \end{bmatrix}^1 \\ \begin{bmatrix} \tilde{\mathbf{A}}_z^{[i]} \\ \mathbf{0} \end{bmatrix}^2 \\ \vdots \\ \begin{bmatrix} \tilde{\mathbf{A}}_z^{[i]} \\ \mathbf{0} \end{bmatrix}^N \end{bmatrix}, \quad \mathbf{p}_N^{[i]} = \begin{bmatrix} \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{c}_A \end{bmatrix} \\ \begin{bmatrix} \tilde{\mathbf{A}}_z^{[i]} \\ \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{0} \\ \mathbf{c}_A \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{c}_A \end{bmatrix} \\ \vdots \\ \sum_{m=1}^N \begin{bmatrix} \tilde{\mathbf{A}}_z^{[i]} \\ \mathbf{0} \end{bmatrix}^{m-1} \begin{bmatrix} \mathbf{0} \\ \mathbf{c}_A \end{bmatrix} \end{bmatrix}, \quad (7.45)$$

$$\vec{\mathbf{H}}^{[i]} = \begin{bmatrix} \mathbf{0} \\ \mathbf{H}_1^{[i]} \\ \mathbf{H}_2^{[i]} \\ \vdots \\ \mathbf{H}_N^{[i]} \end{bmatrix}, \quad \mathbf{H}_h^{[i]} = \left[\cdots \underbrace{\begin{bmatrix} \tilde{\mathbf{A}}_z^{[i]} \\ \mathbf{0} \end{bmatrix}^{h-m} \begin{bmatrix} \tilde{\mathbf{B}}^{[i]} \\ \mathbf{0} \end{bmatrix}}_{m=1,2,\dots,h} \cdots \underbrace{\begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}}_{N-h+1 \text{ terms}} \cdots \right], \quad (7.46)$$

$$\vec{\mathbf{P}}_G^{[i]} = \begin{bmatrix} \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{G}_A \end{bmatrix} \\ \begin{bmatrix} \tilde{\mathbf{A}}_z^{[i]} \\ \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{0} \\ \mathbf{G}_A \end{bmatrix} \\ \vdots \\ \begin{bmatrix} \tilde{\mathbf{A}}_z^{[i]} \\ \mathbf{0} \end{bmatrix}^{N-1} \begin{bmatrix} \mathbf{0} \\ \mathbf{G}_A \end{bmatrix} \end{bmatrix}, \quad (7.47)$$

$$\vec{\mathbf{P}}_M^{[i]} = \begin{bmatrix} \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{M}_A \end{bmatrix} \\ \begin{bmatrix} \tilde{\mathbf{A}}_z^{[i]} \\ \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{0} \\ \mathbf{M}_A \end{bmatrix} \\ \vdots \\ \begin{bmatrix} \tilde{\mathbf{A}}_z^{[i]} \\ \mathbf{0} \end{bmatrix}^{N-1} \begin{bmatrix} \mathbf{0} \\ \mathbf{M}_A \end{bmatrix} \end{bmatrix}, \quad (7.48)$$

$$\vec{\boldsymbol{\alpha}}^{[i]} = \vec{\boldsymbol{\beta}} + \vec{\Upsilon}^{[i]} \mathbf{p}_N^{[i]}, \quad \vec{\boldsymbol{\lambda}}^{[i]} = \vec{\Upsilon}^{[i]} \mathbf{Q}_N^{[i]}, \quad \vec{\boldsymbol{\Omega}}^{[i]} = \vec{\boldsymbol{\Gamma}}^{[i]} + \vec{\Upsilon}^{[i]} \vec{\mathbf{H}}^{[i]}, \quad (7.49)$$

$$\vec{\beta} = \begin{bmatrix} \mathbf{b}_z \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \\ \mathbf{b}_A \\ \mathbf{b}_A \\ \vdots \\ \mathbf{b}_A \end{bmatrix}, \quad \vec{\Upsilon}^{[i]} = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ -\check{\mathbf{A}}_z^{[i]} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & -\check{\mathbf{A}}_z^{[i]} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -\check{\mathbf{A}}_z^{[i]} & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & -\check{\mathbf{A}}_z^{[i]} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \end{bmatrix}, \quad (7.50)$$

$$\vec{\Gamma}^{[i]} = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ -\check{\mathbf{B}}^{[i]} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & -\check{\mathbf{B}}^{[i]} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -\check{\mathbf{B}}^{[i]} & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & -\check{\mathbf{B}}^{[i]} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \end{bmatrix}, \quad (7.51)$$

$$\mathbf{c}_A = (\mathbf{c}_R, \mathbf{c}_w), \quad \mathbf{G}_A = \begin{bmatrix} \mathbf{0} \\ \mathbf{G}_w \end{bmatrix}, \quad \mathbf{M}_A = \begin{bmatrix} \mathbf{M}_R & \mathbf{0} \\ \mathbf{0} & \mathbf{M}_w \end{bmatrix}, \quad \mathbf{S}_A = [\mathbf{0} \ \mathbf{S}_w] \quad \mathbf{A}_A = \mathbf{A}_w, \quad \mathbf{b}_A = \mathbf{b}_w.$$

7.6.2 Output reachable tube

Since $\vec{Z}^{[i]}(\vec{\mathbf{u}})$ is a mixed zonotope, the output reachable tube $\vec{Y}^{[i]}(\vec{\mathbf{u}})$ is then a mixed zonotope obtained in accordance with (7.39c) as

$$\vec{Y}^{[i]}(\vec{\mathbf{u}}) = \vec{\mathbf{F}}^{[i]} \vec{Z}^{[i]}(\vec{\mathbf{u}}) \oplus \vec{\mathbf{D}}^{[i]} \vec{\mathbf{u}} \oplus \vec{\mathbf{D}}_v^{[i]} \vec{V}, \quad (7.52)$$

where

$$\vec{\mathbf{F}}^{[i]} = \begin{bmatrix} \mathbf{F}^{[i]} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{F}^{[i]} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{F}^{[i]} \end{bmatrix}, \quad \vec{\mathbf{D}}^{[i]} = \begin{bmatrix} \mathbf{D}^{[i]} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{D}^{[i]} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{D}^{[i]} \end{bmatrix}, \quad \vec{\mathbf{D}}_v^{[i]} = \begin{bmatrix} \mathbf{D}_v^{[i]} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_v^{[i]} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{D}_v^{[i]} \end{bmatrix}.$$

Using properties (7.24) and (7.25), and letting $V = \{\mathbf{M}_v, \mathbf{G}_v, \mathbf{c}_v, \mathbf{S}_v, \mathbf{A}_v, \mathbf{b}_v\}$, the output reachable tube is given by the MG-rep $\vec{Y}^{[i]}(\vec{\mathbf{u}}) = \{\vec{\mathbf{M}}^{Y[i]}, \vec{\mathbf{G}}^{Y[i]}, \vec{\mathbf{c}}^{Y[i]}(\vec{\mathbf{u}}), \vec{\mathbf{S}}^{Y[i]}, \vec{\mathbf{A}}^{Y[i]}, \vec{\mathbf{b}}^{Y[i]}(\vec{\mathbf{u}})\}$,

in which

$$\vec{\mathbf{c}}^{Y[i]}(\vec{\mathbf{u}}) = \vec{\mathbf{F}}^{[i]} \vec{\mathbf{c}}^{[i]}(\vec{\mathbf{u}}) + \vec{\mathbf{D}}^{[i]} \vec{\mathbf{u}} + \vec{\mathbf{D}}_v^{[i]} \vec{\mathbf{c}}_v, \quad (7.53)$$

$$\vec{\mathbf{M}}^{Y[i]} = [\vec{\mathbf{F}}^{[i]} \vec{\mathbf{M}}^{[i]} \quad \vec{\mathbf{D}}_v^{[i]} \vec{\mathbf{M}}_v^{[i]}], \quad \vec{\mathbf{G}}^{Y[i]} = [\vec{\mathbf{F}}^{[i]} \vec{\mathbf{G}}^{[i]} \quad \vec{\mathbf{D}}_v^{[i]} \vec{\mathbf{G}}_v^{[i]}], \quad (7.54)$$

$$\vec{\mathbf{S}}^{Y[i]} = \begin{bmatrix} \vec{\mathbf{S}}^{[i]} & \mathbf{0} \\ \mathbf{0} & \vec{\mathbf{S}}_v^{[i]} \end{bmatrix}, \quad \vec{\mathbf{A}}^{Y[i]} = \begin{bmatrix} \vec{\mathbf{A}}^{[i]} & \mathbf{0} \\ \mathbf{0} & \vec{\mathbf{A}}_v^{[i]} \end{bmatrix}, \quad \vec{\mathbf{b}}^{Y[i]}(\vec{\mathbf{u}}) = \begin{bmatrix} \vec{\mathbf{b}}^{[i]}(\vec{\mathbf{u}}) \\ \vec{\mathbf{b}}_v^{[i]} \end{bmatrix}. \quad (7.55)$$

7.6.3 Separation of output reachable tubes

Consider an input sequence $\vec{\mathbf{u}} \in \vec{U}$ to be injected into the set of models (7.39), and let $\vec{\mathbf{y}}^{[i]} \triangleq (\mathbf{y}_0^{[i]}, \mathbf{y}_1^{[i]}, \dots, \mathbf{y}_N^{[i]})$ denote the observed output sequence of model $i \in \mathcal{M}$. We are interested in the design of an input sequence such that the relation $\vec{\mathbf{y}}^{[i]} \in \vec{Y}^{[i]}(\vec{\mathbf{u}})$ is valid for only one $i \in \mathcal{M}$.

Definition 7.3. An input sequence $\vec{\mathbf{u}}$ is said to be a *separating input* on $k \in [0, N]$ if, for every $i, j \in \mathcal{M}$, $i \neq j$,

$$\vec{Y}^{[i]}(\vec{\mathbf{u}}) \cap \vec{Y}^{[j]}(\vec{\mathbf{u}}) = \emptyset. \quad (7.56)$$

Clearly, if (7.56) holds for all $i, j \in \mathcal{M}$, $i \neq j$, then $\vec{\mathbf{y}}^{[i]} \in \vec{Y}^{[i]}(\vec{\mathbf{u}})$ must hold only for one $i \in \mathcal{M}$. In the case that this is not valid for any $i \in \mathcal{M}$, one concludes that the real dynamics does not belong to the set of models (7.39). The following theorem is based on the computation of $\vec{Y}^{[i]}(\vec{\mathbf{u}})$ expressed by (7.52) and the results in Raimondo et al. (2016).

Theorem 7.2. An input $\vec{\mathbf{u}} \in \vec{U}$ is a separating input iff

$$\begin{bmatrix} \vec{\mathbf{N}}(i, j) \\ \vec{\mathbf{\Omega}}(i, j) \end{bmatrix} \vec{\mathbf{u}} \notin \vec{\mathcal{Y}}(i, j) \triangleq \left\{ \begin{bmatrix} \vec{\mathbf{M}}^Y(i, j) \\ \vec{\mathbf{S}}^Y(i, j) \end{bmatrix}, \begin{bmatrix} \vec{\mathbf{G}}^Y(i, j) \\ \vec{\mathbf{A}}^Y(i, j) \end{bmatrix}, \begin{bmatrix} \vec{\mathbf{c}}^Y(i, j) \\ -\vec{\mathbf{b}}^Y(i, j) \end{bmatrix}, -, -, - \right\} \quad (7.57)$$

$\forall i, j \in \mathcal{M}$, $i \neq j$, where $\mathbf{N}(i, j) = \vec{\mathbf{F}}^{[j]} \vec{\mathbf{H}}^{[j]} + \vec{\mathbf{D}}^{[j]} - (\vec{\mathbf{F}}^{[i]} \vec{\mathbf{H}}^{[i]} + \vec{\mathbf{D}}^{[i]})$, $\mathbf{\Omega}(i, j) = [(\mathbf{\Omega}_N^{[i]})^T \mathbf{0} \quad (\mathbf{\Omega}_N^{[j]})^T \mathbf{0}]^T$, and

$$\begin{aligned} \vec{\mathbf{M}}^Y(i, j) &\triangleq [\vec{\mathbf{M}}^{Y[i]} \quad -\vec{\mathbf{M}}^{Y[j]}], & \vec{\mathbf{G}}^Y(i, j) &\triangleq [\vec{\mathbf{G}}^{Y[i]} \quad -\vec{\mathbf{G}}^{Y[j]}], & \vec{\mathbf{c}}^Y(i, j) &= \vec{\mathbf{c}}^{Y[i]}(\vec{\mathbf{0}}) - \vec{\mathbf{c}}^{Y[j]}(\vec{\mathbf{0}}), \\ \vec{\mathbf{S}}^Y(i, j) &\triangleq \begin{bmatrix} \vec{\mathbf{S}}^{Y[i]} & \mathbf{0} \\ \mathbf{0} & \vec{\mathbf{S}}^{Y[j]} \end{bmatrix}, & \vec{\mathbf{A}}^Y(i, j) &\triangleq \begin{bmatrix} \vec{\mathbf{A}}^{Y[i]} & \mathbf{0} \\ \mathbf{0} & \vec{\mathbf{A}}^{Y[j]} \end{bmatrix}, & \vec{\mathbf{b}}^Y(i, j) &= \begin{bmatrix} \vec{\mathbf{b}}^{Y[i]}(\vec{\mathbf{0}}) \\ \vec{\mathbf{b}}^{Y[j]}(\vec{\mathbf{0}}) \end{bmatrix}, \end{aligned}$$

where $\vec{\mathbf{0}}$ denotes the zero input sequence.

Proof. The relations below follow from (7.41), (7.44), (7.53), and (7.55),

$$\vec{\mathbf{c}}^{Y[i]}(\vec{\mathbf{u}}) = \vec{\mathbf{c}}^{Y[i]}(\mathbf{0}) + \vec{\mathbf{F}}^{[i]} \vec{\mathbf{H}}^{[i]} \vec{\mathbf{u}} + \vec{\mathbf{D}}^{[i]} \vec{\mathbf{u}} = \vec{\mathbf{c}}^{Y[i]}(\mathbf{0}) + (\vec{\mathbf{F}}^{[i]} \vec{\mathbf{H}}^{[i]} + \vec{\mathbf{D}}^{[i]}) \vec{\mathbf{u}}, \quad (7.58)$$

$$\vec{\mathbf{b}}^{Y[i]}(\vec{\mathbf{u}}) = \vec{\mathbf{b}}^{Y[i]}(\mathbf{0}) + \begin{bmatrix} \vec{\mathbf{\Omega}}^{[i]} \\ \mathbf{0} \end{bmatrix} \vec{\mathbf{u}}. \quad (7.59)$$

From (7.26) with $\mathbf{R} = \mathbf{I}$, (7.56) is true iff $\exists \boldsymbol{\xi} \in B_\infty, \boldsymbol{\delta} \in \mathbb{R}$ such that

$$\begin{bmatrix} \vec{\mathbf{S}}^{Y[i]} & \mathbf{0} \\ \mathbf{0} & \vec{\mathbf{S}}^{Y[j]} \\ \vec{\mathbf{M}}^{Y[i]} & -\vec{\mathbf{M}}^{Y[j]} \end{bmatrix} \boldsymbol{\delta} + \begin{bmatrix} \vec{\mathbf{A}}^{Y[i]} & \mathbf{0} \\ \mathbf{0} & \vec{\mathbf{A}}^{Y[j]} \\ \vec{\mathbf{G}}^{Y[i]} & -\vec{\mathbf{G}}^{Y[j]} \end{bmatrix} \boldsymbol{\xi} = \begin{bmatrix} \vec{\mathbf{b}}^{Y[i]}(\vec{\mathbf{u}}) \\ \vec{\mathbf{b}}^{Y[j]}(\vec{\mathbf{u}}) \\ \vec{\mathbf{c}}^{Y[j]}(\vec{\mathbf{u}}) - \vec{\mathbf{c}}^{Y[i]}(\vec{\mathbf{u}}) \end{bmatrix}.$$

According to (7.58)–(7.59), one has $\vec{\mathbf{c}}^{Y[i]}(\vec{\mathbf{u}}) = \vec{\mathbf{c}}^{Y[i]}(\mathbf{0}) + \vec{\mathbf{F}}^{[i]} \vec{\mathbf{H}}^{[i]} \vec{\mathbf{u}} + \vec{\mathbf{D}}^{[i]} \vec{\mathbf{u}} = \vec{\mathbf{c}}^{Y[i]}(\mathbf{0}) + (\vec{\mathbf{F}}^{[i]} \vec{\mathbf{H}}^{[i]} + \vec{\mathbf{D}}^{[i]}) \vec{\mathbf{u}} = -\vec{\mathbf{c}}^Y(i, j) + \vec{\mathbf{N}}(i, j) \vec{\mathbf{u}}$, $[(\mathbf{b}_N^{Y[i]}(\vec{\mathbf{u}}))^T (\mathbf{b}_N^{Y[j]}(\vec{\mathbf{u}}))^T]^T = \vec{\mathbf{b}}_N^Y(i, j) + \vec{\boldsymbol{\Omega}}(i, j) \vec{\mathbf{u}}$, with $\vec{\mathbf{c}}^Y(i, j)$, $\vec{\mathbf{b}}^Y(i, j)$, $\vec{\mathbf{N}}(i, j)$, and $\vec{\boldsymbol{\Omega}}(i, j)$ defined as in the statement of the theorem. Then, (7.56) holds iff $\exists \boldsymbol{\xi} \in B_\infty$ and $\boldsymbol{\delta} \in \mathbb{R}$ such that $\vec{\mathbf{M}}^Y(i, j) \boldsymbol{\delta} + \vec{\mathbf{G}}^Y(i, j) \boldsymbol{\xi} = -\vec{\mathbf{c}}^Y(i, j) + \vec{\mathbf{N}}(i, j) \vec{\mathbf{u}}$, and $\vec{\mathbf{S}}^Y(i, j) \boldsymbol{\delta} + \vec{\mathbf{A}}^Y(i, j) \boldsymbol{\xi} = \vec{\mathbf{b}}^Y(i, j) + \vec{\boldsymbol{\Omega}}(i, j) \vec{\mathbf{u}}$. This is equivalent to

$$\exists (\boldsymbol{\delta}, \boldsymbol{\xi}) \in \mathbb{R} \times B_\infty : \begin{bmatrix} \vec{\mathbf{M}}^Y(i, j) \\ \vec{\mathbf{S}}^Y(i, j) \end{bmatrix} \boldsymbol{\delta} + \begin{bmatrix} \vec{\mathbf{G}}^Y(i, j) \\ \vec{\mathbf{A}}^Y(i, j) \end{bmatrix} \boldsymbol{\xi} + \begin{bmatrix} \vec{\mathbf{c}}^Y(i, j) \\ -\vec{\mathbf{b}}^Y(i, j) \end{bmatrix} = \begin{bmatrix} \vec{\mathbf{N}}(i, j) \\ \vec{\boldsymbol{\Omega}}(i, j) \end{bmatrix} \vec{\mathbf{u}},$$

which in turn holds iff (7.57) is satisfied, with $\mathbf{M}_N^Y(i, j)$, $\mathbf{G}_N^Y(i, j)$, $\mathbf{S}_N^Y(i, j)$, and $\mathbf{A}_N^Y(i, j)$ defined as in the statement of the theorem. \blacksquare

For simplicity, let $(\cdot)^{[q]} \triangleq (\cdot)(i, j)$, with $q \in \{1, 2, \dots, n_q\}$, where n_q denotes the number of possible combinations of $i, j \in \mathcal{M}$, $i < j$.

Lemma 7.4. Let

$$\mathbf{N}^{\dagger[q]} \triangleq \begin{bmatrix} \vec{\mathbf{N}}^{[q]} \\ \vec{\boldsymbol{\Omega}}^{[q]} \end{bmatrix}, \mathbf{G}^{\dagger[q]} \triangleq \begin{bmatrix} \vec{\mathbf{G}}^{Y[q]} \\ \vec{\mathbf{A}}^{Y[q]} \end{bmatrix}, \mathbf{M}^{\dagger[q]} \triangleq \begin{bmatrix} \vec{\mathbf{M}}^{Y[q]} \\ \vec{\mathbf{S}}^{Y[q]} \end{bmatrix}, \mathbf{c}^{\dagger[q]} \triangleq \begin{bmatrix} \vec{\mathbf{c}}^{Y[q]} \\ -\vec{\mathbf{b}}^{Y[q]} \end{bmatrix}.$$

Moreover, let $\mathbf{M}^{\dagger[q]}$ be full column rank. Then, there exist matrices $\mathbf{M}^{-[q]}$, $\mathbf{M}^{+[q]}$, $\mathbf{G}^{-[q]}$, $\mathbf{G}^{+[q]}$, $\mathbf{N}^{-[q]}$, $\mathbf{N}^{+[q]}$, and vectors $\mathbf{c}^{-[q]}$, $\mathbf{c}^{+[q]}$, such that

$$\mathbf{N}^{\dagger[q]} \vec{\mathbf{u}} \notin \vec{\mathcal{Y}}^{[q]} = \{\mathbf{M}^{\dagger[q]}, \mathbf{G}^{\dagger[q]}, \mathbf{c}^{\dagger[q]}, _, _, _ \} \iff \mathring{\mathbf{N}}^{[q]} \vec{\mathbf{u}} \notin \mathring{\mathcal{Y}}^{[q]} \triangleq \{\mathring{\mathbf{G}}^{[q]}, \mathring{\mathbf{c}}^{[q]}\},$$

with $\mathring{\mathbf{G}}^{[q]} \triangleq \mathbf{G}^{-[q]} - \mathbf{M}^{-[q]}(\mathbf{M}^{+[q]})^{-1} \mathbf{G}^{+[q]}$, $\mathring{\mathbf{c}}^{[q]} \triangleq \mathbf{c}^{-[q]} - \mathbf{M}^{-[q]}(\mathbf{M}^{+[q]})^{-1} \mathbf{c}^{+[q]}$, $\mathring{\mathbf{N}}^{[q]} \triangleq \mathbf{N}^{-[q]} - \mathbf{M}^{-[q]}(\mathbf{M}^{+[q]})^{-1} \mathbf{N}^{+[q]}$.

Proof. Recall that $\mathbf{N}^{\dagger[q]} \vec{\mathbf{u}} \notin \vec{\mathcal{Y}}^{[q]} \iff \exists (\boldsymbol{\delta}, \boldsymbol{\xi}) \in \mathbb{R}^{n_\delta} \times B_\infty^{n_g} : \mathbf{M}^{\dagger[q]} \boldsymbol{\delta} + \mathbf{G}^{\dagger[q]} \boldsymbol{\xi} + \mathbf{c}^{\dagger[q]} = \mathbf{N}^{\dagger[q]} \vec{\mathbf{u}}$. Since $\mathbf{M}^{\dagger[q]}$ is full column rank, then by rearranging the rows of $\mathbf{M}^{\dagger[q]} \boldsymbol{\delta} + \mathbf{G}^{\dagger[q]} \boldsymbol{\xi} + \mathbf{c}^{\dagger[q]} = \mathbf{N}^{\dagger[q]} \vec{\mathbf{u}}$, there exists an invertible matrix $\mathbf{M}^{+[q]}$ satisfying

$$\begin{bmatrix} \mathbf{M}^{+[q]} \\ \mathbf{M}^{-[q]} \end{bmatrix} \boldsymbol{\delta} + \begin{bmatrix} \mathbf{G}^{+[q]} \\ \mathbf{G}^{-[q]} \end{bmatrix} \boldsymbol{\xi} + \begin{bmatrix} \mathbf{c}^{+[q]} \\ \mathbf{c}^{-[q]} \end{bmatrix} = \begin{bmatrix} \mathbf{N}^{+[q]} \\ \mathbf{N}^{-[q]} \end{bmatrix} \vec{\mathbf{u}}. \quad (7.60)$$

Therefore, $\mathbf{M}^{+[q]} \boldsymbol{\delta} + \mathbf{G}^{+[q]} \boldsymbol{\xi} + \mathbf{c}^{+[q]} = \mathbf{N}^{+[q]} \vec{\mathbf{u}}$ holds, implying that $\boldsymbol{\delta} = (\mathbf{M}^{+[q]})^{-1} (-\mathbf{G}^{+[q]} \boldsymbol{\xi} -$

$\mathbf{c}^{+[q]} + \mathbf{N}^{+[q]}\vec{\mathbf{u}}$). Substituting the latter in the lower part of (7.60) leads to

$$(\mathbf{G}^{-[q]} - \mathbf{M}^{-[q]}(\mathbf{M}^{+[q]})^{-1}\mathbf{G}^{+[q]})\boldsymbol{\xi} + (\mathbf{I} - \mathbf{M}^{-[q]}(\mathbf{M}^{+[q]})^{-1})\mathbf{c}^{+[q]} = (\mathbf{N}^{-[q]} - \mathbf{M}^{-[q]}(\mathbf{M}^{+[q]})^{-1}\mathbf{N}^{+[q]})\vec{\mathbf{u}},$$

which proves the lemma. \blacksquare

Let $\mathring{\mathbf{N}}^{[q]}$ and $\mathring{\mathcal{Y}}^{[q]} = \{\mathring{\mathbf{G}}^{[q]}, \mathring{\mathbf{c}}^{[q]}\}$ be defined as in Lemma 7.4, for each $q \in \{1, 2, \dots, n_q\}$. As in the constrained zonotope method, $\mathring{\mathcal{Y}}^{[q]}$ is a zonotope. Then, the relation $\mathring{\mathbf{N}}^{[q]}\vec{\mathbf{u}} \notin \mathring{\mathcal{Y}}^{[q]}$ can be verified by solving an LP. In this sense, the following lemma provides an effective way to verify if a given input sequence is a separating input according to Theorem 7.2, consequently satisfying (7.56).

Lemma 7.5. Let $\mathring{\mathcal{Y}}^{[q]} = \{\mathring{\mathbf{G}}^{[q]}, \mathring{\mathbf{c}}^{[q]}\}$. For each $\vec{\mathbf{u}} \in \vec{U}$ and $q \in \{1, 2, \dots, n_q\}$, define $\hat{\delta}^{[q]}(\vec{\mathbf{u}}) = \min_{\delta^{[q]}, \boldsymbol{\xi}^{[q]}} \delta^{[q]}$, subject to

$$\mathring{\mathbf{N}}^{[q]}\vec{\mathbf{u}} = \mathring{\mathbf{G}}^{[q]}\boldsymbol{\xi}^{[q]} + \mathring{\mathbf{c}}^{[q]}, \quad \|\boldsymbol{\xi}^{[q]}\|_{\infty} \leq 1 + \delta^{[q]}.$$

Then $\mathring{\mathbf{N}}^{[q]}\vec{\mathbf{u}} \notin \mathring{\mathcal{Y}}^{[q]} \iff \hat{\delta}^{[q]}(\vec{\mathbf{u}}) > 0$.

Proof. See Lemma 4 in Scott et al. (2014). \blacksquare

For the AFD of the n_m models in (7.39) of the descriptor system, we consider the design of a separating input of minimum length according to the optimization problem

$$\min_{\vec{\mathbf{u}} \in \vec{U}} \{J(\vec{\mathbf{u}}) : \mathring{\mathbf{N}}^{[q]}\vec{\mathbf{u}} \notin \mathring{\mathcal{Y}}^{[q]}, \forall q = 1, 2, \dots, n_q\}, \quad (7.61)$$

with $J(\vec{\mathbf{u}})$ chosen to minimize any harmful effects caused by injecting $\vec{\mathbf{u}}$ into (7.39). For simplicity we may choose $J(\vec{\mathbf{u}}) = \sum_{j=0}^N \mathbf{u}_j^T \mathbf{R} \mathbf{u}_j$, where \mathbf{R} is a weighting matrix. As in Scott et al. (2014), this is a bilevel optimization problem and can be rewritten as a mixed-integer quadratic program by defining a minimum separation threshold $\varepsilon > 0$ such that $\varepsilon \leq \hat{\delta}^{[q]}(\vec{\mathbf{u}})$, for all $q = 1, 2, \dots, n_q$, similar to the MIQP presented in Chapter 6.

7.7 Numerical examples

7.7.1 State estimation

This section first evaluates the accuracy of the state estimation methods proposed in Sections 7.2 and 7.5 for descriptor systems using CZs and MZs, respectively (denoted by CZ and MZ). Consider system (7.1) with matrices $\mathbf{E} = \text{diag}(1, 1, 0)$, $\mathbf{B}_w = \text{diag}(0.1, 1.5, 0.6)$, $\mathbf{D}_v = \text{diag}(0.5, 1.5)$,

$$\mathbf{A} = \begin{bmatrix} 0.5 & 0 & 0 \\ 0.8 & 0.95 & 0 \\ -1 & 0.5 & 1 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} 1 & 0 & 1 \\ 1 & -1 & 0 \end{bmatrix},$$

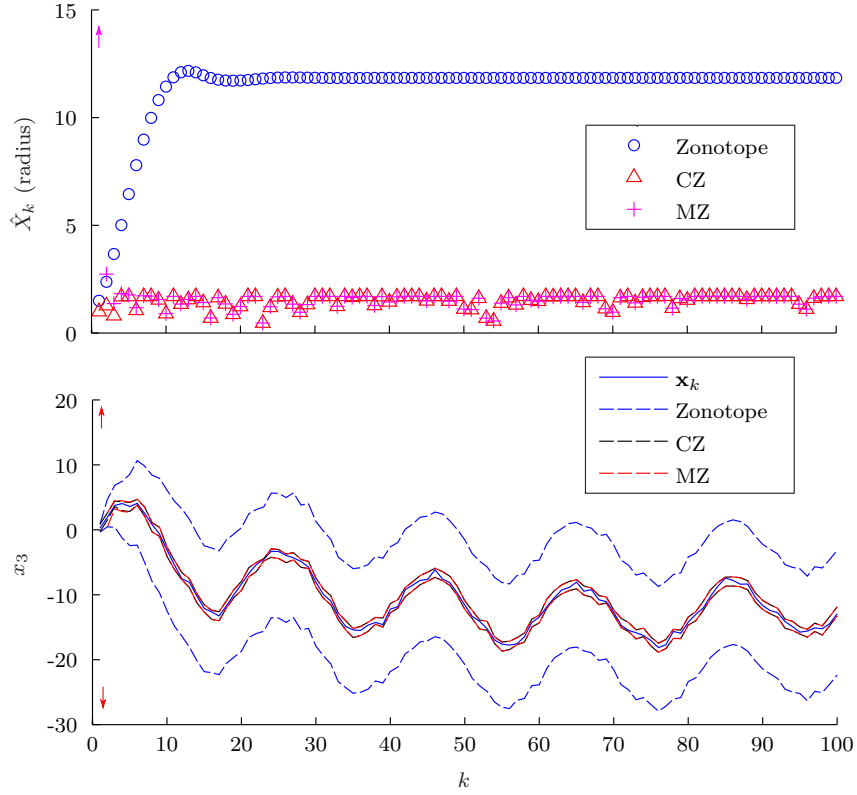


Figure 7.5: The radii of the enclosures \hat{X}_k obtained using zonotope method proposed in Wang et al. (2018a), the constrained zonotope method proposed in Section 7.2, and the mixed zonotope method proposed in Section 7.5 (top), as well as the projections of \hat{X}_k onto x_3 (bottom).

and $\mathbf{D} = \mathbf{0}$. The initial state \mathbf{x}_0 is bounded by the zonotope

$$X_0 = \{\text{diag}(0.1, 1.5, 0.6), [0.5 \ 0.5 \ 0.25]^T\}, \quad (7.62)$$

and the uncertainties are random uniform noises bounded by $\|\mathbf{w}_k\|_\infty \leq 1$, $\|\mathbf{v}_k\|_\infty \leq 1$. The CZ in Assumption 1 is $X_a = \{50 \cdot \mathbf{I}_3, \mathbf{0}\}$. On the other hand, for MZs the considered initial set is the entire state space \mathbb{R}^3 . The simulation is conducted for $\mathbf{x}_0 = [0.5 \ 0.5 \ 0.25]^T$, and the complexity of the CZs and MZs is limited to 30 (bounded) generators and 5 constraints using the constraint elimination algorithm described by Method 3.4 and the generator reduction described by Method 3.5. For MZs, unbounded generators are eliminated prior to eliminating constraints following the scheme proposed in Section 7.4.1.

Figure 7.5 shows the radii of the enclosures \hat{X}_k for $k \in [0, 100]$ obtained using the state estimation algorithms proposed in Sections 7.2 and 7.5. Results obtained using the zonotope method in Wang et al. (2018a) are presented for comparison². The complexity of the zonotopes is limited to 30 generators using Method 3.2. Figure 7.5 shows also the projections of \hat{X}_k onto x_3 . As it can be noticed, both CZ and MZ provide substantially

²Specifically, this is the set-membership approach proposed in Wang et al. (2018a) with Kalman correction matrix.

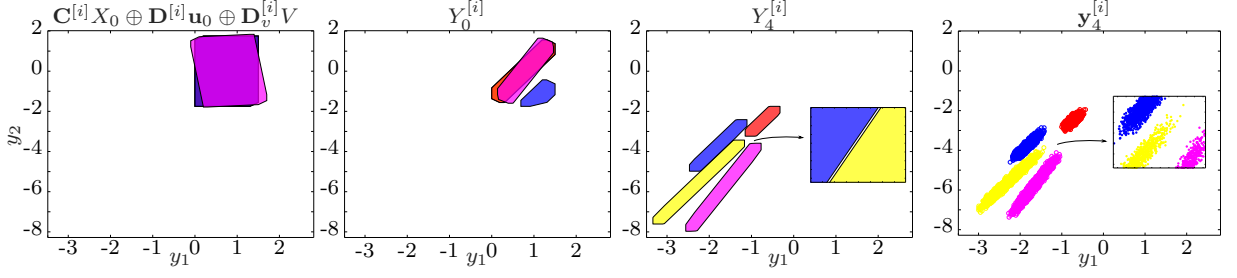


Figure 7.6: The sets $\mathbf{C}^{[i]}X_0 \oplus \mathbf{D}^{[i]}\mathbf{u}_0 \oplus \mathbf{D}_v^{[i]}V$, the output reachable sets $Y_0^{[i]}$, $Y_4^{[i]}$, and 2000 samples for $\mathbf{y}_4^{[i]}$, with $i = 1$ (yellow), $i = 2$ (red), $i = 3$ (blue), and $i = 4$ (magenta), obtained by the injection of the input sequence $\vec{\mathbf{u}}$.

sharper bounds in comparison to zonotopes. This is possible since the enclosures proposed in Lemmas 7.1 and 7.3 take into account the static constraints explicitly, while zonotopes provide only a conservative bound of the corresponding feasible region. In addition, as it can be noticed, MZ is able to provide enclosures that are as accurate as the ones obtained using CZ, but without requiring the knowledge of bounded initial and admissible sets. As a side note, the enclosure \hat{X}_k provided by MZ at $k = 1$ is unbounded (denoted by the arrows towards minus and plus infinity), but this enclosure becomes bounded at $k = 2$ since more measurements are added to the computed set.

7.7.2 Fault diagnosis using constrained zonotopes

We now evaluate the effectiveness of the AFD method proposed in Section 7.3. Consider the set of models (7.2) with model $i = 1$ described in the previous example, and

$$\mathbf{A}^{[2]} = \begin{bmatrix} 0.5 & 0 & 0 \\ 0.8 & 0.6 & 0 \\ -1 & 0.5 & 1 \end{bmatrix}, \quad \mathbf{B}^{[3]} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ -1 & 0 \end{bmatrix}, \quad \mathbf{C}^{[4]} = \begin{bmatrix} 1 & 0.1 & 1 \\ 1 & -1 & 0.1 \end{bmatrix},$$

$\mathbf{E}^{[i]} = \mathbf{E}^{[1]}$, $\mathbf{B}^{[2]} = \mathbf{B}^{[4]} = \mathbf{B}^{[1]}$, $\mathbf{C}^{[2]} = \mathbf{C}^{[3]} = \mathbf{C}^{[1]}$, $\mathbf{B}_w^{[i]} = \mathbf{B}_w^{[1]}$, and $\mathbf{D}_v^{[i]} = \mathbf{D}_v^{[1]}$, $\mathbf{D}^{[i]} = \mathbf{D}^{[1]}$, $i \in \{2, 3, 4\}$. The initial state $\mathbf{x}_0^{[i]}$ is bounded by (7.62), the uncertainties are bounded by $\|\mathbf{w}_k\|_\infty \leq 0.1$, $\|\mathbf{v}_k\|_\infty \leq 0.1$, and the input is limited by $\|\mathbf{u}_k\|_\infty \leq 1$. Let $X_a = \{50 \cdot \mathbf{I}_3, \mathbf{0}\}$ and $\varepsilon = 0.01$. All the models $i \in \mathcal{M}$ are considered to be faulty and must be separated. The number of generators of $\mathcal{Y}(i, j)$ was limited to two times its dimension using Method 3.2. The minimum length optimal input sequence that solves (7.22) was obtained using CPLEX 12.8 and MATLAB 9.1, with $J(\vec{\mathbf{u}}) = \vec{\mathbf{u}}^T \vec{\mathbf{u}}$, and is

$$\vec{\mathbf{u}} = \left(\begin{bmatrix} 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 0.73 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 0.92 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} -0.45 \\ 0 \end{bmatrix} \right).$$

Figure 7.6 shows the output reachable sets $Y_0^{[i]}$ and $Y_4^{[i]}$ for $i = 1, 2, 3, 4$, resulting from the injection of the designed $\vec{\mathbf{u}}$. We also show the sets defined by $\mathbf{C}^{[i]}X_0 \oplus \mathbf{D}^{[i]}\mathbf{u}_0 \oplus \mathbf{D}_v^{[i]}V$,

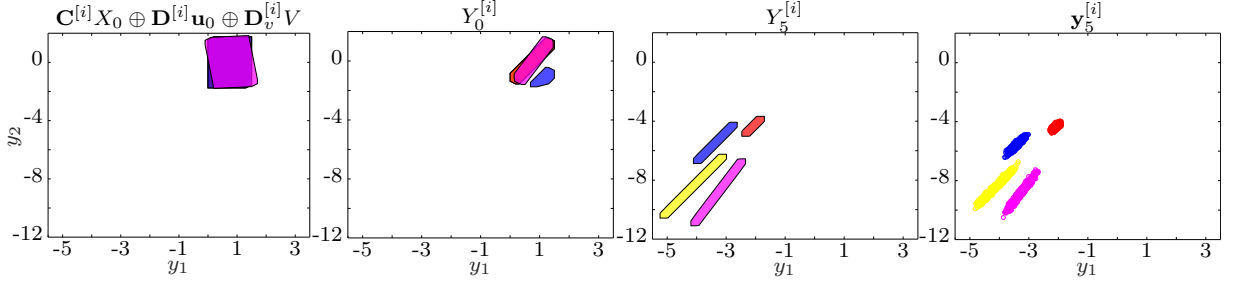


Figure 7.7: The sets $\mathbf{C}^{[i]}X_0 \oplus \mathbf{D}^{[i]}\mathbf{u}_0 \oplus \mathbf{D}_v^{[i]}V$, the output reachable sets $Y_0^{[i]}$, $Y_5^{[i]}$, and 1000 samples for $\mathbf{y}_5^{[i]}$, with $i = 1$ (yellow), $i = 2$ (red), $i = 3$ (blue), and $i = 4$ (magenta), obtained by the injection of the input sequence $\vec{\mathbf{u}}$.

which do not take into account the equality constraint (7.13b). Note that these sets are completely overlapped for $i \in \{1, 2, 3, 4\}$, and $Y_0^{[i]}$ are completely overlapped for $i \in \{1, 2, 4\}$. On the other hand, $Y_4^{[i]}$ are disjoint for every $i \in \mathcal{M}$, showing that the injection of $\vec{\mathbf{u}}$ guarantees fault diagnosis at $k = 4$. Figure 7.6 also shows clouds containing 2000 samples of the output $\mathbf{y}_4^{[i]}$ for each model i , demonstrating that the outputs of the models (7.2) are in fact separated.

7.7.3 Fault diagnosis using mixed zonotopes

We now evaluate the effectiveness of the AFD method based on mixed zonotopes proposed in Section 7.6. We consider the same set of models described in the previous example (Section 7.7.2), with also the same uncertainty and input bounds. The number of generators of the zonotope $\mathring{\mathcal{Y}}(i, j)$ (Lemma 7.4) was limited to 1.6^3 times its dimension using Method 3.2. The minimum length optimal input sequence that solves (7.61) was obtained using CPLEX 12.8 and MATLAB 9.1, with $J(\vec{\mathbf{u}}) = \vec{\mathbf{u}}^T \vec{\mathbf{u}}$, $\varepsilon = 1 \cdot 10^{-6}$, and $N = 5$, and it is given by

$$\vec{\mathbf{u}} = \left(\left[\begin{array}{c} 1 \\ 0.7636 \end{array} \right], \left[\begin{array}{c} 1 \\ 0.2664 \end{array} \right], \left[\begin{array}{c} 0.9892 \\ 1 \end{array} \right], \left[\begin{array}{c} -0.0879 \\ 1 \end{array} \right], \left[\begin{array}{c} -0.3478 \\ 1 \end{array} \right], \left[\begin{array}{c} -0.7202 \\ -1 \end{array} \right] \right).$$

Figure 7.7 shows the output reachable sets $Y_0^{[i]}$ and $Y_5^{[i]}$ for $i = 1, 2, 3, 4$, resulting from the injection of the designed $\vec{\mathbf{u}}$. We also show the sets defined by $\mathbf{C}^{[i]}X_0 \oplus \mathbf{D}^{[i]}\mathbf{u}_0 \oplus \mathbf{D}_v^{[i]}V$, which do not take into account the equality constraint (7.39b). Note that these sets are completely overlapped for $i \in \{1, 2, 3, 4\}$, and $Y_0^{[i]}$ are completely overlapped for $i \in \{1, 2, 4\}$. On the other hand, $Y_5^{[i]}$ are disjoint for every $i \in \mathcal{M}$, showing that the injection of $\vec{\mathbf{u}}$ guarantees fault diagnosis at $k = 5$. Figure 7.7 also shows clouds containing 1000 samples of the output $\mathbf{y}_5^{[i]}$ for each model i , demonstrating that the outputs of the models (7.2) are in fact separated. In comparison to the results presented in the previous section using constrained zonotopes, the output reachable tube method using mixed zonotopes did not

³The tuning parameters for this experiment differ from the previous one due to the higher dimension of the output reachable tubes in comparison to the output reachable sets.

require the knowledge of an admissible state set. However, a longer input sequence was required for fault diagnosis. The reasons for this disadvantage are a subject of future investigation.

7.8 Final remarks

This chapter proposed novel algorithms for set-based state estimation and AFD of linear descriptor systems with unknown-but-bounded uncertainties. The methods use constrained zonotopes and mixed zonotopes. The latter is a generalization of zonotopes capable of describing unbounded sets. Both methods lead to significantly tighter results than zonotope approaches. In addition, AFD was enabled without assuming rank properties on the structure of the system. The effectiveness of the new methods was corroborated by numerical examples.

8

Joint state and parameter estimation

This chapter presents a new method for set-based joint state and parameter estimation of nonlinear discrete-time systems. The new method is based on the algorithms proposed in Chapter 5. By extending the nonlinear state estimation methods using constrained zonotopes to include parameter estimation in a unified framework, and therefore, maintaining existing dependencies between states, algebraic variables, and unknown parameters, the accuracy of both state and parameter estimation is significantly improved. The advantages of the new approach are highlighted in two numerical examples.

The chapter is organized as follows. The class of nonlinear systems considered in this chapter is presented in Section 8.1, together with the proposed estimator framework. Section 8.2 develops the proposed method for set-based joint state and parameters estimation using constrained zonotopes. The numerical examples are presented in Section 8.3, and Section 8.4 concludes the chapter.

8.1 Problem formulation

Consider a class of nonlinear discrete-time system with unknown parameters and algebraic variables, described by

$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \boldsymbol{\lambda}_{k-1}, \mathbf{p}, \mathbf{w}_{k-1}), \quad (8.1a)$$

$$\mathbf{0} = \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k, \boldsymbol{\lambda}_k, \mathbf{p}, \mathbf{d}_k), \quad (8.1b)$$

$$\mathbf{y}_k = \mathbf{g}(\mathbf{x}_k, \mathbf{u}_k, \boldsymbol{\lambda}_k, \mathbf{p}, \mathbf{v}_k), \quad (8.1c)$$

where $\mathbf{x}_k \in \mathbb{R}^n$ are the system states, $\mathbf{u}_k \in \mathbb{R}^{n_u}$ are the known inputs, $\mathbf{w}_k \in \mathbb{R}^{n_w}$ and $\mathbf{d}_k \in \mathbb{R}^{n_d}$ are the process uncertainties, $\mathbf{y}_k \in \mathbb{R}^{n_y}$ are the measured outputs, and $\mathbf{v}_k \in \mathbb{R}^{n_v}$ are the measurement uncertainties. Moreover, $\boldsymbol{\lambda}_k \in \mathbb{R}^{n_\lambda}$ are the algebraic variables (i.e., static variables satisfying the algebraic constraints (8.1b)), and $\mathbf{p} \in \mathbb{R}^{n_p}$ are unknown but constant model parameters. The nonlinear functions $\mathbf{f} : \mathbb{R}^n \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_\lambda} \times \mathbb{R}^{n_p} \times \mathbb{R}^{n_w} \rightarrow \mathbb{R}^n$, $\mathbf{g} : \mathbb{R}^n \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_\lambda} \times \mathbb{R}^{n_p} \times \mathbb{R}^{n_v} \rightarrow \mathbb{R}^{n_y}$, and $\mathbf{h} : \mathbb{R}^n \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_\lambda} \times \mathbb{R}^{n_p} \times \mathbb{R}^{n_d} \rightarrow \mathbb{R}^{n_h}$, are assumed to be of class \mathcal{C}^2 . The initial condition, algebraic variables, unknown model parameters, and uncertainties are assumed to be unknown-but-bounded, i.e., $\mathbf{x}_0 \in X_0$, $\boldsymbol{\lambda}_k \in L_k$, $\mathbf{p} \in P$, $\mathbf{w}_k \in W$, $\mathbf{d}_k \in D$, and $\mathbf{v}_k \in V$, $\forall k \geq 0$, where X_0 , L_k , P , W , D , and V are known convex polytopic sets.

Remark 8.1. The nonlinear algebraic constraints (8.1b) may not hold exactly in practice. This condition is described by the presence of the uncertainty $\mathbf{d}_k \in D$.

Remark 8.2. The nonlinear system (8.1) is equivalent to (5.1) in the particular case where $n_\lambda = n_p = n_d = 0$, and if Assumption 5.1 holds.

For any $k \geq 0$, the objective of this chapter is to approximate the solution set of (8.1) as accurately as possible by a guaranteed enclosure $\hat{Z}_k \subset \mathbb{R}^{n+n_\lambda+n_p}$ satisfying $(\mathbf{x}_k, \boldsymbol{\lambda}_k, \mathbf{p}) \in \hat{Z}_k$. We accomplish this here by extending the prediction-update-consistency structure proposed in Chapter 5 to a joint state and parameter estimation framework, considering the presence of both algebraic variables $\boldsymbol{\lambda}_k$ and unknown model parameters \mathbf{p} . The generalized scheme is given by the following recursion:

$$\bar{Z}_k \supseteq \{(\mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \boldsymbol{\lambda}_{k-1}, \mathbf{p}, \mathbf{w}_{k-1}), \boldsymbol{\lambda}_k, \mathbf{p}) : (\mathbf{x}_{k-1}, \boldsymbol{\lambda}_{k-1}, \mathbf{p}) \in \hat{Z}_{k-1}, \boldsymbol{\lambda}_k \in L_k, \mathbf{w}_{k-1} \in W\}, \quad (8.2)$$

$$\tilde{Z}_k \supseteq \{(\mathbf{x}_k, \boldsymbol{\lambda}_k, \mathbf{p}) \in \bar{Z}_k : \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k, \boldsymbol{\lambda}_k, \mathbf{p}, \mathbf{d}_k) = \mathbf{0}, \mathbf{d}_k \in D\}, \quad (8.3)$$

$$\hat{Z}_k \supseteq \{(\mathbf{x}_k, \boldsymbol{\lambda}_k, \mathbf{p}) \in \tilde{Z}_k : \mathbf{g}(\mathbf{x}_k, \mathbf{u}_k, \boldsymbol{\lambda}_k, \mathbf{p}, \mathbf{v}_k) = \mathbf{y}_k, \mathbf{v}_k \in V\}, \quad (8.4)$$

where (8.2) is the *joint prediction step*, (8.3) is the *joint consistency step*, (8.4) is the *joint update step*, and the scheme is initialized with $\bar{Z}_0 \triangleq \bar{X}_0 \times L_0 \times P$ in the consistency step. If \hat{Z}_{k-1} is a valid enclosure of $(\mathbf{x}_{k-1}, \boldsymbol{\lambda}_{k-1}, \mathbf{p})$ for some $k \geq 1$, then standard results in set-valued state estimation show that $(\mathbf{x}_k, \boldsymbol{\lambda}_k, \mathbf{p}) \in \bar{Z}_k$ given by (8.2). By construction, this leads to $(\mathbf{x}_k, \boldsymbol{\lambda}_k, \mathbf{p}) \in \tilde{Z}_k$ and $(\mathbf{x}_k, \boldsymbol{\lambda}_k, \mathbf{p}) \in \hat{Z}_k$, from (8.3) and (8.4), respectively.

Remark 8.3. In contrast to the method proposed in Chapter 5, to improve the accuracy in the update step (8.4), in this chapter we consider computing the joint consistency step prior to the joint update step. This is because the known set L_k enclosing the algebraic variables $\boldsymbol{\lambda}_k$ may be very conservative in practice, with $\boldsymbol{\lambda}_k$ being usually determined by the nonlinear algebraic constraints (8.1b).

In the remainder of the chapter, our goal is to develop methods for computing accurate enclosures for each of the three steps (8.2)–(8.4). The main results of this chapter include

generalizations of the prediction, update, and consistency methods proposed in Chapter 5, which consider state variables, algebraic variables, and unknown parameters, as well as nonlinear dynamics, nonlinear measurement, and nonlinear algebraic constraints.

8.2 Proposed scheme based on constrained zonotopes

In the following, we rewrite Lemma 5.1 to consider more general arguments than just states and process disturbances. This generalized result will be necessary for the new method developed in this chapter.

Lemma 8.1. Let $\alpha : \mathbb{R}^{n_r} \times \mathbb{R}^{n_s} \rightarrow \mathbb{R}^{n_\alpha}$ be of class \mathcal{C}^1 , and let $\nabla_r \alpha$ denote the gradient of α with respect to its first argument. Let $R \subset \mathbb{R}^{n_r}$ and $S \subset \mathbb{R}^{n_s}$ be constrained zonotopes, and let $\mathbf{J} \in \mathbb{I}\mathbb{R}^{n_\alpha \times n_r}$ be an interval matrix satisfying

$$\nabla_r^T \alpha(\square R, S) \triangleq \{\nabla_r^T \alpha(\mathbf{r}, \mathbf{s}) : \mathbf{r} \in \square R, \mathbf{s} \in S\} \subseteq \mathbf{J}. \quad (8.5)$$

Then, for every $\mathbf{r} \in R$, $\mathbf{s} \in S$, and $\gamma_r \in \square R$, there exists $\hat{\mathbf{J}} \in \mathbf{J}$ such that

$$\alpha(\mathbf{r}, \mathbf{s}) = \alpha(\gamma_r, \mathbf{s}) + \hat{\mathbf{J}}(\mathbf{r} - \gamma_r). \quad (8.6)$$

Proof. Choose any $(\mathbf{r}, \mathbf{s}) \in R \times S$. Since $\mathbf{r} \in R \subseteq \square R$ and $\gamma_r \in \square R$, the Mean Value Theorem ensures that, for any $i = 1, 2, \dots, n_\alpha$, $\exists \delta^{[i]} \in \square R$ such that $\alpha_i(\mathbf{r}, \mathbf{s}) = \alpha_i(\gamma_r, \mathbf{s}) + \nabla_r^T \alpha_i(\delta^{[i]}, \mathbf{s})(\mathbf{r} - \gamma_r)$. But $\nabla_r^T \alpha_i(\delta^{[i]}, \mathbf{s})$ is contained in the i -th row of \mathbf{J} by hypothesis, and since this is true for all $i = 1, 2, \dots, n_\alpha$, $\exists \hat{\mathbf{J}} \in \mathbf{J}$ such that $\alpha(\mathbf{r}, \mathbf{s}) = \alpha(\gamma_r, \mathbf{s}) + \hat{\mathbf{J}}(\mathbf{r} - \gamma_r)$. ■

The following proposition provides an enclosure for the state \mathbf{x}_k portion in the prediction step (8.2). A method to compute an enclosure \bar{Z}_k satisfying the joint prediction step is derived based on the result from Proposition 8.1, which is given by Corollary 8.1.

Proposition 8.1. (State prediction) Let $\mathbf{f} : \mathbb{R}^n \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_\lambda} \times \mathbb{R}^{n_p} \times \mathbb{R}^{n_w} \rightarrow \mathbb{R}^n$ be of class \mathcal{C}^1 , and let $\nabla_x \mathbf{f}$ denote the gradient of \mathbf{f} with respect to its first argument. Let $\mathbf{u} \in \mathbb{R}^{n_u}$, and let $X \subset \mathbb{R}^n$, $L \subset \mathbb{R}^{n_\lambda}$, $P \subset \mathbb{R}^{n_p}$, and $W \subset \mathbb{R}^{n_w}$ be constrained zonotopes. Let $Z = X \times L \times P$, and choose any $\gamma_z = (\gamma_x, \gamma_\lambda, \gamma_p) \in \square X \times \square L \times \square P$. If Z_w is a constrained zonotope such that $\mathbf{f}(\gamma_x, \mathbf{u}, \gamma_\lambda, \gamma_p, W) \subseteq Z_w$ and $\mathbf{J} \in \mathbb{I}\mathbb{R}^{n \times n}$ is an interval matrix satisfying $\nabla_x^T \mathbf{f}(\square X, \mathbf{u}, \square L, \square P, W) \subseteq \mathbf{J}$, then $\mathbf{f}(X, \mathbf{u}, L, P, W) \subseteq Z_w \oplus \triangleleft(\mathbf{J}, Z - \gamma_z)$.

Proof. Choose any $(\mathbf{x}, \boldsymbol{\lambda}, \mathbf{p}, \mathbf{w}) \in X \times L \times P \times W$. Lemma 8.1 (with $\alpha \triangleq \mathbf{f}$, $\mathbf{r} \triangleq (\mathbf{x}, \boldsymbol{\lambda}, \mathbf{p})$, $\mathbf{s} \triangleq \mathbf{w}$, $R \triangleq X \times L \times P$, and $S \triangleq W$) ensures that there exists a real matrix $\hat{\mathbf{J}} \in \mathbf{J}$ such that $\mathbf{f}(\mathbf{x}, \mathbf{u}, \boldsymbol{\lambda}, \mathbf{p}, \mathbf{w}) = \mathbf{f}(\gamma_x, \mathbf{u}, \gamma_\lambda, \gamma_p, \mathbf{w}) + \hat{\mathbf{J}}(\mathbf{z} - \gamma_z)$, with $\mathbf{z} = (\mathbf{x}, \boldsymbol{\lambda}, \mathbf{p})$. By Theorem 4.1 and the choice of Z_w , it follows that $\mathbf{f}(\mathbf{x}, \mathbf{u}, \boldsymbol{\lambda}, \mathbf{p}, \mathbf{w}) \in Z_w \oplus \triangleleft(\mathbf{J}, Z - \gamma_z)$, as desired. ■

Corollary 8.1. (Joint prediction) Let $\mathbf{f} : \mathbb{R}^n \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_\lambda} \times \mathbb{R}^{n_p} \times \mathbb{R}^{n_w} \rightarrow \mathbb{R}^n$ be of class \mathcal{C}^1 , and let $\nabla_x \mathbf{f}$ denote the gradient of \mathbf{f} with respect to its first argument. For $k \geq 1$, let $\mathbf{u}_{k-1} \in \mathbb{R}^{n_u}$,

and assume that $\mathbf{z}_{k-1} = (\mathbf{x}_{k-1}, \boldsymbol{\lambda}_{k-1}, \mathbf{p}) \in \hat{Z}_{k-1} = \{\hat{\mathbf{G}}_{k-1}, \hat{\mathbf{c}}_{k-1}, \hat{\mathbf{A}}_{k-1}, \hat{\mathbf{b}}_{k-1}\}$. Moreover, let $\mathbf{w}_{k-1} \in W = \{\mathbf{G}_w, \mathbf{c}_w, \mathbf{A}_w, \mathbf{b}_w\}$, $\boldsymbol{\lambda}_k \in L_k$, and $\mathbf{p} \in P$. Choose any $\boldsymbol{\gamma}_z = (\boldsymbol{\gamma}_x, \boldsymbol{\gamma}_\lambda, \boldsymbol{\gamma}_p) \in \square \hat{Z}_{k-1}$. In addition, let $\mathbf{E} = [\mathbf{0}_{n_p \times n} \quad \mathbf{0}_{n_p \times n_\lambda} \quad \mathbf{I}_{n_p}]$. Let Z_w be a constrained zonotope such that $\mathbf{f}(\boldsymbol{\gamma}_x, \mathbf{u}_{k-1}, \boldsymbol{\gamma}_\lambda, \boldsymbol{\gamma}_p, \mathbf{w}_{k-1}) \subseteq Z_w$ for all $\mathbf{w}_{k-1} \in W$, and let $\mathbf{J}_z \in \mathbb{R}^{n \times n}$ be an interval matrix satisfying $\nabla_z^T \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \boldsymbol{\lambda}_{k-1}, \mathbf{p}, \mathbf{w}_{k-1}) \subseteq \mathbf{J}_z$ for all $(\mathbf{x}_{k-1}, \boldsymbol{\lambda}_{k-1}, \mathbf{p}) \in \square \hat{Z}_{k-1}$, and $\mathbf{w}_{k-1} \in W$. If $\{\hat{\mathbf{G}}, \hat{\mathbf{c}}\}$ is a zonotope with \underline{n}_g generators satisfying $\hat{Z}_{k-1} - \boldsymbol{\gamma}_z \subseteq \{\hat{\mathbf{G}}, \hat{\mathbf{c}}\}$, $\mathbf{m} \triangleq (\mathbf{J}_z - \text{mid}(\mathbf{J}_z))\hat{\mathbf{c}} \in \mathbb{R}^n$, and $\hat{\mathbf{P}} \in \mathbb{R}^{n \times n}$ is a diagonal matrix with $\hat{P}_{ii} = \text{rad}(m_i) + \sum_{j=1}^{\underline{n}_g} \sum_{\ell=1}^{n+n_\lambda+n_p} \text{rad}(J_{z,i\ell})[\hat{G}_{\ell j}]$, then $(\mathbf{x}_k, \boldsymbol{\lambda}_k, \mathbf{p}) = \mathbf{z}_k \in \bar{Z}_k$, with

$$\bar{Z}_k = \begin{bmatrix} \text{mid}(\mathbf{J}_z) \\ \mathbf{0} \\ \mathbf{E} \end{bmatrix} \hat{Z}_{k-1} \oplus \begin{bmatrix} \text{mid}(\mathbf{J}_z) \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} (-\boldsymbol{\gamma}_z) \oplus \begin{bmatrix} \hat{\mathbf{P}} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} B_\infty^n \oplus \begin{bmatrix} \mathbf{I} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} Z_w \oplus \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \\ \mathbf{0} \end{bmatrix} L_k. \quad (8.7)$$

Proof. Choose any $(\mathbf{x}_{k-1}, \boldsymbol{\lambda}_{k-1}, \mathbf{p}) = \mathbf{z}_{k-1} \in \hat{Z}_{k-1}$, $\mathbf{w}_{k-1} \in W$. From (8.1a), Proposition 8.1 and Theorem 4.1, there must exist $\boldsymbol{\delta} \in B_\infty^n$ such that

$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \boldsymbol{\lambda}_{k-1}, \mathbf{p}, \mathbf{w}_{k-1}) = \mathbf{f}(\boldsymbol{\gamma}_x, \mathbf{u}_{k-1}, \boldsymbol{\gamma}_\lambda, \boldsymbol{\gamma}_p, \mathbf{w}_{k-1}) + \text{mid}(\mathbf{J}_z)(\mathbf{z}_{k-1} - \boldsymbol{\gamma}_z) + \hat{\mathbf{P}}\boldsymbol{\delta}$$

with $\hat{\mathbf{P}}$ defined as in the statement of the corollary. Then, since $\boldsymbol{\lambda}_k \in L_k$ and, by the definition of \mathbf{z}_{k-1} and \mathbf{E} , $\mathbf{p} = \mathbf{E}\mathbf{z}_{k-1}$ holds, we have that

$$\begin{aligned} (\mathbf{x}_k, \boldsymbol{\lambda}_k, \mathbf{p}) &= (\mathbf{f}(\boldsymbol{\gamma}_x, \mathbf{u}_{k-1}, \boldsymbol{\gamma}_\lambda, \boldsymbol{\gamma}_p, \mathbf{w}_{k-1}) + \text{mid}(\mathbf{J}_z)(\mathbf{z}_{k-1} - \boldsymbol{\gamma}_z) + \hat{\mathbf{P}}\boldsymbol{\delta}, \boldsymbol{\lambda}_k, \mathbf{E}\mathbf{z}_{k-1}) \\ &= (\text{mid}(\mathbf{J}_z)\mathbf{z}_{k-1} - \text{mid}(\mathbf{J}_z)\boldsymbol{\gamma}_z + \hat{\mathbf{P}}\boldsymbol{\delta} + \mathbf{f}(\boldsymbol{\gamma}_x, \mathbf{u}_{k-1}, \boldsymbol{\gamma}_\lambda, \boldsymbol{\gamma}_p, \mathbf{w}_{k-1}), \boldsymbol{\lambda}_k, \mathbf{E}\mathbf{z}_{k-1}) \\ &= \begin{bmatrix} \text{mid}(\mathbf{J}_z) \\ \mathbf{0} \\ \mathbf{E} \end{bmatrix} \mathbf{z}_{k-1} + \begin{bmatrix} \text{mid}(\mathbf{J}_z) \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} (-\boldsymbol{\gamma}_z) + \begin{bmatrix} \hat{\mathbf{P}} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \boldsymbol{\delta} + \begin{bmatrix} \mathbf{I} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \mathbf{f}(\boldsymbol{\gamma}_x, \mathbf{u}_{k-1}, \boldsymbol{\gamma}_\lambda, \boldsymbol{\gamma}_p, \mathbf{w}_{k-1}) + \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \\ \mathbf{0} \end{bmatrix} \boldsymbol{\lambda}_k \\ &\in \begin{bmatrix} \text{mid}(\mathbf{J}_z) \\ \mathbf{0} \\ \mathbf{E} \end{bmatrix} \hat{Z}_{k-1} \oplus \begin{bmatrix} \text{mid}(\mathbf{J}_z) \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} (-\boldsymbol{\gamma}_z) \oplus \begin{bmatrix} \hat{\mathbf{P}} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} B_\infty^n \oplus \begin{bmatrix} \mathbf{I} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} Z_w \oplus \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \\ \mathbf{0} \end{bmatrix} L_k, \end{aligned}$$

which proves the corollary. \blacksquare

The following two propositions provide enclosures \tilde{Z}_k and \hat{Z}_k satisfying the joint consistency and update steps defined by (8.3) and (8.4), respectively. These results are both based on Lemma 8.1 and Theorem 4.1.

Proposition 8.2. (*Joint consistency*) Let $\mathbf{h} : \mathbb{R}^n \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_\lambda} \times \mathbb{R}^{n_p} \times \mathbb{R}^{n_d} \rightarrow \mathbb{R}^{n_h}$ be of class \mathcal{C}^1 . For $k \geq 0$, let $\mathbf{u}_k \in \mathbb{R}^{n_u}$, assume that $\mathbf{z}_k = (\mathbf{x}_k, \boldsymbol{\lambda}_k, \mathbf{p}) \in \bar{Z}_k = \{\bar{\mathbf{G}}_k, \bar{\mathbf{c}}_k, \bar{\mathbf{A}}_k, \bar{\mathbf{b}}_k\}$, and let $\mathbf{d}_k \in D = \{\mathbf{G}_d, \mathbf{c}_d, \mathbf{A}_d, \mathbf{b}_d\}$. Let $\mathbf{h}(\mathbf{x}_k, \mathbf{u}_k, \boldsymbol{\lambda}_k, \mathbf{p}, \mathbf{d}_k) = \mathbf{0}$ for at least one $(\mathbf{x}_k, \boldsymbol{\lambda}_k, \mathbf{p}) \in \bar{Z}_k$, and $\mathbf{d}_k \in D$. Choose any $\boldsymbol{\gamma}_z = (\boldsymbol{\gamma}_x, \boldsymbol{\gamma}_\lambda, \boldsymbol{\gamma}_p) \in \square \bar{Z}_k$. Let $Z_d = \{\mathbf{G}_{Z_d}, \mathbf{c}_{Z_d}, \mathbf{A}_{Z_d}, \mathbf{b}_{Z_d}\}$ be a constrained zonotope such that $-\mathbf{h}(\boldsymbol{\gamma}_x, \mathbf{u}, \boldsymbol{\gamma}_\lambda, \boldsymbol{\gamma}_p, \mathbf{d}_k) \in Z_d = \{\mathbf{G}_{Z_d}, \mathbf{c}_{Z_d}, \mathbf{A}_{Z_d}, \mathbf{b}_{Z_d}\}$ for all $\mathbf{d}_k \in D$, and let

$\mathbf{J}_z \in \mathbb{R}^{n_h \times n + n_\lambda + n_p}$ satisfy $\nabla_z^T \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k, \boldsymbol{\lambda}_k, \mathbf{p}, \mathbf{d}_k) \subseteq \mathbf{J}_z$ for all $(\mathbf{x}_k, \boldsymbol{\lambda}_k, \mathbf{p}) \in \bar{Z}_k$ and $\mathbf{d}_k \in D$. If $\{\bar{\mathbf{G}}, \bar{\mathbf{c}}\}$ is a zonotope with \underline{n}_g generators satisfying $\bar{Z}_k - \gamma_z \subseteq \{\bar{\mathbf{G}}, \bar{\mathbf{c}}\}$, $\mathbf{m} \triangleq (\mathbf{J}_z - \text{mid}(\mathbf{J}_z))\bar{\mathbf{c}} \in \mathbb{R}^n$, and $\bar{\mathbf{P}} \in \mathbb{R}^{n_h \times n_h}$ is a diagonal matrix with $\bar{P}_{ii} = \text{rad}(m_i) + \sum_{j=1}^{\underline{n}_g} \sum_{\ell=1}^{n+n_\lambda+n_p} \text{rad}(J_{z,i\ell})|\bar{G}_{\ell j}|$, then $\{(\mathbf{x}_k, \boldsymbol{\lambda}_k, \mathbf{p}) \in \bar{Z}_k : \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k, \boldsymbol{\lambda}_k, \mathbf{p}, \mathbf{d}_k) = \mathbf{0}, \mathbf{d}_k \in D\} \subseteq \tilde{Z}_k$, where

$$\tilde{Z}_k = \left\{ \left[\bar{\mathbf{G}}_z \quad \mathbf{0} \quad \mathbf{0} \right], \bar{\mathbf{c}}_z, \begin{bmatrix} \bar{\mathbf{A}}_z & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{A}_{Z_d} \\ \text{mid}(\mathbf{J}_z)\bar{\mathbf{G}}_z & -\bar{\mathbf{P}} & -\mathbf{G}_{Z_d} \end{bmatrix}, \begin{bmatrix} \bar{\mathbf{b}}_z \\ \mathbf{b}_{Z_d} \\ \text{mid}(\mathbf{J}_z)(\gamma_z - \bar{\mathbf{c}}_z) + \mathbf{c}_{Z_d} \end{bmatrix} \right\}. \quad (8.8)$$

Proof. Choose any $(\mathbf{x}_k, \boldsymbol{\lambda}_k, \mathbf{p}) = \mathbf{z}_k \in \bar{Z}_k$, $\mathbf{d}_k \in D$, satisfying $\mathbf{h}(\mathbf{x}_k, \mathbf{u}_k, \boldsymbol{\lambda}_k, \mathbf{p}, \mathbf{d}_k) = \mathbf{0}$. Lemma 8.1 (with $\boldsymbol{\alpha} \triangleq \mathbf{h}$, $\mathbf{r} \triangleq (\mathbf{x}_k, \boldsymbol{\lambda}_k, \mathbf{p})$, $\mathbf{s} \triangleq \mathbf{d}_k$, $R \triangleq \bar{Z}_k$, and $S \triangleq D$) ensures that there exists a real matrix $\hat{\mathbf{J}} \in \mathbf{J}_z$ such that $\mathbf{h}(\mathbf{x}_k, \mathbf{u}_k, \boldsymbol{\lambda}_k, \mathbf{p}, \mathbf{d}_k) = \mathbf{h}(\gamma_x, \mathbf{u}, \gamma_\lambda, \gamma_p, \mathbf{d}_k) + \hat{\mathbf{J}}_k(\mathbf{z}_k - \gamma_z)$, with $\mathbf{z} = (\mathbf{x}, \boldsymbol{\lambda}, \mathbf{p})$. Therefore, from Theorem 4.1, there must exist $\boldsymbol{\delta} \in B_\infty^{n_h}$ such that

$$\mathbf{0} = \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k, \boldsymbol{\lambda}_k, \mathbf{p}, \mathbf{d}_k) = \mathbf{h}(\gamma_x, \mathbf{u}_k, \gamma_\lambda, \gamma_p, \mathbf{d}_k) + \text{mid}(\mathbf{J}_z)(\mathbf{z}_k - \gamma_z) + \bar{\mathbf{P}}\boldsymbol{\delta},$$

with $\bar{\mathbf{P}}$ defined as in the statement of this proposition. The equation above can be rewritten as $\text{mid}(\mathbf{J}_z)\mathbf{z}_k = -\mathbf{h}(\gamma_x, \mathbf{u}_k, \gamma_\lambda, \gamma_p, \mathbf{d}_k) + \text{mid}(\mathbf{J}_z)\gamma_z - \bar{\mathbf{P}}\boldsymbol{\delta} \in (\text{mid}(\mathbf{J}_z)\gamma_z) \oplus Z_d \oplus \bar{\mathbf{P}}B_\infty^{n_h}$. Therefore, $\{(\mathbf{x}_k, \boldsymbol{\lambda}_k, \mathbf{p}) \in \bar{Z}_k : \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k, \boldsymbol{\lambda}_k, \mathbf{p}, \mathbf{d}_k) = \mathbf{0}, \mathbf{d}_k \in D\} \subseteq \{(\mathbf{x}_k, \boldsymbol{\lambda}_k, \mathbf{p}) = \mathbf{z}_k \in \bar{Z}_k : \text{mid}(\mathbf{J}_z)\mathbf{z}_k \in (\text{mid}(\mathbf{J}_z)\gamma_z) \oplus Z_d \oplus \bar{\mathbf{P}}B_\infty^{n_h}\}$. By considering the generalized intersection (3.10) and the definition of \tilde{Z}_k , the proposition is proven. \blacksquare

Proposition 8.3. (*Joint update*) Let $\mathbf{g} : \mathbb{R}^n \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_\lambda} \times \mathbb{R}^{n_p} \times \mathbb{R}^{n_v} \rightarrow \mathbb{R}^{n_y}$ be of class \mathcal{C}^1 . For $k \geq 0$, let $\mathbf{u}_k \in \mathbb{R}^{n_u}$, assume that $\mathbf{z}_k = (\mathbf{x}_k, \boldsymbol{\lambda}_k, \mathbf{p}) \in \tilde{Z}_k = \{\tilde{\mathbf{G}}_k, \tilde{\mathbf{c}}_k, \tilde{\mathbf{A}}_k, \tilde{\mathbf{b}}_k\}$, and let $\mathbf{v}_k \in V = \{\mathbf{G}_v, \mathbf{c}_v, \mathbf{A}_v, \mathbf{b}_v\}$. Let $\mathbf{y}_k \in \mathbb{R}^{n_y}$ such that $\mathbf{y}_k = \mathbf{g}(\mathbf{x}_k, \mathbf{u}_k, \boldsymbol{\lambda}_k, \mathbf{p}, \mathbf{v}_k)$ for some $(\mathbf{x}_k, \boldsymbol{\lambda}_k, \mathbf{p}) \in \tilde{Z}_k$, and $\mathbf{v}_k \in V$. Choose any $\gamma_z = (\gamma_x, \gamma_\lambda, \gamma_p) \in \square\tilde{Z}_k$. Let $Z_v = \{\mathbf{G}_{Z_v}, \mathbf{c}_{Z_v}, \mathbf{A}_{Z_v}, \mathbf{b}_{Z_v}\}$ be a constrained zonotope such that $-\mathbf{g}(\gamma_x, \mathbf{u}, \gamma_\lambda, \gamma_p, \mathbf{v}_k) \in Z_v = \{\mathbf{G}_{Z_v}, \mathbf{c}_{Z_v}, \mathbf{A}_{Z_v}, \mathbf{b}_{Z_v}\}$ for all $\mathbf{v}_k \in V$, and let $\mathbf{J} \in \mathbb{R}^{n_y \times n + n_\lambda + n_p}$ satisfy $\nabla_z^T \mathbf{g}(\mathbf{x}_k, \mathbf{u}_k, \boldsymbol{\lambda}_k, \mathbf{p}, \mathbf{v}_k) \subseteq \mathbf{J}_z$ for all $(\mathbf{x}_k, \boldsymbol{\lambda}_k, \mathbf{p}) \in \tilde{Z}_k$ and $\mathbf{v}_k \in V$. If $\{\tilde{\mathbf{G}}, \tilde{\mathbf{c}}\}$ is a zonotope with \underline{n}_g generators satisfying $\tilde{Z}_k - \gamma_z \subseteq \{\tilde{\mathbf{G}}, \tilde{\mathbf{c}}\}$, $\mathbf{m} \triangleq (\mathbf{J}_z - \text{mid}(\mathbf{J}_z))\tilde{\mathbf{c}} \in \mathbb{R}^n$, and $\tilde{\mathbf{P}} \in \mathbb{R}^{n \times n}$ is a diagonal matrix with $\tilde{P}_{ii} = \text{rad}(m_i) + \sum_{j=1}^{\underline{n}_g} \sum_{\ell=1}^{n+n_\lambda+n_p} \text{rad}(J_{z,i\ell})|\tilde{G}_{\ell j}|$, then $\{(\mathbf{x}_k, \boldsymbol{\lambda}_k, \mathbf{p}) \in \tilde{Z}_k : \mathbf{g}(\mathbf{x}_k, \mathbf{u}_k, \boldsymbol{\lambda}_k, \mathbf{p}, \mathbf{v}_k) = \mathbf{y}_k, \mathbf{v}_k \in V\} \subseteq \hat{Z}_k$, where

$$\hat{Z}_k = \left\{ \left[\tilde{\mathbf{G}}_z \quad \mathbf{0} \quad \mathbf{0} \right], \tilde{\mathbf{c}}_z, \begin{bmatrix} \tilde{\mathbf{A}}_z & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{A}_{Z_v} \\ \text{mid}(\mathbf{J})\tilde{\mathbf{G}}_z & -\tilde{\mathbf{P}} & -\mathbf{G}_{Z_v} \end{bmatrix}, \begin{bmatrix} \tilde{\mathbf{b}}_z \\ \mathbf{b}_{Z_v} \\ \text{mid}(\mathbf{J}_z)(\gamma_z - \tilde{\mathbf{c}}_z) + \mathbf{y}_k + \mathbf{c}_{Z_v} \end{bmatrix} \right\}. \quad (8.9)$$

Proof. Choose any $(\mathbf{x}_k, \boldsymbol{\lambda}_k, \mathbf{p}) = \mathbf{z}_k \in \tilde{Z}_k$, $\mathbf{v}_k \in V$, satisfying $\mathbf{g}(\mathbf{x}_k, \mathbf{u}_k, \boldsymbol{\lambda}_k, \mathbf{p}, \mathbf{v}_k) = \mathbf{y}_k$. Lemma 8.1 (with $\boldsymbol{\alpha} \triangleq \mathbf{g}$, $\mathbf{r} \triangleq (\mathbf{x}_k, \boldsymbol{\lambda}_k, \mathbf{p})$, $\mathbf{s} \triangleq \mathbf{v}_k$, $R \triangleq \tilde{Z}_k$, and $S \triangleq V$) ensures that there exists a real matrix $\hat{\mathbf{J}} \in \mathbf{J}_z$ such that $\mathbf{g}(\mathbf{x}_k, \mathbf{u}_k, \boldsymbol{\lambda}_k, \mathbf{p}, \mathbf{v}_k) = \mathbf{g}(\gamma_x, \mathbf{u}, \gamma_\lambda, \gamma_p, \mathbf{v}_k) + \hat{\mathbf{J}}_k(\mathbf{z}_k - \gamma_z)$, with

$\mathbf{z} = (\mathbf{x}, \boldsymbol{\lambda}, \mathbf{p})$. Therefore, from Theorem 4.1, there must exist $\boldsymbol{\delta} \in B_\infty^{n_y}$ such that

$$\mathbf{y}_k = \mathbf{g}(\mathbf{x}_k, \mathbf{u}_k, \boldsymbol{\lambda}_k, \mathbf{p}, \mathbf{v}_k) = \mathbf{g}(\boldsymbol{\gamma}_x, \mathbf{u}_k, \boldsymbol{\gamma}_\lambda, \boldsymbol{\gamma}_p, \mathbf{v}_k) + \text{mid}(\mathbf{J}_z)(\mathbf{z}_k - \boldsymbol{\gamma}_z) + \tilde{\mathbf{P}}\boldsymbol{\delta},$$

with $\tilde{\mathbf{P}}$ defined as in the statement of this proposition. The equation above can be rewritten as $\text{mid}(\mathbf{J}_z)\mathbf{z}_k = \mathbf{y}_k - \mathbf{g}(\boldsymbol{\gamma}_x, \mathbf{u}_k, \boldsymbol{\gamma}_\lambda, \boldsymbol{\gamma}_p, \mathbf{v}_k) + \text{mid}(\mathbf{J}_z)\boldsymbol{\gamma}_z - \tilde{\mathbf{P}}\boldsymbol{\delta} \in (\mathbf{y}_k + \text{mid}(\mathbf{J}_z)\boldsymbol{\gamma}_z) \oplus Z_v \oplus \tilde{\mathbf{P}}B_\infty^{n_y}$. Therefore, $\{(\mathbf{x}_k, \boldsymbol{\lambda}_k, \mathbf{p}) \in \tilde{Z}_k : \mathbf{g}(\mathbf{x}_k, \mathbf{u}_k, \boldsymbol{\lambda}_k, \mathbf{p}, \mathbf{v}_k) = \mathbf{y}_k, \mathbf{v}_k \in V\} \subseteq \{(\mathbf{x}_k, \boldsymbol{\lambda}_k, \mathbf{p}) = \mathbf{z}_k \in \tilde{Z}_k : \text{mid}(\mathbf{J}_z)\mathbf{z}_k \in (\mathbf{y}_k + \text{mid}(\mathbf{J}_z)\boldsymbol{\gamma}_z) \oplus Z_v \oplus \tilde{\mathbf{P}}B_\infty^{n_y}\}$. By considering the generalized intersection (3.10) and the definition of \hat{Z}_k , the proposition is proven. \blacksquare

Remark 8.4. The constrained zonotopes Z_w , Z_d , and Z_v in Corollary 8.1, and Propositions 8.2 and 8.3, can be obtained as $Z_w = \mathbf{f}(\boldsymbol{\gamma}_x, \mathbf{u}_{k-1}, \boldsymbol{\gamma}_\lambda, \boldsymbol{\gamma}_p, \boldsymbol{\gamma}_w) \oplus \triangleleft(\mathbf{J}_w, W - \boldsymbol{\gamma}_w) \supseteq \mathbf{f}(\boldsymbol{\gamma}_x, \mathbf{u}, \boldsymbol{\gamma}_\lambda, \boldsymbol{\gamma}_p, W)$, $Z_d = -\mathbf{h}(\boldsymbol{\gamma}_x, \mathbf{u}_k, \boldsymbol{\gamma}_\lambda, \boldsymbol{\gamma}_p, \boldsymbol{\gamma}_d) \oplus \triangleleft(-\mathbf{J}_d, D - \boldsymbol{\gamma}_d) \supseteq -\mathbf{h}(\boldsymbol{\gamma}_x, \mathbf{u}, \boldsymbol{\gamma}_\lambda, \boldsymbol{\gamma}_p, D)$, and $Z_v = -\mathbf{g}(\boldsymbol{\gamma}_x, \mathbf{u}, \boldsymbol{\gamma}_\lambda, \boldsymbol{\gamma}_p, \boldsymbol{\gamma}_v) \oplus \triangleleft(-\mathbf{J}_v, V - \boldsymbol{\gamma}_v) \supseteq -\mathbf{g}(\boldsymbol{\gamma}_x, \mathbf{u}, \boldsymbol{\gamma}_\lambda, \boldsymbol{\gamma}_p, V)$, respectively, for a chosen $(\boldsymbol{\gamma}_w, \boldsymbol{\gamma}_d, \boldsymbol{\gamma}_v) \in \square W \times \square D \times \square V$, and interval matrices $\mathbf{J}_w \supseteq \nabla_w^T \mathbf{f}(\boldsymbol{\gamma}_x, \mathbf{u}, \boldsymbol{\gamma}_\lambda, \boldsymbol{\gamma}_p, \square W)$, $\mathbf{J}_d \supseteq \nabla_d^T \mathbf{h}(\boldsymbol{\gamma}_x, \mathbf{u}, \boldsymbol{\gamma}_\lambda, \boldsymbol{\gamma}_p, \square D)$, and $\mathbf{J}_v \supseteq \nabla_v^T \mathbf{g}(\boldsymbol{\gamma}_x, \mathbf{u}, \boldsymbol{\gamma}_\lambda, \boldsymbol{\gamma}_p, \square V)$.

Remark 8.5. If the constrained zonotopes $(\hat{Z}_{k-1}, \bar{Z}_k, \tilde{Z}_k, L_k, W, D, V)$ have $(\hat{n}_g, \bar{n}_g, \tilde{n}_g, n_{g_\lambda}, n_{g_w}, n_{g_d}, n_{g_v})$ generators, and $(\hat{n}_c, \bar{n}_c, \tilde{n}_c, n_{c_\lambda}, n_{c_w}, n_{c_d}, n_{c_v})$ constraints, respectively, then the enclosures obtained by: (i) Corollary 8.1 have $\hat{n}_g + 2n + n_{g_w} + n_{g_\lambda}$ generators and $\hat{n}_c + n_{c_w} + n_{g_\lambda}$ constraints; (ii) Proposition 8.2 have $\bar{n}_g + 2n_h + n_{g_d}$ generators and $\bar{n}_c + n_h + n_{c_d}$ constraints; and (iii) Proposition 8.3 have $\tilde{n}_g + 2n_y + n_{g_v}$ generators and $\tilde{n}_c + n_y + n_{c_v}$ constraints.

The following Algorithm summarizes the set-based joint state and parameter estimation method proposed in this chapter, which is denoted by CZMV-J.

Algorithm 8.1 CZMV-J

- 1: (Joint prediction step) Given the constrained zonotopes $\hat{Z}_{k-1} \times L_k \times W \subset \mathbb{R}^{n+n_\lambda+n_p} \times \mathbb{R}^{n_\lambda} \times \mathbb{R}^{n_w}$, and input $\mathbf{u}_{k-1} \in \mathbb{R}^{n_u}$, compute the joint prediction enclosure \bar{Z}_k given by (8.7). This enclosure satisfies (8.2).
 - 2: (Joint consistency step) Given the constrained zonotopes $\bar{Z}_k \times D \subset \mathbb{R}^{n+n_\lambda+n_p} \times \mathbb{R}^{n_d}$, and input $\mathbf{u}_k \in \mathbb{R}^{n_u}$, compute the joint consistency enclosure \tilde{Z}_k given by (8.8). This enclosure satisfies (8.3).
 - 3: (Joint update step) Given the constrained zonotopes $\tilde{Z}_k \times V \subset \mathbb{R}^{n+n_\lambda+n_p} \times \mathbb{R}^{n_v}$, and input $\mathbf{u}_k \in \mathbb{R}^{n_u}$, compute the joint update enclosure \hat{Z}_k given by (8.9). This enclosure satisfies (8.4).
-

Remark 8.6. In the set-based joint and state estimation method developed in this chapter, we use the approximation point $\boldsymbol{\gamma}_z$ given by $C2$ proposed in Chapter 5 (see Section 5.2.4).

8.3 Numerical example

This section presents numerical results for the set-based joint state and parameter estimation method proposed in this chapter. We also compare the results provided by CZMV-J with the method proposed in Chapter 5 based on the mean value extension, denoted by CZMV (in which the model parameters \mathbf{p} are additional components to the uncertainties \mathbf{w}_k), and with a ‘naive’ methodology denoted by CZMV-N. In this latter, the set-based joint and state estimation in (8.2)–(8.4) is performed without the proposed unified framework, i.e., the sets \bar{Z}_k , \tilde{Z}_k , and \hat{Z}_k are disjointed between each step of the algorithm into three lower-dimensional enclosures, given by $\bar{X}_k \times \bar{L}_k \times \bar{P}_k$, $\tilde{X}_k \times \tilde{L}_k \times \tilde{P}_k$, and $\hat{X}_k \times \hat{L}_k \times \hat{P}_k$, respectively. The disadvantage of CZMV-N with respect to CZMV-J is related to the fact that the mutual coupling between states, algebraic variables, and unknown parameters is ignored from one step of the algorithm to the next step.

8.3.1 Example 1

Consider a nonlinear discrete-time system with nonlinear dynamics and nonlinear measurement described by

$$\begin{aligned} x_{1,k} &= 3x_{1,k-1} - px_{1,k-1}^2 - \frac{4x_{1,k-1}x_{2,k-1}}{4 + x_{1,k-1}} + w_{1,k-1} \\ x_{2,k} &= -2x_{2,k-1} + \frac{3x_{1,k-1}x_{2,k-1}}{4 + x_{1,k-1}} + w_{2,k-1} \\ y_{1,k} &= x_{1,k} - \sin\left(\frac{x_{2,k}}{2}\right) + v_{1,k}, \\ y_{2,k} &= -x_{1,k}x_{2,k} + x_{2,k} + (7p - 1) + v_{2,k}, \end{aligned} \quad (8.10)$$

with $\|\mathbf{w}_k\|_\infty \leq 0.2$, $\|\mathbf{v}_k\|_\infty \leq 0.1$, and $p \in P \subset \mathbb{R}$ is an unknown model parameter. The initial state enclosure \bar{X}_0 and the known parameter bounds P are zonotopes given by

$$\bar{X}_0 = \left\{ \left[\begin{array}{ccc} 0.3 & 0.6 & -0.3 \\ 0.3 & 0.3 & 0 \end{array} \right], \left[\begin{array}{c} 10 \\ 0.5 \end{array} \right] \right\}, P = \{5, 1/7\}. \quad (8.11)$$

To obtain process measurements, (8.10) was simulated with $\mathbf{x}_0 = (10.2, 0.65) \in \bar{X}_0$, and $p = 1/7 \in P$, and the process and measurement uncertainties are generated from uniform random distributions. The numbers of generators and constraints of the estimated state and parameter enclosures were limited to 8 and 3, respectively.

Figure 8.1 shows the initial set \bar{X}_0 , together with the enclosures obtained by CZMV, CZMV-N, and CZMV-J, in the update step using \mathbf{y}_0 , projected into the state space. It can be noticed that the enclosure provided by all the estimators for the state \mathbf{x}_0 are equivalent. However, in Figure 8.2, which shows the set $\bar{X}_0 \times P$ together with the enclosures obtained by CZMV, CZMV-N, and CZMV-J in the first update step, it can be noticed that while the

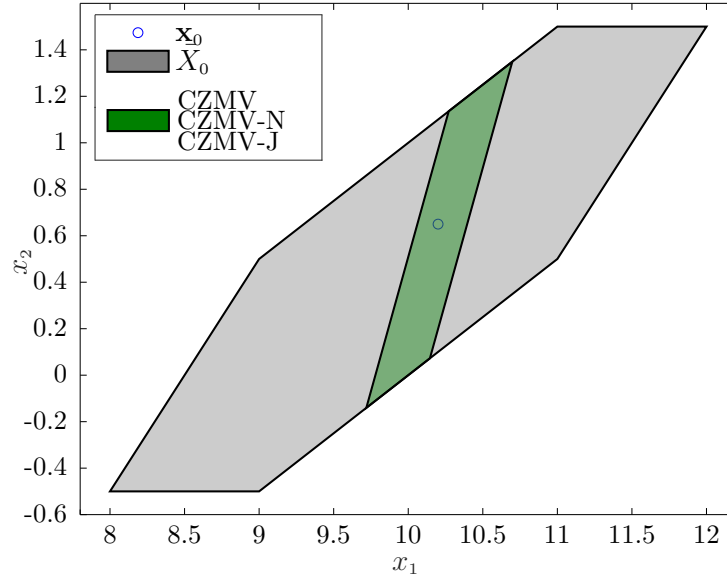


Figure 8.1: The initial state \mathbf{x}_0 , the set \bar{X}_0 , and the projections onto \mathbf{x} of the enclosures obtained by CZMV, CZMV-N, and CZMV-J, in the update step using \mathbf{y}_0 for the first example (which are exactly the same in this case).

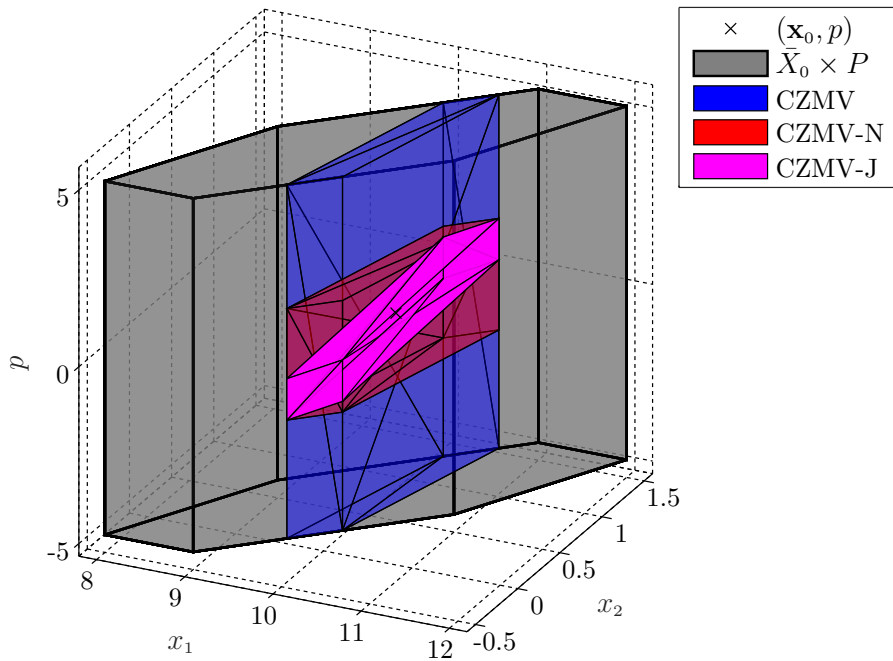


Figure 8.2: The variable (\mathbf{x}_0, p) , the initial set $\bar{X}_0 \times P$, and the enclosures obtained by CZMV, CZMV-N, and CZMV-J, in the first update step for the first example.

parameter uncertainty remains the same with CZMV (since the parameters are considered as disturbances, and therefore not estimated), while by using CZMV-N the enclosure P is significantly refined. In addition, using the proposed unified approach given by CZMV-J, the obtained enclosure is still smaller, since it takes into account the mutual coupling between states and parameter, which is a result from the output equation in (8.10).

Figure 8.3 shows the projections \hat{X}_K of the enclosures obtained using CZMV, CZMV-N,

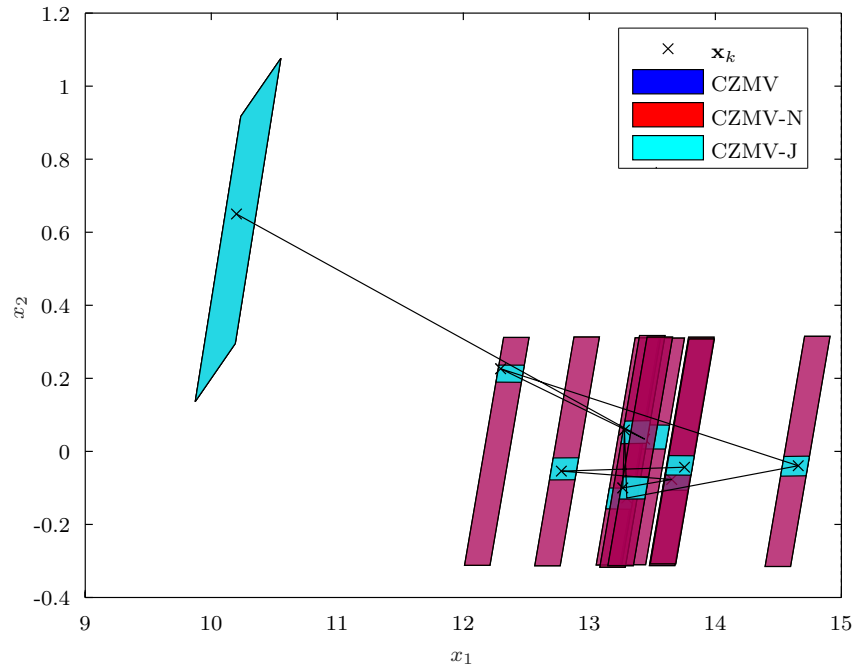


Figure 8.3: The states \mathbf{x}_k , and the projections onto the state space of the enclosures provided by CZMV, CZMV-N, and CZMV-J, for the first example and $k \in [0, 200]$.

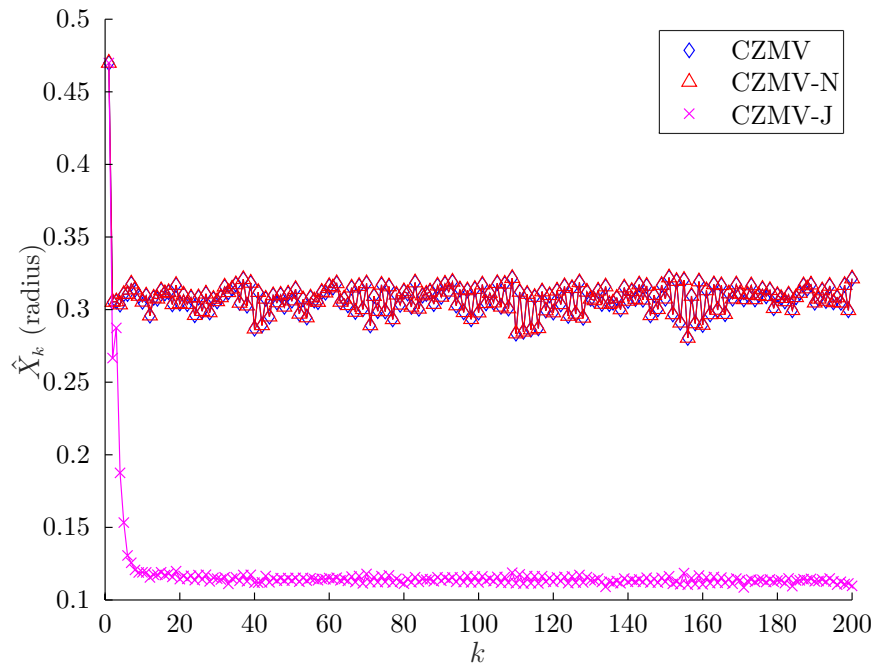


Figure 8.4: The radii of the projections onto the state space of the enclosures provided by CZMV, CZMV-N, and CZMV-J, for the first example and $k \in [0, 200]$.

and CZMV-J, for $k \in [0, 200]$, in intervals of 20 time steps each. It can be noticed that both the previous state estimation method CZMV and the naive strategy CZMV-N are not able to provide accurate enclosures of the states, while the proposed method CZMV-J is able to provide sharp bounds. This is corroborated by Figures 8.4 and 8.5, which show the radii and volumes of the computed state enclosures.

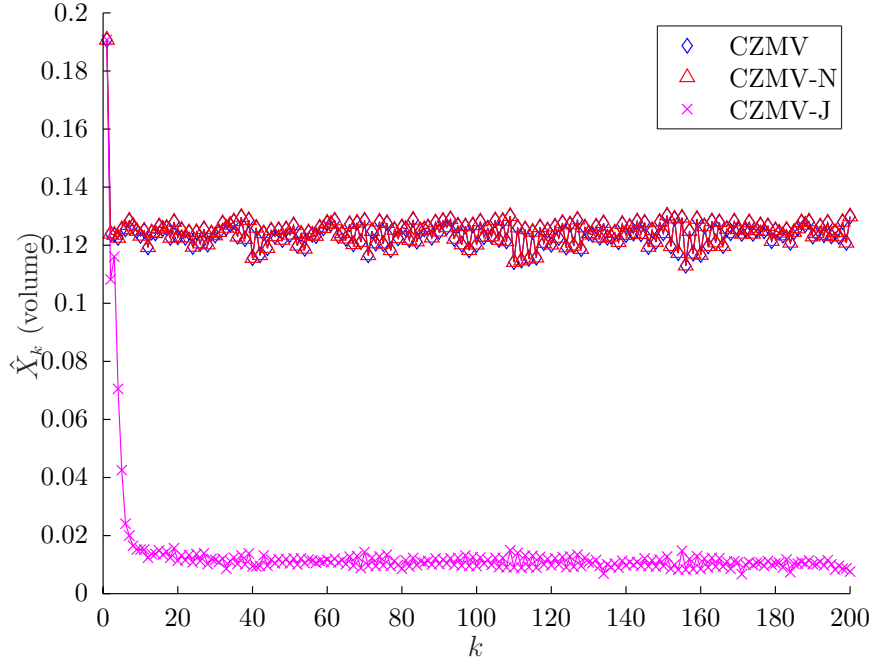


Figure 8.5: The volumes of the projections onto the state space of the enclosures provided by CZMV, CZMV-N, and CZMV-J, for the first example and $k \in [0, 200]$.

Figure 8.6 illustrates the radii of the parameter enclosures provided by CZMV, CZMV-N, and CZMV-J. As it can be noticed, the parameter enclosures used in CZMV are not refined during the entire simulation. On the other hand, CZMV-N is able to refine the parameter bounds, but it is not sufficiently accurate to result in tighter state bounds than CZMV (as shown in Figure 8.3). Nevertheless, the parameter bounds provided by CZMV-J are significantly refined over time. Consequently, this resulted in much more accurate state enclosures as expected, in comparison to the CZMV and CZMV-N, which do consider the unified framework proposed in this chapter.

8.3.2 Example 2

Consider a nonlinear discrete-time system with nonlinear dynamics and nonlinear algebraic constraints described by (Sjöberg & Glad, 2005)

$$\begin{aligned}
 x_{1,k} &= x_{1,k-1} + T_s(p x_{2,k-1}), \\
 x_{2,k} &= x_{2,k-1} + T_s(e^{\lambda_{k-1}} - 1 + 0.5u_{k-1}), \\
 0 &= x_{1,k} - \arcsin(1 - e^{\lambda_k} + 0.5u_k), \\
 y_{1,k} &= p x_{1,k} + v_{1,k}, \\
 y_{2,k} &= x_{2,k} + v_{2,k},
 \end{aligned} \tag{8.12}$$

with $\|\mathbf{v}_k\|_\infty \leq 0.01$, $T_s = 0.01$ s, and $p \in P \subset \mathbb{R}$ is an unknown model parameter. The initial state enclosure \bar{X}_0 , the known bounds L_k on the algebraic variable λ_k , and the known

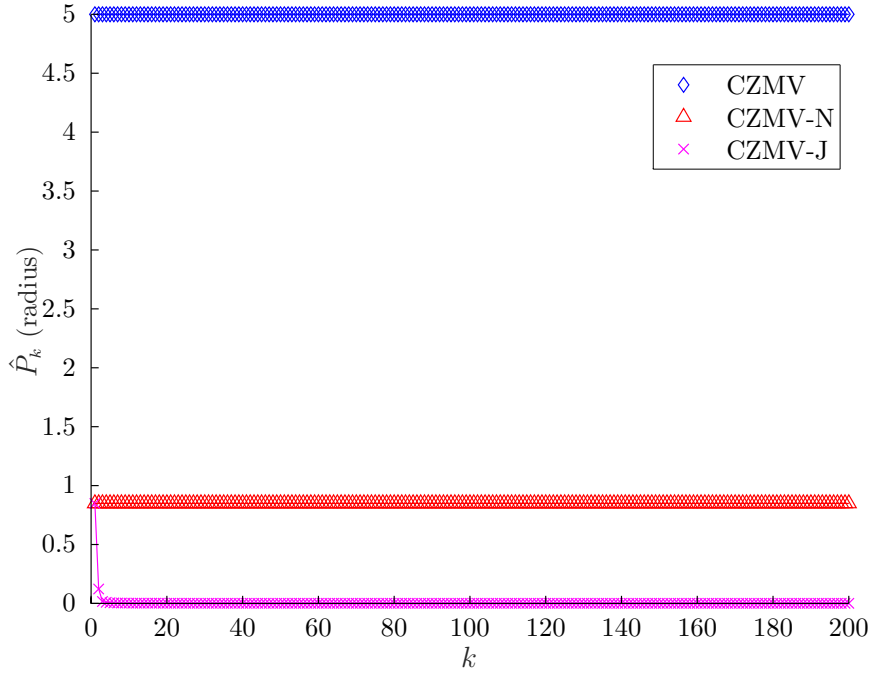


Figure 8.6: The enclosures of the unknown parameter p provided by CZMV, CZMV-N, and CZMV-J, for the first example and $k \in [0, 200]$.

parameter bounds P , are zonotopes given by

$$\bar{X}_0 = \left\{ \begin{bmatrix} 0.1 & 0.2 & -0.1 \\ 0.1 & 0.1 & 0 \end{bmatrix}, \begin{bmatrix} -0.2 \\ -0.15 \end{bmatrix} \right\}, \quad L_k = \{0.1, \lambda_k\}, \quad P = \{0.1, 1\}. \quad (8.13)$$

The initial set \bar{Z}_0 is given by $\bar{Z}_0 = \bar{X}_0 \times L_0 \times P$. To obtain process measurements, (8.12) was simulated with $\mathbf{x}_0 = (0, 0) \in \bar{X}_0$, and $p = 1 \in P$, and the process and measurement uncertainties are generated from uniform random distributions. For simulation purposes, the algebraic variable λ_k is obtained by solving the nonlinear algebraic constraint (8.12) at each time step k . The numbers of generators and constraints of the estimated state, algebraic variable, and parameter enclosures were limited to 8 and 3, respectively.

Figure 8.7 shows the projections \hat{X}_k of the enclosures obtained using CZMV, CZMV-N, and CZMV-J, for $k \in [0, 500]$, in intervals of 50 time steps each. As it can be noticed, both CZMV and CZMV-N are able to provide reasonable enclosures of the states. However, CZMV-J is able to provide sharper bounds in comparison to the other methods. This is corroborated by Figure 8.8, which show the volumes of the computed state enclosures.

Figure 8.9 illustrates the radii of the enclosures \hat{L}_k of λ_k provided by CZMV, CZMV-N, and CZMV-J. As expected, the enclosures \hat{L}_k used in CZMV are not refined during the entire simulation. On the other hand, both CZMV-N and CZMV-J are able to refine the enclosures of the algebraic variable, but the latter is notably more accurate. In addition, Figure 8.10 illustrates the radii of the parameter enclosures provided by CZMV, CZMV-N, and CZMV-J. As in the case of the algebraic variable λ_k , the parameter enclosures used

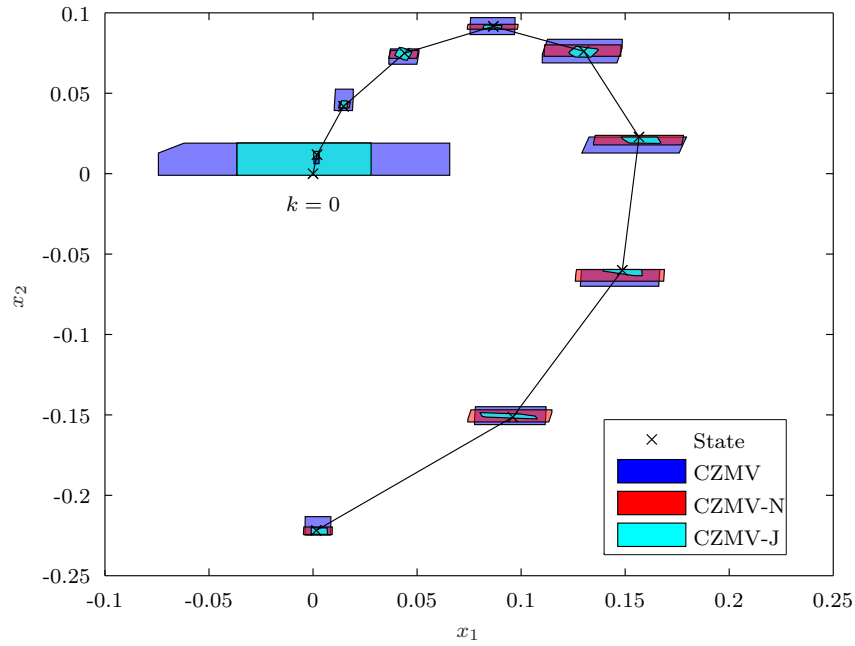


Figure 8.7: The states \mathbf{x}_k , and the projections onto the state space of the enclosures provided by CZMV, CZMV-N, and CZMV-J, for the second example and $k \in [0, 500]$.

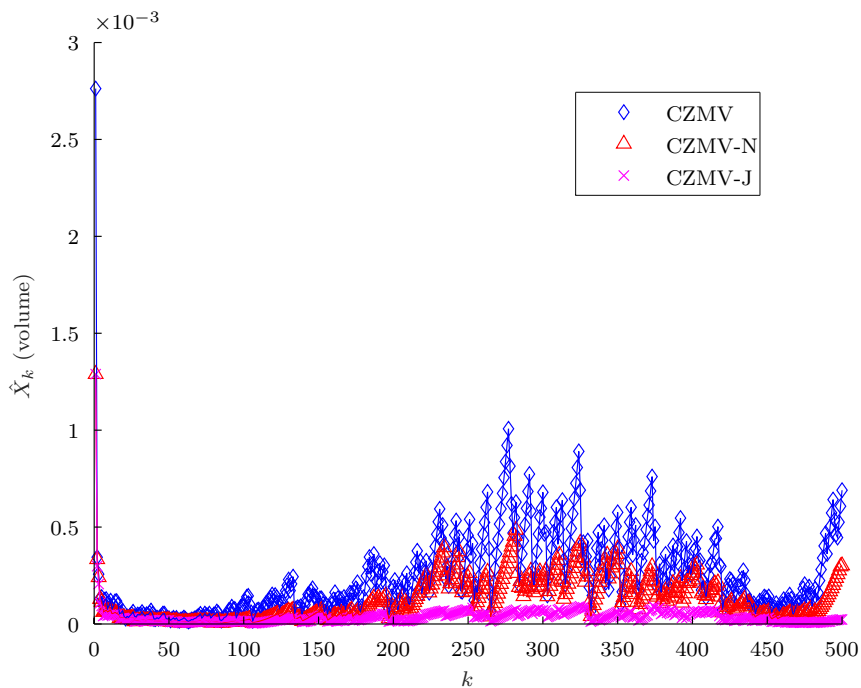


Figure 8.8: The volumes of the projections onto the state space of the enclosures provided by CZMV, CZMV-N, and CZMV-J, for the second example and $k \in [0, 500]$.

in CZMV are not refined during the simulation. On the other side, CZMV-N is also not able to refine the parameter bounds in this case, while the parameter bounds provided by CZMV-J are significantly refined over time. This result corroborates the advantages of using the proposed unified framework for joint state and parameter estimation in comparison to other techniques.

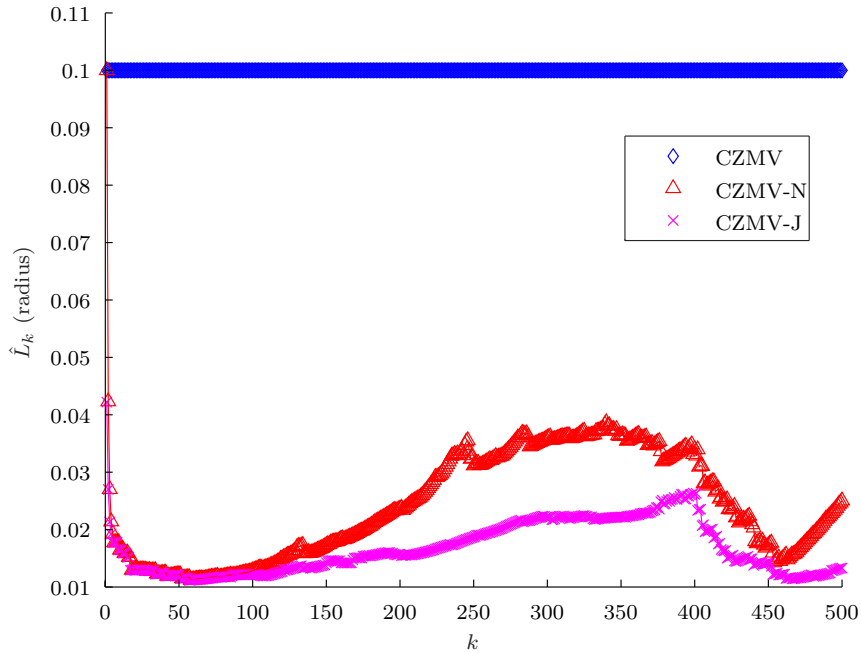


Figure 8.9: The enclosures of the algebraic variable λ_k provided by CZMV, CZMV-N, and CZMV-J, for the second example and $k \in [0, 500]$.

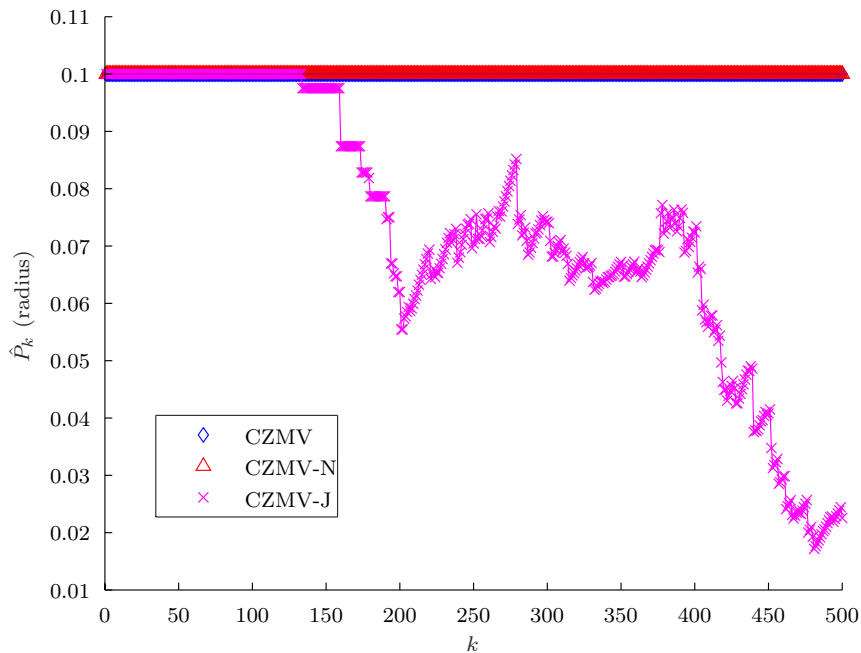


Figure 8.10: The enclosures of the unknown parameter p provided by CZMV, CZMV-N, and CZMV-J, for the second example and $k \in [0, 200]$.

8.4 Final remarks

This chapter developed a new method for set-based joint state and parameter estimation of nonlinear discrete-time systems with algebraic variables and unknown model parameters. By extending the nonlinear state estimation methods using constrained zonotopes to include parameter estimation in a unified framework, and therefore maintaining existing mutual

dependencies between states, algebraic variables, and unknown parameters, the accuracy of both state and parameter estimation was significantly improved, as corroborated by two numerical examples. The next chapters present a few practical applications of the set-based state estimation and fault diagnosis methods developed in this thesis, specifically: unmanned aerial vehicles (Chapter 9), water distribution networks (Chapter 10), and Lithium-ion batteries (Chapter 11).

9

Application: Unmanned aerial vehicles

This chapter presents the application of set-based state estimation and fault diagnosis methods using constrained zonotopes to unmanned aerial vehicles. It is organized according to the following topics: (i) linear state estimation applied to aerial load transportation; (ii) nonlinear state estimation applied to a quadrotor UAV considering sensors located at the UAV; and (iii) active fault diagnosis of a quadrotor UAV subject to actuator and sensor faults.

9.1 Linear state estimation applied to aerial load transportation

The problem of slung load transportation arises in a variety of essential tasks, such as transportation of containers in harbors ([Ngo & Hong, 2012](#)), aerial delivery of supplies in search-and-rescue missions ([Bernard et al., 2011](#)), and landmine detection ([Bisgaard, 2008](#)). The suspended load is usually connected to the mobile platform by means of a rope, considerably changing its dynamic behavior and adding unactuated degrees of freedom to the whole system. Moreover, the rope is a non-rigid body and is not always taut, which increases the task challenge. Several studies can be found in the literature, concerning different modeling approaches and control strategies for load transportation using overhead cranes ([Wu et al., 2015](#)), robotic manipulators ([Chen et al., 2007](#)), and aerial vehicles ([Bisgaard et al., 2009](#)).

Unmanned aerial vehicles have been extensively used for load transportation, thanks to their versatility and autonomous operation (Sreenath et al., 2013). Control objectives include path tracking of the UAV with load swing attenuation (Raffo & Almeida, 2016), and path tracking of the suspended load (Sreenath et al., 2013). The latter is the appropriate goal for tasks requiring precise maneuvering of the load, for which the knowledge of the load position is of utmost importance.

As mentioned above, a recurrent issue in aerial load transportation is the necessity of knowing the load position to accomplish the task. Since available sensors are often embedded in the mobile platform, information on the load position may not be directly available. The problem of estimating the load position is commonly addressed through visual systems and Bayesian state estimators, such as the linear Kalman filter (Jain, 2015), and the unscented Kalman filter (UKF) (Bisgaard et al., 2007b,a, 2010). In particular, in Bisgaard et al. (2007a), algorithms based on the UKF are proposed for estimation of the full state vector of a helicopter with suspended load, with measurements provided by a Global Positioning System (GPS) equipment, a magnetometer, a camera, an IMU on the helicopter and another one on the load. However, such algorithms require knowledge on statistical properties of existing process and measurement disturbances, which may not be easily obtained. On the other hand, set-based estimation requires knowledge only on bounds of existing disturbances, and are guaranteed to include the system states consistent with available measurements.

Most of the unmanned aerial vehicles used in load transportation tasks are in helicopter and quadrotor configurations. These rotary-wing UAVs have vertical take-off and landing (VTOL) and hovering capabilities, and achieve high maneuverability in low velocities. However, due to their limited flight envelope, such UAVs are not appropriate for missions that require long distance traveling, such as deployment of supplies to risky zones. To overcome such constraint, recent researches are looking into the design of small-scale hybrid aircrafts, being the tilt-rotor configuration among the most popular ones (Amiri et al., 2011; Park et al., 2013; Cardoso et al., 2016; Santos et al., 2017a; Cardoso et al., 2021). Provided with both fixed and rotary wings, tilt-rotor UAVs achieve an enlarged flight envelope by switching between helicopter and airplane flight-modes through thrusters tilting. However, such advantages come with several design and control challenges, since these aircrafts are complex, underactuated mechanical systems with highly coupled dynamics. Additionally, when these UAVs are connected to a payload through a rope, the dynamic behavior of the system varies due to the load's swing, which can destabilize the whole system if it is not well attenuated.

The use of tilt-rotor UAVs for load transportation is recent. A three-level cascade strategy composed of feedback linearization controllers is proposed in Almeida & Raffo (2015), which considers load swing attenuation, but unrealistically assumes that all the states are precisely known. On the other hand, a model predictive control (MPC) strategy

is designed in Santos et al. (2017b) for path tracking of a tilt-rotor UAV with suspended load, in which the aircraft followed a desired trajectory, while the load remained stable. The strategy takes into account time-varying load mass and rope length, and estimates the load position and orientation by means of an unscented Kalman filter. However, the state estimation is not guaranteed, and nothing can be said about the transient response of the closed-loop system. The use of set-based state estimation based on zonotopes for aerial load transportation with a tilt-rotor UAV was first proposed in Rego & Raffo (2016, 2019). However, the latter is considerably conservative, and requires iterative procedures to take into account multiple measurements.

This section solves the path tracking problem of a suspended load using a tilt-rotor UAV, with states obtained through set-based state estimation using constrained zonotopes. The realistic scenario in which the load position is not measured is taken into account, considering sensors with different sampling times, and unknown-but-bounded disturbances. In order to provide feedback to the controller based on the estimated state set, which is a constrained zonotope, a novel optimal state choice is proposed taking into consideration a *constrained minimum-variance criteria*, resulting in the main contribution of this section. Path tracking control for the proposed aerial load transportation task is then designed by a robust discrete-time mixed $\mathcal{H}_2/\mathcal{H}_\infty$ state-feedback approach. The estimation and control strategies are validated through numerical experiments, performed in a platform based on the Gazebo simulator with a Computer Aided Design (CAD) model of the system¹ (Lara et al., 2017), to corroborate the performance of the proposed method. This section improves the work published in Rego & Raffo (2019) by: (i) the design of a far more accurate set-based state estimator; (ii) a novel criteria for improved feedback connection based on constrained minimum-variance; (iii) improved control design taking into account the time-varying nature of the trajectory; and (iv) rigorous validation of the proposed strategies considering the CAD model of the system. The content of this section was published in Rego et al. (2018a), which is a continuity of the previous work published in Rego & Raffo (2019).

9.1.1 Tilt-rotor UAV with suspended load modeling

9.1.1.1 Equations of motion

This section describes the tilt-rotor UAV with suspended load (Figure 9.1) as a multi-body mechanical system composed of: (i) the aircraft's main structure (including batteries, landing skids, electronics and instrumentation); (ii) the right thruster group (right thruster and revolute joint); (iii) the left thruster group (left thruster and revolute joint); and (iv) the load group (load and rope). For control purposes, the rope is assumed to be rigid and massless.

¹See <https://github.com/Guiraffo/ProVANT-Simulator>.

The equations of motion of the entire system are formulated from the perspective of the load, by considering the load as a free rigid body with the tilt-rotor UAV rigidly attached to it. This choice of perspective allows state-feedback control strategies to directly steer the trajectory of the load (Rego & Raffo, 2019). Nine reference frames are defined (Figure 9.1): (i) the inertial reference frame, \mathcal{I} ; (ii) the load center of mass frame, \mathcal{L} ; (iii) the aircraft's geometric center frame, \mathcal{B} ; (iv) the main body center of mass frame, \mathcal{C}_1 ; (v) the right thruster center of mass frame, \mathcal{C}_2 ; (vi) the left thruster center of mass frame, \mathcal{C}_3 ; (vii) the rope's point of connection frame, \mathcal{A}_1 ; (viii) the right tilting axis frame, \mathcal{A}_2 ; and (ix) the left tilting axis frame, \mathcal{A}_3 .

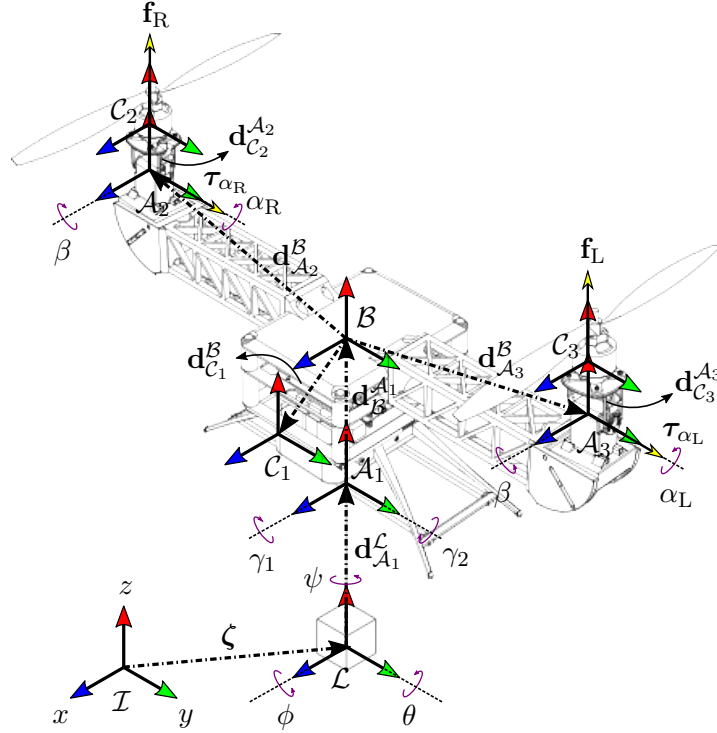


Figure 9.1: The tilt-rotor UAV with suspended load, kinematic definitions, input forces and torques.

The position of the load with respect to \mathcal{I} is denoted by $\zeta \triangleq [x \ y \ z]^T$. The displacement from \mathcal{L} to \mathcal{A}_1 is defined in \mathcal{L} as $\mathbf{d}_{\mathcal{A}_1}^{\mathcal{L}} \triangleq [0 \ 0 \ l]^T$, where l corresponds to the rope length. The displacements from \mathcal{A}_1 to \mathcal{B} , from \mathcal{B} to \mathcal{C}_1 , from \mathcal{B} to \mathcal{A}_i , and from \mathcal{A}_i to \mathcal{C}_i are denoted by $\mathbf{d}_{\mathcal{B}}^{\mathcal{A}_1}$, $\mathbf{d}_{\mathcal{C}_1}^{\mathcal{B}}$, $\mathbf{d}_{\mathcal{A}_i}^{\mathcal{B}}$, $\mathbf{d}_{\mathcal{C}_i}^{\mathcal{A}_i}$, respectively, in the respective preceding frames, with $i \in \{2, 3\}$. The orientation of the load with respect to \mathcal{I} is parametrized by Euler angles, $\boldsymbol{\eta} \triangleq [\phi \ \theta \ \psi]^T$, using the ZYX convention about local axes, and is described by

$$\mathbf{R}_{\mathcal{L}}^{\mathcal{I}} \triangleq \mathbf{R}_{z, \psi} \mathbf{R}_{y, \theta} \mathbf{R}_{x, \phi} = \begin{bmatrix} c_{\psi} c_{\theta} & c_{\psi} s_{\theta} s_{\phi} - s_{\psi} c_{\phi} & c_{\psi} s_{\theta} c_{\phi} + s_{\psi} s_{\phi} \\ s_{\psi} c_{\theta} & s_{\psi} s_{\theta} s_{\phi} + c_{\psi} c_{\phi} & s_{\psi} s_{\theta} c_{\phi} - c_{\psi} s_{\phi} \\ -s_{\theta} & c_{\theta} s_{\phi} & c_{\theta} c_{\phi} \end{bmatrix}. \quad (9.1)$$

with $s_{(\cdot)} \triangleq \sin(\cdot)$ and $c_{(\cdot)} \triangleq \cos(\cdot)$. The orientation of the UAV with respect to \mathcal{L} is described

by $\boldsymbol{\gamma} \triangleq [\gamma_1 \ \gamma_2]^T$, with $\mathbf{R}_{\mathcal{A}_1}^{\mathcal{C}} \triangleq \mathbf{R}_{x,-\gamma_1} \mathbf{R}_{y,-\gamma_2}$. The orientation of the thrusters' groups with respect to \mathcal{B} are defined by $\mathbf{R}_{\mathcal{A}_2}^{\mathcal{B}} \triangleq \mathbf{R}_{x,-\beta} \mathbf{R}_{y,\alpha_R}$ and $\mathbf{R}_{\mathcal{A}_3}^{\mathcal{B}} \triangleq \mathbf{R}_{x,\beta} \mathbf{R}_{y,\alpha_L}$, where α_R and α_L are the right and left tilting angles, respectively, and β is a fixed inclination angle of the thrusters towards \mathcal{B} (Rego & Raffo, 2019). Moreover, \mathcal{A}_1 , \mathcal{B} , and \mathcal{C}_1 are parallel and attached to the same rigid body, thus $\mathbf{R}_{\mathcal{B}}^{\mathcal{A}_1} = \mathbf{R}_{\mathcal{C}_1}^{\mathcal{B}} = \mathbf{R}_{\mathcal{C}_i}^{\mathcal{A}_i} = \mathbf{I}_3$, $\mathbf{R}_{\mathcal{B}}^{\mathcal{C}} = \mathbf{R}_{\mathcal{A}_1}^{\mathcal{C}}$, $\mathbf{R}_{\mathcal{C}_i}^{\mathcal{B}} = \mathbf{R}_{\mathcal{A}_i}^{\mathcal{B}}$, $i \in \{2, 3\}$.

Let $\boldsymbol{\omega}_{\mathcal{I}\mathcal{L}}^{\mathcal{C}}$ denote the angular velocity of frame \mathcal{L} with respect to \mathcal{I} , expressed in \mathcal{L} . Then, $\boldsymbol{\omega}_{\mathcal{I}\mathcal{L}}^{\mathcal{C}} = \mathbf{W}_{\boldsymbol{\eta}} \dot{\boldsymbol{\eta}}$, $\boldsymbol{\omega}_{\mathcal{L}\mathcal{A}_1}^{\mathcal{A}_1} = \mathbf{Q} \dot{\boldsymbol{\gamma}}$, $\boldsymbol{\omega}_{\mathcal{B}\mathcal{A}_2}^{\mathcal{A}_2} = \mathbf{a}_y \dot{\alpha}_R$, $\boldsymbol{\omega}_{\mathcal{B}\mathcal{A}_3}^{\mathcal{A}_3} = \mathbf{a}_y \dot{\alpha}_L$, $\boldsymbol{\omega}_{\mathcal{A}_1\mathcal{B}}^{\mathcal{B}} = \boldsymbol{\omega}_{\mathcal{B}\mathcal{C}_1}^{\mathcal{C}_1} = \boldsymbol{\omega}_{\mathcal{A}_i\mathcal{C}_i}^{\mathcal{C}_i} = \mathbf{0}_{3 \times 1}$ ², $\boldsymbol{\omega}_{\mathcal{L}\mathcal{B}}^{\mathcal{B}} = \boldsymbol{\omega}_{\mathcal{L}\mathcal{A}_1}^{\mathcal{A}_1}$, and $\boldsymbol{\omega}_{\mathcal{B}\mathcal{C}_i}^{\mathcal{C}_i} = \boldsymbol{\omega}_{\mathcal{B}\mathcal{A}_i}^{\mathcal{A}_i}$, with

$$\mathbf{W}_{\boldsymbol{\eta}} \triangleq \begin{bmatrix} 1 & 0 & -s_{\theta} \\ 0 & c_{\phi} & s_{\phi} c_{\theta} \\ 0 & -s_{\phi} & c_{\phi} c_{\theta} \end{bmatrix}, \quad \mathbf{Q} \triangleq \begin{bmatrix} -c_{\gamma_2} & 0 \\ 0 & -1 \\ s_{\gamma_2} & 0 \end{bmatrix}, \quad \mathbf{a}_y \triangleq \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}.$$

The generalized coordinates are defined according to $\mathbf{q} \triangleq [\boldsymbol{\zeta}^T \ \boldsymbol{\eta}^T \ \boldsymbol{\gamma}^T \ \alpha_R \ \alpha_L]^T$, which include the load position and orientation given in the inertial frame and reflect the choice of perspective. Therefore, through the Euler-Lagrange formulation (Rego & Raffo, 2019), the equations of motion are written as

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \boldsymbol{\vartheta}(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{u}, \mathbf{d}), \quad (9.2)$$

where $\mathbf{M}(\mathbf{q})$ is the inertia matrix, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ is the Coriolis and centripetal forces matrix, $\mathbf{g}(\mathbf{q})$ is the gravitational forces vector, and $\boldsymbol{\vartheta}(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{u}, \mathbf{d}) = \mathbf{L}_{\text{in}}(\mathbf{q})\mathbf{u} - \mathbf{L}_{\text{fr}}\dot{\mathbf{q}} + \mathbf{L}_{\text{db}}\mathbf{d}$, with $\mathbf{u} = [f_R \ f_L \ \tau_{\alpha_R} \ \tau_{\alpha_L}]^T \in \mathbb{R}^4$ denoting the system inputs (thrust forces and servomotor torques, see Figure 9.1), $\mathbf{d} \in \mathbb{R}^3$ denoting external disturbances affecting the load, $\mathbf{L}_{\text{in}}(\mathbf{q})$ is the input coupling matrix, $\mathbf{L}_{\text{fr}}(\mathbf{q})$ is the matrix of viscous friction coefficients, and $\mathbf{L}_{\text{d}}(\mathbf{q})$ is the disturbance mapping matrix (see Rego & Raffo (2019) for details).

Finally, by defining the system states $\mathbf{x} \triangleq [\mathbf{q}^T \ \dot{\mathbf{q}}^T]^T \in \mathbb{R}^{20}$, the equations (9.2) are written as $\dot{\mathbf{x}} = \boldsymbol{\varphi}(\mathbf{x}, \mathbf{u}, \mathbf{d})$,

$$\boldsymbol{\varphi}(\mathbf{x}, \mathbf{u}, \mathbf{d}) = \begin{bmatrix} \dot{\mathbf{q}} \\ \mathbf{M}^{-1} [-(\mathbf{C} + \mathbf{L}_{\text{fr}})\dot{\mathbf{q}} - \mathbf{g}(\mathbf{q}) + \mathbf{L}_{\text{in}}(\mathbf{q})\mathbf{u} + \mathbf{L}_{\text{db}}\mathbf{d}] \end{bmatrix}, \quad (9.3)$$

which describes the load behavior in the inertial frame explicitly, with the aircraft position and orientation being described only with respect to the load.

9.1.1.2 Measurement equation

This doctoral thesis assumes that the load position and orientation (and corresponding velocities) with respect to the inertial frame are not measured. Available measurements are provided by: (i) a Global Positioning System (GPS), providing the planar position of

² \mathbf{I}_n is an identity matrix with dimension n , and $\mathbf{0}_{n \times m}$ denotes an n by m matrix of zeros.

the tilt-rotor UAV with respect to \mathcal{I} ; (ii) a barometer, providing the altitude of the UAV with respect to \mathcal{I} ; (iii) an Inertial Measurement Unit (IMU), providing the orientation and angular velocities of the UAV with respect to \mathcal{I} ; (iv) a camera, providing the position of the load with respect to the \mathcal{A}_1 ; and (v) embedded sensors at the servos, providing the tilting angles and their time derivatives.

The GPS, barometer, and IMU are located at \mathcal{B} , while the camera is placed at the origin of \mathcal{A}_1 . Let $\boldsymbol{\zeta}_{\mathcal{B}} \triangleq [x_{\mathcal{B}} \ y_{\mathcal{B}} \ z_{\mathcal{B}}]^T$ denote the position of the UAV with respect to \mathcal{I} . Then,

$$\boldsymbol{\zeta}_{\mathcal{B}}(\boldsymbol{\zeta}, \boldsymbol{\eta}) = \boldsymbol{\zeta} + \mathbf{R}_{\mathcal{L}}^{\mathcal{I}} \mathbf{d}_{\mathcal{A}_1}^{\mathcal{L}} + \mathbf{R}_{\mathcal{L}}^{\mathcal{I}} \mathbf{R}_{\mathcal{B}}^{\mathcal{L}} \mathbf{d}_{\mathcal{B}}^{\mathcal{A}_1}. \quad (9.4)$$

The orientation of the UAV with respect to \mathcal{I} is parametrized by $\boldsymbol{\eta}_{\mathcal{B}} \triangleq [\phi_{\mathcal{B}} \ \theta_{\mathcal{B}} \ \psi_{\mathcal{B}}]^T$, using the local roll-pitch-yaw convention. Therefore, $\mathbf{R}_{\mathcal{B}}^{\mathcal{I}} \triangleq \mathbf{R}_{z, \psi_{\mathcal{B}}} \mathbf{R}_{y, \theta_{\mathcal{B}}} \mathbf{R}_{x, \phi_{\mathcal{B}}} = \mathbf{R}_{\mathcal{L}}^{\mathcal{I}} \mathbf{R}_{\mathcal{B}}^{\mathcal{L}}$, leading to

$$\phi_{\mathcal{B}}(\boldsymbol{\eta}, \boldsymbol{\gamma}) = \arctan((\mathbf{R}_{\mathcal{L}}^{\mathcal{I}} \mathbf{R}_{\mathcal{B}}^{\mathcal{L}})_{32} / (\mathbf{R}_{\mathcal{L}}^{\mathcal{I}} \mathbf{R}_{\mathcal{B}}^{\mathcal{L}})_{33}), \quad (9.5)$$

$$\theta_{\mathcal{B}}(\boldsymbol{\eta}, \boldsymbol{\gamma}) = \arcsin(-(\mathbf{R}_{\mathcal{L}}^{\mathcal{I}} \mathbf{R}_{\mathcal{B}}^{\mathcal{L}})_{31}), \quad (9.6)$$

$$\psi_{\mathcal{B}}(\boldsymbol{\eta}, \boldsymbol{\gamma}) = \arctan((\mathbf{R}_{\mathcal{L}}^{\mathcal{I}} \mathbf{R}_{\mathcal{B}}^{\mathcal{L}})_{21} / (\mathbf{R}_{\mathcal{L}}^{\mathcal{I}} \mathbf{R}_{\mathcal{B}}^{\mathcal{L}})_{11}), \quad (9.7)$$

for $\theta_{\mathcal{B}} \neq \pm\pi/2$, where $(\cdot)_{ij}$ denotes the element from the i -th row and j -th column. Moreover, the angular velocity provided by the IMU is given by

$$\boldsymbol{\omega}_{\mathcal{I}\mathcal{B}}^{\mathcal{B}}(\boldsymbol{\eta}, \boldsymbol{\gamma}, \dot{\boldsymbol{\eta}}, \dot{\boldsymbol{\gamma}}) = \boldsymbol{\omega}_{\mathcal{I}\mathcal{L}}^{\mathcal{B}} + \boldsymbol{\omega}_{\mathcal{L}\mathcal{B}}^{\mathcal{B}} = (\mathbf{R}_{\mathcal{B}}^{\mathcal{I}})^T \mathbf{W}_{\boldsymbol{\eta}} \dot{\boldsymbol{\eta}} + \mathbf{Q} \dot{\boldsymbol{\gamma}}. \quad (9.8)$$

On the other hand, let $\mathbf{d}_{\mathcal{A}_1\mathcal{L}}^{\mathcal{A}_1}$ denote the displacement from \mathcal{A}_1 to \mathcal{L} , expressed in \mathcal{A}_1 , provided by the camera. Then,

$$\mathbf{d}_{\mathcal{A}_1\mathcal{L}}^{\mathcal{A}_1}(\boldsymbol{\gamma}) = -(\mathbf{R}_{\mathcal{A}_1}^{\mathcal{L}})^T \mathbf{d}_{\mathcal{A}_1}^{\mathcal{L}} = -(\mathbf{R}_{\mathcal{B}}^{\mathcal{L}})^T \mathbf{d}_{\mathcal{A}_1}^{\mathcal{L}}. \quad (9.9)$$

Putting together (9.4) through (9.9), and considering $\alpha_{\mathcal{R}}$, $\alpha_{\mathcal{L}}$, $\dot{\alpha}_{\mathcal{R}}$, and $\dot{\alpha}_{\mathcal{L}}$, provided by sensors at the servos, lead to the nonlinear measurement equation $\mathbf{y}_k = \boldsymbol{\pi}(\mathbf{x}_k) + \mathbf{v}_k$, where \mathbf{v}_k denotes measurement noise, and

$$\boldsymbol{\pi}(\mathbf{x}_k) = \left[\boldsymbol{\zeta}_{\mathcal{B}}^T \ \boldsymbol{\eta}_{\mathcal{B}}^T \ (\boldsymbol{\omega}_{\mathcal{I}\mathcal{B}}^{\mathcal{B}})^T \ (\mathbf{d}_{\mathcal{A}_1\mathcal{L}}^{\mathcal{A}_1})^T \ \alpha_{\mathcal{R}} \ \alpha_{\mathcal{L}} \ \dot{\alpha}_{\mathcal{R}} \ \dot{\alpha}_{\mathcal{L}} \right]^T.$$

Finally, let \mathbb{I}_k denote the set of available measurements at time instant k , such that $\mathbb{I}_k \subseteq \{1, 2, \dots, 16\}$, and ι_k denote the number of elements in \mathbb{I}_k . Then, $\mathbf{y}_k, \mathbf{v}_k \in \mathbb{R}^{\iota_k}$, and

$$\boldsymbol{\pi}^{[k]}(\mathbf{x}_k) \triangleq [\boldsymbol{\pi}(\mathbf{x}_k)(i)]_{i \in \mathbb{I}_k}, \quad (9.10)$$

where $\boldsymbol{\pi}(\mathbf{x}_k)(i)$ denotes the i -th row of $\boldsymbol{\pi}(\mathbf{x}_k)$, and the brackets denote vertical concatenation.

9.1.1.3 Linearized model for state estimation

Since the method proposed in this section is formulated for linear systems, the state-space equations (9.3) and measurement mapping (9.10) are linearized around equilibrium values³ \mathbf{x}^{eq} , \mathbf{u}^{eq} , $\mathbf{d} = \mathbf{0}$, through first-order Taylor series expansion, and discretized by the zero-order-hold method, yielding to the disturbed switching linear discrete-time system

$$\begin{aligned}\Delta \mathbf{x}_k &= \mathbf{A} \Delta \mathbf{x}_{k-1} + \mathbf{B} \Delta \mathbf{u}_{k-1} + \mathbf{F} \mathbf{d}_{k-1} + \mathbf{w}_{k-1}, \\ \mathbf{y}_k &= \mathbf{H}^{[k]} \Delta \mathbf{x}_k + \boldsymbol{\pi}^{[k]}(\mathbf{x}^{\text{eq}}) + \mathbf{v}_k,\end{aligned}\tag{9.11}$$

where the process noise \mathbf{w}_k corresponds to unmodeled dynamics due to linearization and discretization, and the measurement noise \mathbf{v}_k now includes the nonlinearities of (9.10). Moreover, $\Delta \mathbf{x}_k \triangleq \mathbf{x}_k - \mathbf{x}^{\text{eq}}$ and $\Delta \mathbf{u}_k \triangleq \mathbf{u}_k - \mathbf{u}^{\text{eq}}$, where $\mathbf{A} \in \mathbb{R}^{20 \times 20}$, $\mathbf{B} \in \mathbb{R}^{20 \times 4}$, $\mathbf{F} \in \mathbb{R}^{20 \times 3}$, and $\mathbf{H}^{[k]} \in \mathbb{R}^{l_k \times 20}$ are the respective Jacobian matrices evaluated at the equilibrium values followed by the discretization procedure.

For robust state estimation, the state vector is augmented with the external disturbances \mathbf{d}_k , yielding the augmented dynamics

$$\begin{aligned}\begin{bmatrix} \Delta \mathbf{x}_k \\ \mathbf{d}_k \end{bmatrix} &= \begin{bmatrix} \mathbf{A} & \mathbf{F} \\ \mathbf{0}_{3 \times 20} & \mathbf{I}_3 \end{bmatrix} \begin{bmatrix} \Delta \mathbf{x}_{k-1} \\ \mathbf{d}_{k-1} \end{bmatrix} + \begin{bmatrix} \mathbf{B} \\ \mathbf{0}_{3 \times 4} \end{bmatrix} \Delta \mathbf{u}_{k-1} + \begin{bmatrix} \mathbf{w}_{k-1} \\ \tilde{\mathbf{d}}_{k-1} \end{bmatrix}, \\ \mathbf{y}_k &= \begin{bmatrix} \mathbf{H}^{[k]} & \mathbf{0}_{l_k \times 3} \end{bmatrix} \begin{bmatrix} \Delta \mathbf{x}_k \\ \mathbf{d}_k \end{bmatrix} + \boldsymbol{\pi}^{[k]}(\mathbf{x}^{\text{eq}}) + \mathbf{v}_k,\end{aligned}\tag{9.12}$$

with $\tilde{\mathbf{d}}_{k-1} \triangleq \mathbf{d}_k - \mathbf{d}_{k-1}$.

Accordingly, define $\boldsymbol{\nu}_k \triangleq [\Delta \mathbf{x}_k^T \ \mathbf{d}_k^T]^T$, $\bar{\mathbf{w}}_k \triangleq [\mathbf{w}_k^T \ \tilde{\mathbf{d}}_k^T]^T$, and $\bar{\mathbf{v}}_k \triangleq \boldsymbol{\pi}^{[k]}(\mathbf{x}^{\text{eq}}) + \mathbf{v}_k$. Thus, the linear set-based state estimation method is formulated in the next subsection.

9.1.2 Problem formulation

As previously commented in Chapter 3, constrained zonotopes were effectively used in Scott et al. (2016) for set-based state estimation of linear systems, using the method illustrated in Section 3.5. This chapter extends this estimation method for discrete-time switching systems. Consider the augmented linear discrete-time switching system described by (9.12), and let \mathbf{A}_ν , \mathbf{B}_ν and $\mathbf{H}_\nu^{[k]}$ denote the associated matrices. Assuming bounded uncertainties $\bar{\mathbf{w}}_k \in W_k$ and $\bar{\mathbf{v}}_k \in V_k$, and $\boldsymbol{\nu}_0 \in X_0$ ⁴, where X_0 , W_k and V_k are known compact sets, linear set-based state estimation consists in computing enclosures \bar{X}_k and \hat{X}_k such

³Due to limited computational resources, analytical expressions for the Jacobian and Hessians of the nonlinear mapping $\boldsymbol{\varphi}(\mathbf{x}, \mathbf{u}, \mathbf{d})$ could not be obtained. Therefore, we resort to linear state estimation for aerial load transportation using the tilt-rotor UAV. Moreover, equilibrium values are used since online evaluation of a time-varying model around trajectory values has demonstrated to be intractable.

⁴Note that $X_0 \subset \mathbb{R}^{23}$, since this set is associated to the augmented state variables $\boldsymbol{\nu}_k$.

that

$$\bar{X}_k \supseteq \{\mathbf{A}_\nu \boldsymbol{\nu} + \mathbf{B}_\nu \Delta \mathbf{u}_{k-1} + \bar{\mathbf{w}} : \boldsymbol{\nu} \in \hat{X}_{k-1}, \bar{\mathbf{w}} \in W_{k-1}\}, \quad (9.13)$$

$$\hat{X}_k \supseteq \{\boldsymbol{\nu} \in \bar{X}_k : \mathbf{H}_\nu^{[k]} \boldsymbol{\nu} + \bar{\mathbf{v}} = \mathbf{y}_k, \bar{\mathbf{v}} \in V_k\}, \quad (9.14)$$

with (9.13) referred to as *prediction step*, and (9.14) as *update step*. These enclosures can be obtained exactly with constrained zonotopes through

$$\bar{X}_k = \mathbf{A}_\nu \hat{X}_{k-1} \oplus \mathbf{B}_\nu \Delta \mathbf{u}_{k-1} \oplus W_{k-1}, \quad (9.15)$$

$$\hat{X}_k = \bar{X}_k \cap_{\mathbf{H}_\nu^{[k]}} (\mathbf{y}_k \oplus (-V_k)), \quad (9.16)$$

using (3.8)–(3.10).

As discussed in Chapter 3, the right-hand side of (9.15) cannot be computed using interval arithmetic without wrapping. On the other hand, the equality in (9.15) can be achieved using zonotopes and constrained zonotopes, but the intersection in (9.16) can not be computed exactly using zonotopes. As a consequence, the enclosures of the system states obtained after many iterations of prediction and update may be quite conservative using zonotopes. Nevertheless, with constrained zonotopes both (9.15) and (9.16) are easily computed using (3.8)–(3.10). Moreover, iterated computations of (9.15) and (9.16) result in only a linear increase in the complexity of the CG-rep (3.7). This can be effectively addressed using the complexity reduction methods presented in Section 3.3.

9.1.3 Minimum-variance feedback

Despite the accuracy and low computational burden of the method described by (9.15)–(9.16), to perform state-feedback control with estimated states one must choose a point satisfying $\hat{\boldsymbol{\nu}}_k \in \hat{X}_k$. Therefore, we propose an optimal choice according to a *minimum-variance criterion*, which is one of the main contributions of this chapter, and can be straightforwardly adapted to more general classes of linear discrete-time switching dynamical systems.

The proposed criterion is based on the well-known Kalman filter algorithm (Simon, 2006). Let \mathbf{P}_k^ν , \mathbf{P}_k^w , and \mathbf{P}_k^v , denote the covariance matrices of the estimation error $\boldsymbol{\nu}_k - \hat{\boldsymbol{\nu}}_k$, process noise $\bar{\mathbf{w}}_k$, and measurement noise $\bar{\mathbf{v}}_k$, respectively. Consider the constrained optimization problem

$$\min_{\mathbf{N}_k} \text{trace}(\mathbf{P}_k^\nu) \quad \text{s.t.} \quad \hat{\boldsymbol{\nu}}_k \in \hat{X}_k, \quad (9.17)$$

where $\hat{\boldsymbol{\nu}}_k$ is given according to the correction step

$$\hat{\boldsymbol{\nu}}_k = \bar{\boldsymbol{\nu}}_k + \mathbf{N}_k (\mathbf{y}_k - (\mathbf{H}_\nu^{[k]} \bar{\boldsymbol{\nu}}_k + \boldsymbol{\pi}^{[k]}(\mathbf{x}^{\text{eq}}))), \quad (9.18)$$

with $\bar{\mathbf{v}}_k = \mathbf{A}_\nu \hat{\mathbf{v}}_{k-1} + \mathbf{B}_\nu \Delta \mathbf{u}_{k-1}$, and \mathbf{N}_k is the Kalman gain. Moreover,

$$\mathbf{P}_k^\nu = (\mathbf{I}_{23} - \mathbf{N}_k \mathbf{H}_\nu^{[k]}) \bar{\mathbf{P}}_k^\nu (\mathbf{I}_{23} - \mathbf{N}_k \mathbf{H}_\nu^{[k]})^T + \mathbf{N}_k \mathbf{P}_k^\nu \mathbf{N}_k^T, \quad (9.19)$$

where $\bar{\mathbf{P}}_k^\nu = \mathbf{A}_\nu \mathbf{P}_{k-1}^\nu \mathbf{A}_\nu^T + \mathbf{P}_k^w$. Considering Property 3.1 and $\hat{X}_k = \{\mathbf{G}_x, \mathbf{c}_x, \mathbf{A}_x, \mathbf{b}_x\}$, the optimization problem (9.17) is rewritten as a quadratic program with linear constraints,

$$\begin{aligned} \min_{\mathbf{N}_k, \boldsymbol{\xi}} \text{trace}(\mathbf{P}_k^\nu) & \quad (9.20) \\ \text{s.t. } \bar{\mathbf{v}}_k + \mathbf{N}_k (\mathbf{y}_k - (\mathbf{H}_\nu^{[k]} \bar{\mathbf{v}}_k + \boldsymbol{\pi}^{[k]}(\mathbf{x}^{\text{eq}}))) &= \mathbf{c}_x + \mathbf{G}_x \boldsymbol{\xi}, \\ \|\boldsymbol{\xi}\|_\infty \leq 1, \quad \mathbf{A}_x \boldsymbol{\xi} &= \mathbf{b}_x, \end{aligned}$$

with \mathbf{P}_k^ν given by (9.19). The minimum-variance choice $\hat{\mathbf{v}}_k$ is then obtained by solving (9.20), and computing (9.18) afterwards. The optimization problem (9.20) can be regarded as *constrained Kalman filtering*, in which divergence issues are not present since $\hat{\mathbf{v}}_k \in \hat{X}_k$ is guaranteed by construction.⁵

9.1.4 Path tracking control

9.1.4.1 Linearized parameter-varying error dynamics

The tilt-rotor UAV with suspended load is an underactuated mechanical system with four control inputs. For path tracking of the suspended load, the position $\boldsymbol{\zeta} = [x \ y \ z]^T$ and yaw angle ψ of the load are chosen to be regulated. Let $\mathbf{q}^{\text{tr}} = [\boldsymbol{\zeta}^{\text{tr}} \ \boldsymbol{\eta}^{\text{tr}} \ \gamma^{\text{eq}} \ \alpha_{\text{R}}^{\text{eq}} \ \alpha_{\text{L}}^{\text{eq}}]$, with $\boldsymbol{\eta}^{\text{tr}} = [\phi^{\text{eq}} \ \theta^{\text{eq}} \ \psi^{\text{tr}}]$, where ‘tr’ denotes trajectory values. Assuming ψ^{tr} constant, the linearized error dynamics of (9.3) are given by $\delta \dot{\mathbf{x}} = \mathbf{A}_c(t) \delta \mathbf{x} + \mathbf{B}_c \delta \mathbf{u} + \mathbf{F}_c \mathbf{d}$, where $\delta \mathbf{x} \triangleq \mathbf{x} - \mathbf{x}^{\text{tr}}$, $\delta \mathbf{u} \triangleq \mathbf{u} - \mathbf{u}^{\text{tr}}$, $\mathbf{x}^{\text{tr}} \triangleq [(\mathbf{q}^{\text{tr}})^T \ (\dot{\mathbf{q}}^{\text{tr}})^T]^T$, and $\mathbf{A}_c(t)$ is a function of \mathbf{u}^{tr} (Rego & Raffo, 2019).

Considering a feasible reference trajectory, the desired input can be obtained from (9.3) as follows:

$$\mathbf{u}^{\text{tr}} = \mathbf{L}_{\text{in}}(\mathbf{q}^{\text{tr}})^+ [\mathbf{M}(\mathbf{q}^{\text{tr}}) \ddot{\mathbf{q}}^{\text{tr}} + (\mathbf{C}(\mathbf{q}^{\text{tr}}, \dot{\mathbf{q}}^{\text{tr}}) + \mathbf{L}_{\text{fr}}) \dot{\mathbf{q}}^{\text{tr}} + \mathbf{g}(\mathbf{q}^{\text{tr}})], \quad (9.21)$$

where $\mathbf{L}_{\text{in}}(\mathbf{q}^{\text{tr}})^+$ denotes the pseudo-inverse of $\mathbf{L}_{\text{in}}(\mathbf{q}^{\text{tr}})$. It can be shown that (9.21) is a function only of the accelerations $\ddot{\boldsymbol{\zeta}}^{\text{tr}}$ (Rego & Raffo, 2019). By defining the vector of parameters $\boldsymbol{\sigma} \triangleq \ddot{\boldsymbol{\zeta}}^{\text{tr}}$, we have that $\mathbf{u}^{\text{tr}}(t) \triangleq \mathbf{u}(\boldsymbol{\sigma})$, which is affine in $\boldsymbol{\sigma}$. Then, substituting (9.21) in $\mathbf{A}_c(t)$ yields the parameter-varying matrix $\mathbf{A}_c(\boldsymbol{\sigma})$, which is affine in $\boldsymbol{\sigma}$ since (9.3) is affine in \mathbf{u} .

For an improved trajectory tracking and rejection of constant disturbances, the state vector $\delta \mathbf{x}$ is augmented with integral actions over the regulated variables, as $\boldsymbol{\chi} \triangleq [\delta \mathbf{x}^T \ \int (\boldsymbol{\zeta} - \boldsymbol{\zeta}^{\text{tr}})^T \ \int (\psi - \psi^{\text{tr}})]^T$. The resulting augmented dynamics are then discretized through Euler

⁵Despite this guarantee, the stochastic properties of Gaussian distribution associated to $\hat{\mathbf{v}}_k$ has no connection with the set \hat{X}_k . This issue is a subject of future investigation.

approximation⁶, yielding the discrete-time, augmented LPV error dynamics

$$\boldsymbol{\chi}_{k+1} = \mathbf{A}_\chi(\boldsymbol{\sigma})\boldsymbol{\chi}_k + \mathbf{B}_\chi\delta\mathbf{u}_k + \mathbf{F}_\chi\mathbf{d}_k, \quad (9.22)$$

with $\mathbf{A}_\chi(\boldsymbol{\sigma}) \in \mathbb{R}^{24 \times 24}$, $\mathbf{B}_\chi \in \mathbb{R}^{24 \times 4}$ and $\mathbf{F}_\chi \in \mathbb{R}^{24 \times 3}$. Finally, assuming bounded trajectory accelerations, the LPV system (9.22) can be rewritten in a convex polytopic representation, by defining $\mathbf{A}_\chi(\boldsymbol{\sigma}) \triangleq \mathbf{A}_\chi(\tilde{\boldsymbol{\sigma}}) = \sum_{i=1}^8 \tilde{\sigma}_i \mathbf{A}_\chi^i$, with \mathbf{A}_χ^i denoting the i -th vertex of $\mathbf{A}_\chi(\tilde{\boldsymbol{\sigma}})$, and $\sum_{i=1}^8 \tilde{\sigma}_i = 1$. Resulting errors from linearization and discretization are taken into account in this section as unmodeled dynamics. Note that the LPV dynamics (9.22) describe the behavior of the tracking error $\delta\mathbf{x}$ augmented with integral actions, while the time-invariant linear dynamics (9.12) describe the evolution of the system states \mathbf{x} around equilibrium values augmented with the external disturbances.

9.1.4.2 Discrete-time mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control with pole placement constraints

The aim of the proposed controller is to perform robust trajectory tracking of the load with stabilization of the tilt-rotor UAV, with estimation provided by the novel method presented in Section 9.1.3. The desired accelerations for the load are required to be smooth and bounded. The present method extends the approach in Chilali & Gahinet (1996) for discrete-time linear systems. Let

$$\begin{aligned} \mathbf{z}_k^{(2)} &= \mathbf{H}_z\boldsymbol{\chi}_k + \mathbf{D}_{zu}\delta\mathbf{u}_k, \\ \mathbf{z}_k^{(\infty)} &= \mathbf{H}_z\boldsymbol{\chi}_k + \mathbf{D}_{zu}\delta\mathbf{u}_k + \mathbf{D}_{zd}\mathbf{d}_k, \end{aligned}$$

where $\mathbf{H}_z \in \mathbb{R}^{n_z \times 24}$, $\mathbf{D}_{zu} \in \mathbb{R}^{n_z \times 4}$, $\mathbf{D}_{zd} \in \mathbb{R}^{n_z \times 3}$. Let $\Psi_{dz}^{(2)}(\varsigma)$ and $\Psi_{dz}^{(\infty)}(\varsigma)$ denote the discrete-time transfer matrices from \mathbf{d}_k to $\mathbf{z}_k^{(2)}$ and \mathbf{d}_k to $\mathbf{z}_k^{(\infty)}$, and $\|\Psi_{dz}^{(2)}(\varsigma)\|_2$ and $\|\Psi_{dz}^{(\infty)}(\varsigma)\|_\infty$ be the corresponding \mathcal{H}_2 and \mathcal{H}_∞ norms, respectively. Given the state-feedback control law $\mathbf{u}_k = -\mathbf{K}\boldsymbol{\chi}$, the gain matrix \mathbf{K} that minimizes $\text{trace}(\boldsymbol{\Omega}) > \|\Psi_{dz}^{(2)}\|_2^2$ while guaranteeing an upper-bound $\tilde{\gamma} > \|\Psi_{dz}^{(\infty)}\|_\infty^2$ is given by $\mathbf{K} = -\mathbf{Y}\mathbf{X}^{-1}$, where \mathbf{Y} and \mathbf{X} are the solution of the optimization problem

$$\begin{aligned} \min_{\mathbf{P}, \mathbf{X}, \mathbf{Y}, \boldsymbol{\Omega}} \text{trace}(\boldsymbol{\Omega}) \quad \text{s.t.} \quad & (9.23) \\ \begin{bmatrix} \boldsymbol{\Omega} & \mathbf{H}_z\mathbf{X} + \mathbf{D}_{zu}\mathbf{Y} \\ * & \mathbf{X} + \mathbf{X}^T - \mathbf{P} \end{bmatrix} & > 0, \\ \begin{bmatrix} \mathbf{P} & \mathbf{A}_\chi^i\mathbf{X} + \mathbf{B}_\chi\mathbf{Y} & \mathbf{F}_\chi & \mathbf{0}_{24 \times n_z} \\ * & \mathbf{X} + \mathbf{X}^T - \mathbf{P} & \mathbf{0}_{24 \times 3} & \mathbf{X}^T\mathbf{H}_z^T + \mathbf{Y}^T\mathbf{D}_{zu}^T \\ * & * & \mathbf{I}_3 & \mathbf{D}_{zd}^T \\ * & * & * & \tilde{\gamma}\mathbf{I}_{n_z} \end{bmatrix} & > 0, \end{aligned}$$

⁶The Euler approximation is used here in order to obtain a discrete-time LPV system. On the other hand, the zero-order-hold method is used in (9.11) for reduced discretization error.

with $i = 1, 2, \dots, 8$, $\mathbf{P} = \mathbf{P}^T > 0$, $\mathbf{X} > 0$ and $\boldsymbol{\Omega} = \boldsymbol{\Omega}^T$.

To ensure transient properties for the closed-loop system, additional constraints are imposed to pole placement. Seeking minimum and maximum settling times, maximum percentage overshoot, and also to avoid ringing, three regions are of interest: $\mathbb{D}_1 \triangleq \{\varsigma \in \mathbb{C} : \text{Re}(\varsigma) > \varepsilon \geq 0\}$, $\mathbb{D}_2 \triangleq \{\varsigma \in \mathbb{C} : 0 \leq |\varsigma| < \varpi\}$, and $\mathbb{D}_3 \triangleq \{\varsigma \in \mathbb{C} : 0 \leq |\text{Im}(\varsigma)| < \tau\}$. The eigenvalues of $\mathbf{A}_x(\boldsymbol{\sigma}) - \mathbf{B}_x \mathbf{K}$ belong to $\bigcap_{j=1}^3 \mathbb{D}_j$ iff (Rego & Raffo, 2019)

$$\begin{aligned} & \mathbf{X}(\mathbf{A}_x^i)^T + \mathbf{A}_x^i \mathbf{X} + \mathbf{Y}^T \mathbf{B}_x^T + \mathbf{B}_x \mathbf{Y} - 2\varepsilon \mathbf{X} > 0, \\ & \begin{bmatrix} -\varpi \mathbf{X} & \mathbf{A}_x^i \mathbf{X} + \mathbf{B}_x \mathbf{Y} \\ * & -\varpi \mathbf{X} \end{bmatrix} < 0, \\ & \begin{bmatrix} -2\tau \mathbf{X} & \mathbf{X}(\mathbf{A}_x^i)^T - \mathbf{A}_x^i \mathbf{X} + \mathbf{Y}^T \mathbf{B}_x^T - \mathbf{B}_x \mathbf{Y} \\ * & -2\tau \mathbf{X} \end{bmatrix} < 0. \end{aligned}$$

Finally, the control signals applied to (9.3) are obtained from $\mathbf{u}_k = \delta \mathbf{u}_k + \mathbf{u}_k^{\text{tr}} = -\mathbf{K} \hat{\boldsymbol{\chi}}_k + \mathbf{u}_k^{\text{tr}}$, with \mathbf{u}_k^{tr} given by (9.21), where the estimated states $\hat{\boldsymbol{\chi}}_k$ are retrieved from $\hat{\boldsymbol{\nu}}_k$ according to the definitions of the state vectors $\boldsymbol{\nu}_k$ and $\boldsymbol{\chi}_k$.

9.1.5 Numerical experiments

This section evaluates the performance of the proposed method through numerical experiments.

The experiment is performed in the ProVANT simulator platform, using a Computer Aided Design (CAD) model of the system. The model parameters are shown in Table⁷ 9.1, where m_j is the mass of each rigid body, \mathbf{I}_j is the inertia tensor, $j \in \{\mathcal{L}, 1, 2, 3\}$, $\hat{\mathbf{g}}$ is the gravity acceleration vector in \mathcal{I} , k_τ and b are drag torque parameters obtained experimentally, $\lambda_R, \lambda_L \in \{-1, 1\}$ are given according to the direction of rotation of the propellers, μ_γ and μ_α are constant parameters associated with viscous friction (Rego & Raffo, 2019). These model parameters yield the equilibrium point

$$\begin{aligned} \mathbf{q}^{\text{eq}} &= [\mathbf{0}_{1 \times 6} \quad 0.000132 \quad 0.01396 \quad 0.01401 \quad 0.01381]^T, \\ \mathbf{u}^{\text{eq}} &= [11.7323 \quad 11.7676 \quad 4.1389 \cdot 10^{-7} \quad 1.0121 \cdot 10^{-5}]^T. \end{aligned}$$

The desired trajectory is composed of several paths, with $\psi^{\text{tr}} = 0$ rad. The sensor parameters are shown in Table 9.2.

The optimization problem (9.23) was solved using SDPT3, with $\varepsilon = 0.55$, $\varpi = 0.994$, $\tau = 0.3$, $\tilde{\gamma} = 81$, $\ddot{x}^{\text{tr}}(t) \in [-0.5, 0.5]$, $\ddot{y}^{\text{tr}}(t) \in [-0.5, 0.5]$, $\ddot{z}^{\text{tr}}(t) \in [-0.3, 0.3]$,

$$\mathbf{H}_z = \text{diag}\left(\frac{\sqrt{10}}{2} \mathbf{1}_{3 \times 1}, \frac{\sqrt{0.5}}{\pi/2} \mathbf{1}_{2 \times 1}, \frac{\sqrt{5}}{\pi}, \frac{2}{\pi} \mathbf{1}_{2 \times 1}, \frac{0.2}{\pi} \mathbf{1}_{2 \times 1}, \frac{1}{2} \mathbf{1}_{3 \times 1}, \frac{3}{\pi} \mathbf{1}_{2 \times 1}, \frac{4}{\pi}, \frac{\sqrt{5}}{3\pi} \mathbf{1}_{2 \times 1}, \frac{0.1}{3\pi} \mathbf{1}_{2 \times 1}, \sqrt{5} \cdot \mathbf{1}_{3 \times 1}, \sqrt{0.1}\right),$$

⁷Parameter uncertainties can be considered immediately through $\bar{\mathbf{w}}_k, \bar{\mathbf{v}}_k$.

Table 9.1: Model parameters of the system.

Parameter	Value
$(m_{\mathcal{L}}, m_1, m_2, m_3)$	$(0.5, 1.7068, 0.08978, 0.08978)$ kg
$(\mathbf{d}_{A_1}^{\mathcal{L}}, \mathbf{d}_B^{A_1})$	$([0 \ 0 \ 0.5]^T, [0 \ 0 \ 0.119]^T)$ m
$\mathbf{d}_{C_1}^B$	$[-0.004321 \ 0.000601 \ -0.045113]^T$ m
$\mathbf{d}_{A_2}^B$	$[0 \ -0.275433 \ 0.056262]^T$ m
$\mathbf{d}_{A_3}^B$	$[0 \ 0.275433 \ 0.056261]^T$ m
$(\mathbf{d}_{C_2}^{A_2}, \mathbf{d}_{C_3}^{A_3})$	$([0 \ 0 \ 0.05647]^T, [0 \ 0 \ 0.05648]^T)$ m
$\mathbf{I}_{\mathcal{L}}$	$8.333 \cdot 10^{-6} \cdot \mathbf{I}_3$ kg·m ²
\mathbf{I}_1	$\begin{bmatrix} 4047.04 & 0.8606 & 9.6577 \\ * & 881.62 & -0.8731 \\ * & * & 4173.18 \end{bmatrix} \cdot 10^{-6}$ kg·m ²
$\mathbf{I}_2, \mathbf{I}_3$	$\text{diag}(335.74, 335.74, 641.59) \cdot 10^{-6}$ kg·m ²
$\hat{\mathbf{g}}$	$[0 \ 0 \ -9.81]^T$ m/s ²
(k_{τ}, b)	$(1.7 \cdot 10^{-7} \text{ N} \cdot \text{m} \cdot \text{s}^2, 9.5 \cdot 10^{-6} \text{ N} \cdot \text{s}^2)$
$(\lambda_R, \lambda_L, \beta)$	$(1, -1, 5^\circ)$
$\mu_{\gamma}, \mu_{\alpha}$	$0.005 \text{ N} \cdot \text{m} / (\text{rad/s})$

Table 9.2: Parameters of the sensors.

Sensor	\mathbb{I}	Noise bound	Sampling	PDF
GPS	$\{1, 2\}$	± 0.15 m	120 ms	Gauss.
Barometer	$\{3\}$	± 0.51 m	12 ms	Gauss.
IMU	$\{4, 5, 6\}$	$\pm 2.618 \cdot 10^{-3}$ rad	12 ms	Gauss.
	$\{7, 8, 9\}$	$\pm 16.558 \cdot 10^{-3}$ rad/s		
Camera	$\{10, 11\}$	± 0.005 m	24 ms	Unif.
	$\{12\}$	± 0.02 m		
Servos	$\{13, 14\}$	$\pm 5.67 \cdot 10^{-3}$ rad	12 ms	Unif.
	$\{15, 16\}$	± 0.50772 rad/s		

$$\mathbf{D}_{\mathbf{z}\mathbf{u}} = \begin{bmatrix} \text{diag} \left(\frac{\sqrt{750}}{30-f_R^{\text{eq}}}, \frac{\sqrt{750}}{30-f_L^{\text{eq}}}, \frac{\sqrt{5000}}{2-\tau_{\alpha_R}} \right) & \mathbf{0}_{3 \times 1} \\ \mathbf{0}_{2 \times 3} & \mathbf{0}_{2 \times 1} \\ \mathbf{0}_{1 \times 3} & \frac{\sqrt{5000}}{2-\tau_{\alpha_L}} \\ \mathbf{0}_{18 \times 3} & \mathbf{0}_{18 \times 1} \end{bmatrix},$$

$$\mathbf{D}_{\mathbf{z}\mathbf{d}} = [\mathbf{0}_{3 \times 10} \ \mathbf{I}_3 \ \mathbf{V}^T \ 0.5 \cdot \mathbf{1}_{3 \times 1} \ \mathbf{V}^T \ \mathbf{0}_{3 \times 2} \ \mathbf{I}_3 \ 0.5 \cdot \mathbf{1}_{3 \times 1}]^T,$$

$\mathbf{V} \triangleq \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$ ⁸. These matrices were adjusted empirically using the Bryson's method (Johnson & Grimble, 1987) as starting point. For comparison with results obtained using the zonotope-based algorithm of Rego & Raffo (2019), the sets X_0 , W_k , and V_k , were chosen as the zonotopes $X_0 = \{\mathbf{G}_{x_0}, [(\boldsymbol{\zeta}_0^{\text{tr}})^T \ \mathbf{0}_{1 \times 20}]^T\}$, $W_k = \{\mathbf{G}_w, \mathbf{0}_{23 \times 1}\}$ and $V_k = \{\mathbf{G}_v, \boldsymbol{\pi}(\mathbf{x}^{\text{eq}})\}$, with $\mathbf{G}_{x_0} = \text{diag}(0.5 \cdot \mathbf{1}_{3 \times 1}, 0.02, 0.02, \pi/180, 1.5 \cdot |\gamma_1^{\text{eq}}|, 1.5 \cdot |\gamma_2^{\text{eq}}|, 1.5 \cdot |\alpha_R^{\text{eq}}|, 1.5 \cdot |\alpha_L^{\text{eq}}|, 0.02 \cdot \mathbf{1}_{13 \times 1})$, $\mathbf{G}_w = \text{diag}(10^{-4} \cdot \mathbf{1}_{13 \times 1}, 0.01 \cdot \mathbf{1}_{3 \times 1}, 0.05 \cdot \mathbf{1}_{2 \times 1}, 10^{-4} \cdot \mathbf{1}_{2 \times 1}, 0.01 \cdot \mathbf{1}_{3 \times 1})$, $\mathbf{G}_v = \text{diag}(0.18 \cdot \mathbf{1}_{2 \times 1}, 0.612, 3.1416 \cdot 10^{-3} \cdot \mathbf{1}_{2 \times 1}, 0.03, 19.872 \cdot 10^{-3} \cdot \mathbf{1}_{2 \times 1}, 0.24, 0.006 \cdot \mathbf{1}_{2 \times 1}, 0.06, 6.8067 \cdot 10^{-3} \cdot \mathbf{1}_{2 \times 1}, 0.6093 \cdot \mathbf{1}_{2 \times 1})$. These bounds were adjusted by trial and error to accommodate the linearization errors.

⁸ $\mathbf{1}_{n \times m}$ is an n by m matrix of ones.

The number of generators and constraints of \hat{X}_k were limited to 50 and 10, respectively, using Methods 3.2 and 3.4, while the number of generators of the zonotopes was limited to 1725 using Method 3.1. The optimal estimate $\hat{\nu}_k \in \hat{X}_k$ was obtained considering $\mathbf{P}_0^v = \mathbf{I}_{23}$, with $\hat{\nu}_0$ as the center of the interval hull⁹ of $\hat{X}_0 = X_0 \cap_{\mathbf{H}_\nu^{[0]}} (\mathbf{y}_0 \oplus (-V_0))$, $\mathbf{P}_k^w = \frac{1}{3}\mathbf{G}_w^2$, and $\mathbf{P}_k^v = \frac{1}{3}\mathbf{G}_v^2$. These choices of \mathbf{P}_k^w and \mathbf{P}_k^v assume W_k and V_k as being the respective 95% confidence regions for the minimum-variance feedback. The quadratic program (9.20) was solved using CPLEX (IBM, 2018). The initial state is $\mathbf{x}_0 = [0 \ 0 \ 1 \ \mathbf{0}_{1 \times 13}]^T$.

The trajectories performed by the load and the UAV are shown in Figure 9.2, and the tracking error is shown in Figure 9.3. External forces were added and removed from the load, as filtered steps with magnitude 0.5 N. The estimation error of the regulated variables is shown in Figure 9.4, along with confidence limits (interval hull), and results using the strategy described in Rego & Raffo (2019). Sharper bounds are observed for the proposed method, which demonstrate the reduced conservatism of constrained zonotopes. This is a direct result of the exact intersection (9.16), which was severely over-estimated by zonotopic enclosures even with a much higher number of generators. Note that the confidence limits are not symmetric, since the optimal choice ν_k is not the geometric center of \hat{X}_k in general. Moreover, since outliers can happen due to unmodeled dynamics, if $\hat{X}_k = \emptyset$ (verified through Property 3.1), \bar{X}_k was used instead. The path tracking was performed successfully, as shown by the tracking error of the regulated variables, which remained bounded during the experiment¹⁰. The control signals applied to the aircraft are shown in Figure 9.5. To conclude, experiments were conducted with an alternative feedback $\hat{\nu}_k \in \hat{X}_k$ given by the center of the interval hull of \hat{X}_k . Although less computationally demanding (since it requires the solution of LPs instead of a QP), such choice resulted to be substantially noisy, destabilizing the closed-loop system in the experiment.

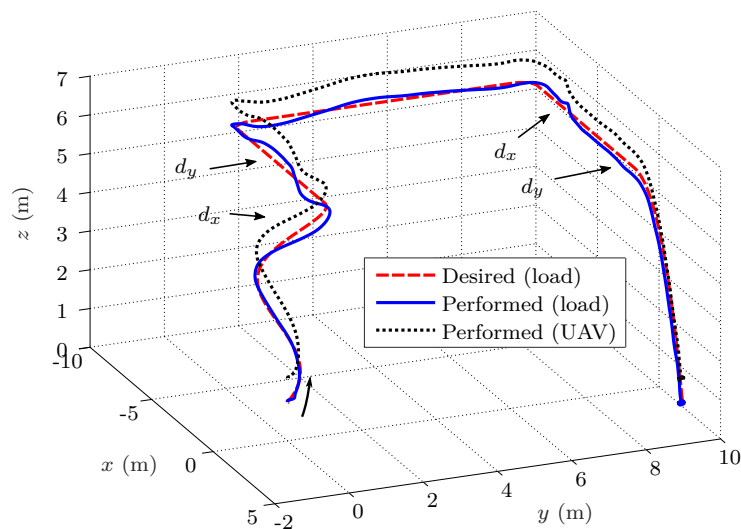


Figure 9.2: Trajectories performed by the UAV and the load.

⁹The interval hull is computed as in Property 3.3.

¹⁰A video of the performed experiment is available in <https://youtu.be/x-oJNQqdavs>.

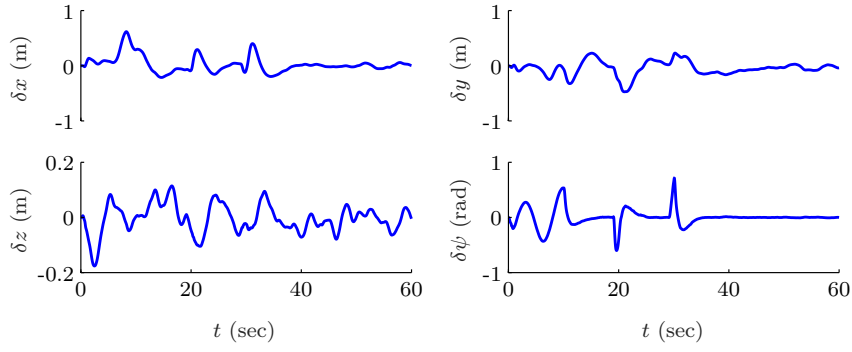


Figure 9.3: Tracking error of the regulated variables.

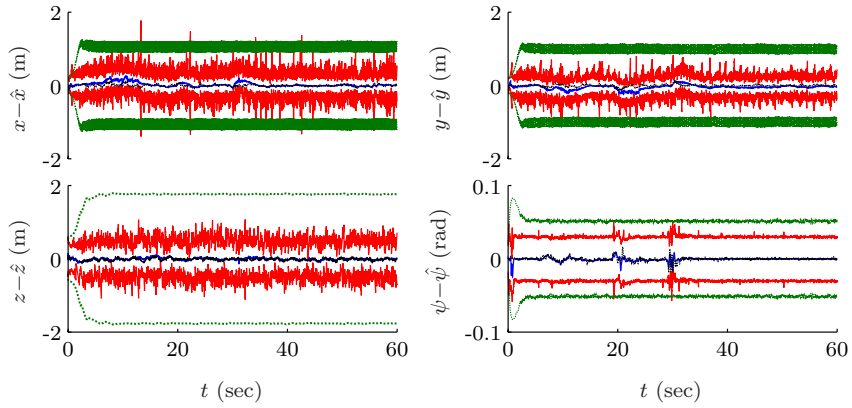


Figure 9.4: Estimation error of the regulated variables obtained using the CZs (blue solid line) and the zonotope-based algorithm from Rego & Raffo (2019) (black dotted line), along with confidence limits (red solid lines for CZs, and green dotted lines for zonotopes).

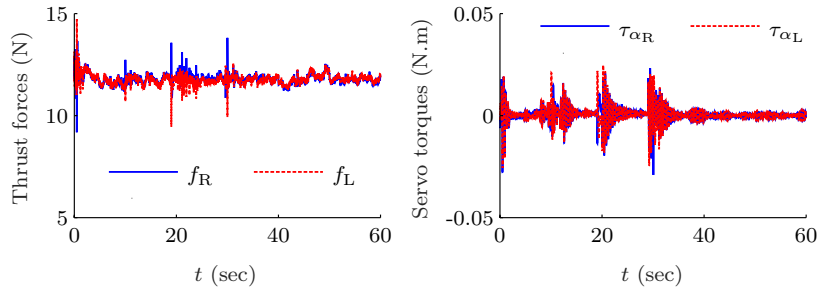


Figure 9.5: Control signals applied to the UAV.

9.2 Nonlinear state estimation of a quadrotor UAV

This section presents numerical results provided by the mean value and first-order Taylor extensions proposed in Chapter 4 when applied to nonlinear set-based state estimation of a quadrotor UAV. The content of this section as published in Rego et al. (2021d).

Consider the quadrotor UAV described in Mistler et al. (2001) with state vector $\zeta = [x \ y \ z \ u \ v \ w \ \phi \ \theta \ \psi \ p \ q \ r]^T$, where $[x \ y \ z]^T$ is the position of the UAV with respect to the inertial frame, $[u \ v \ w]^T$ is the velocity vector expressed in the inertial frame, $[\phi \ \theta \ \psi]^T$ are Euler angles describing the orientation of the UAV, and $[p \ q \ r]^T$ is the angular velocity

Table 9.3: Model parameters of the quadrotor UAV considered in the state estimation experiment.

Parameter	m	l	I_{xx}, I_{yy}, I_{zz}	g
Value	0.7 kg	0.3 m	1.2416 kg·m ²	9.81 m/s ²

vector expressed in the body frame. The equations of motion are (Mistler et al., 2001):

$$\dot{\zeta} = \begin{cases} \dot{x} = u, & \dot{u} = \frac{1}{m}(\cos \psi \sin \theta \cos \phi + \sin \psi \sin \phi)U_1 + \frac{1}{m}D_x, \\ \dot{y} = v, & \dot{v} = \frac{1}{m}(\sin \psi \sin \theta \cos \phi - \cos \psi \sin \phi)U_1 + \frac{1}{m}D_y, \\ \dot{z} = w, & \dot{w} = -g + \frac{1}{m}(\cos \theta \cos \phi)U_1 + \frac{1}{m}D_z, \\ \dot{\phi} = p + q \sin \phi \tan \theta + r \cos \phi \tan \theta, & \dot{p} = \frac{I_{yy} - I_{zz}}{I_{xx}}qr + \frac{l}{I_{xx}}U_2, \\ \dot{\theta} = q \cos \phi - r \sin \phi, & \dot{q} = \frac{I_{zz} - I_{xx}}{I_{yy}}pr + \frac{l}{I_{yy}}U_3, \\ \dot{\psi} = q \sin \phi \sec \theta + r \cos \phi \sec \theta, & \dot{r} = \frac{I_{xx} - I_{yy}}{I_{zz}}pq + \frac{1}{I_{zz}}U_4, \end{cases} \quad (9.24)$$

where m , I_{xx} , I_{yy} , I_{zz} , and l are physical parameters, g is the gravitational acceleration, U_1 is the total thrust generated by the propellers, U_2 is the difference of thrusts between the left and right propellers, U_3 is the difference of thrusts between the front and back propellers, U_4 is the difference of torques between clockwise and counter-clockwise turning propellers, and $\mathbf{d} = [D_x \ D_y \ D_z]^T$ are disturbance forces applied to the UAV with $\|\mathbf{d}\|_\infty \leq 1$. The experiment consists in obtaining guaranteed bounds on the system states ζ while the quadrotor UAV tracks a vertical helix trajectory defined by the reference values $x^{\text{ref}}(t) = \frac{1}{2} \cos(\frac{t}{2})$, $y^{\text{ref}}(t) = \frac{1}{2} \sin(\frac{t}{2})$, $z^{\text{ref}}(t) = 1 + \frac{t}{10}$, $\psi^{\text{ref}} = \frac{\pi}{3}$, subject to the disturbance forces described by $D_x = 1$ N for $t \in [5, 15)$ s, $D_y = 1$ N for $t \in [8, 15)$ s, and $D_z = 1$ N for $t \in [10, 15)$ s. These forces are zero otherwise. The dynamic feedback controller in Mistler et al. (2001) is used to track the reference trajectory above¹¹. The simulation parameters are shown in Table 9.3.

We consider a realistic scenario in which the available measurements are provided by sensors located at the quadrotor UAV, which include: (i) a Global Positioning System (GPS); (ii) a barometer; and (iii) an Inertial Measurement Unit (IMU). The measurements are affected by bounded uncertainties as described in Table 9.4. The velocity vector $[u \ v \ w]^T$ is not measured.

The nonlinear equations (9.24) were discretized by Euler approximation with sampling time 0.01 s. The initial states ζ_0 are bounded by $X_0 = \{\mathbf{G}_0, \mathbf{0}\}$, where $\mathbf{G}_0 = \text{diag}(2, 2, 2, 1, 1, 1, \frac{\pi}{6}, \frac{\pi}{6}, \frac{\pi}{2}, \frac{\pi}{12}, \frac{\pi}{12}, \frac{\pi}{12})$. To generate process measurements, the discrete-time

¹¹In this experiment, the control action is computed using the real states ζ_k . An extension of the minimum-variance approach proposed in Section 9.1.3 can be used for feedback connection using a point that belongs to \hat{X}_k .

Table 9.4: Measured variables with error bounds.

Sensor	Variables	Noise bounds
GPS	$\{x, y\}$	± 0.15 m
Barometer	$\{z\}$	± 0.51 m
IMU	$\{\phi, \theta, \psi\}$	$\pm 2.618 \cdot 10^{-3}$ rad
	$\{p, q, r\}$	$\pm 16.558 \cdot 10^{-3}$ rad/s

dynamics were simulated with $\zeta_0 = [0.5 \ 0 \ 1 \ \mathbf{0}_{1 \times 5} \ \pi/3 \ \mathbf{0}_{1 \times 3}]^T \in X_0$ and process and measurement noises drawn from uniform distributions. Figure 9.6 shows the trajectory performed by the quadrotor UAV along with the interval hulls¹² of the enclosures computed by the methods CZMV and ZMV, projected onto (x, y, z) -axes. CZMV was implemented with γ_x given by $C2$, and since (9.24) is also affine in $\mathbf{w}_k \triangleq \mathbf{d}_k$, Theorem 4.2 was implemented with $Z_w \supseteq \mu(\gamma_w, W)$ computed as described at the end of Remark 4.6. The number of constraints and generators of the computed constrained zonotopes was limited to 40 and 12, respectively, while the number of generators of the computed zonotopes was limited to 40.

The interval hulls of the constrained zonotopes obtained by CZMV were smaller than those from ZMV, demonstrating the accuracy of the proposed method. Figure 9.7 shows the radii of the constrained zonotopes and zonotopes computed by CZMV and ZMV, respectively. Both algorithms were capable of providing tight bounds on the system states $\zeta_k \in \mathbb{R}^{12}$. Nevertheless, CZMV provided less conservative bounds than ZMV, even for a high-order nonlinear dynamical system such as (9.24) (the CZMV-to-ZMV ARR was 74.41%). Finally, Figure 9.8 compares the radii of the update sets computed by ZFO and CZFO with γ_x given by $C3$ ¹³. Once again, CZFO provided less conservative bounds than ZFO (the CZFO-to-ZFO ARR was 74.45%). The results from CZMV and CZFO were again very similar, with CZMV providing marginally better results (the CZMV-to-CZFO ARR was 99.93%). The ARR for different numbers of constraints are shown in Table 9.5, with the average computed considering in addition simulations with different numbers of generators. Execution times are shown in Table 9.6. Note that most of the CZFO execution times are smaller than the ones presented in Table 4.4. This might be counter-intuitive since the current example has more state variables. However, the use of $C4$ in Example 1 results in a relatively more complex enclosure and therefore requires a much higher execution time for generator reduction and constraint elimination. Note that in this example, the computational times of the state estimators were greater than the considered sampling time of 0.01 s. Nevertheless, this fact does not invalidate the obtained results since these were run in a numerical simulation. Better times can be achieved by optimized implementation of the algorithms and using more powerful hardware, for instance. Besides, note that even

¹²The conversion from CG-rep to H-rep (see Property 3.2) for the purposes of exact drawing is intractable for the constrained zonotopes in this example.

¹³The increased complexity of the constrained zonotopes provided by $C4$ proved to be intractable for this example.

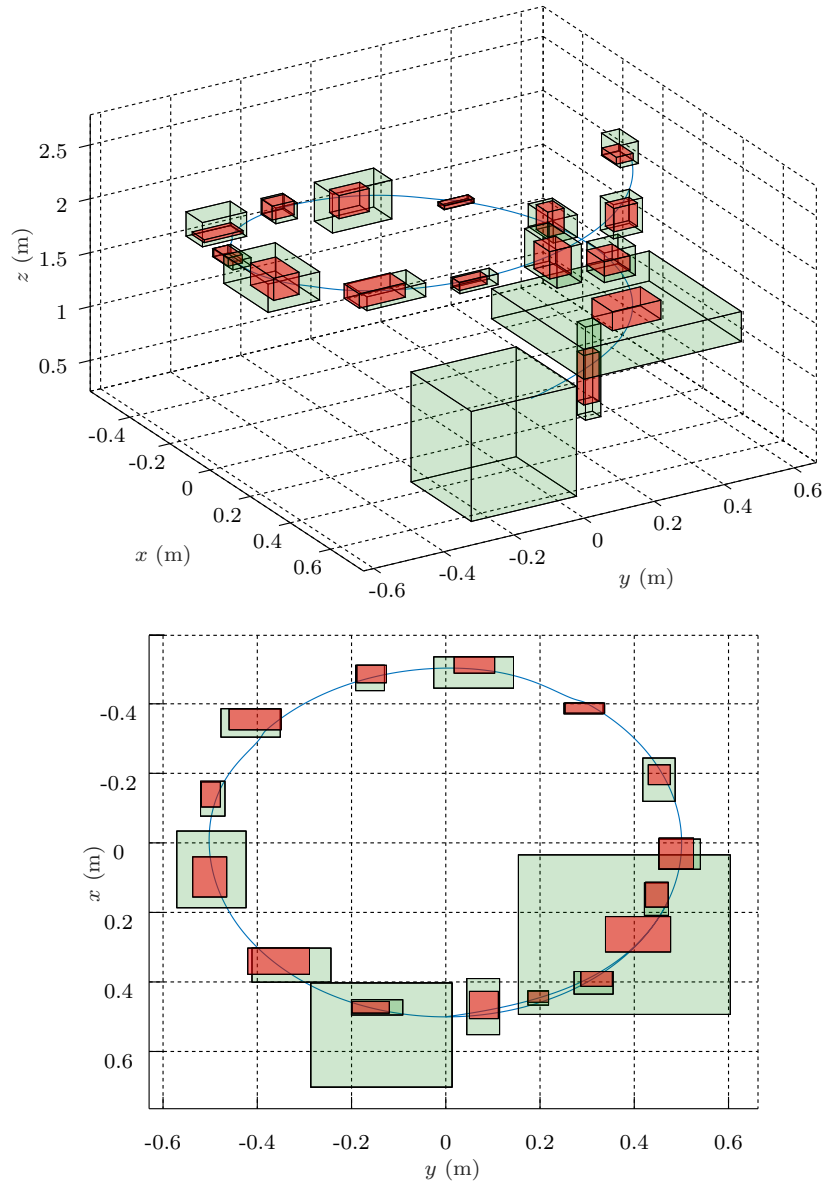


Figure 9.6: The trajectory performed by the quadrotor UAV (solid line), and the interval hulls of the constrained zonotopes (red boxes) and zonotopes (green boxes) estimated by CZMV and ZMV, respectively, projected onto (x, y, z) .

though the current execution times of CZMV and CZFO would in principle prevent their use in fast applications, the improved accuracy can significantly reduce the number of time steps required for guaranteed fault diagnosis for systems in which execution time is not critical.

Table 9.5: Average radius ratio of the estimators with varying numbers of constraints. Each average is taken over 3 separate simulations with $n_g \in \{25, 30, 40\}$.

n_c	CZMV/ZMV	CZFO/ZFO	CZFO/CZMV
3	82.5%	82.2%	99.9%
6	76.7%	77.1%	100.7%
12	75.0%	74.8%	99.7%

Table 9.6: Average total times per iteration of the estimators with varying numbers of constraints. Each average is taken over 15 separate simulations using $n_g \in \{25, 30, 40\}$. Times spent only on complexity reduction are shown in parenthesis.

n_c	ZMV	ZFO	CZMV	CZFO
0	53.3 (1.0) ms	174.2 (54.2) ms	–	–
3	–	–	104.7 (7.3) ms	266.9 (128.6) ms
6	–	–	109.5 (8.4) ms	615.0 (471.6) ms
12	–	–	127.8 (12.3) ms	2.62 (2.46) s

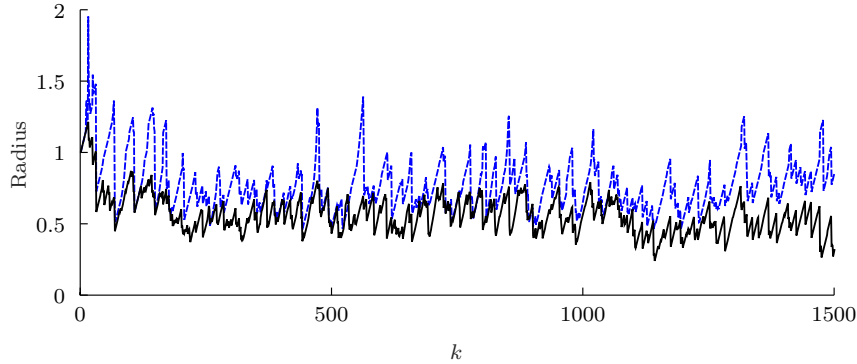


Figure 9.7: Radii of the update sets computed by CZMV (solid black line) and ZMV (dashed blue line) for the quadrotor UAV experiment.

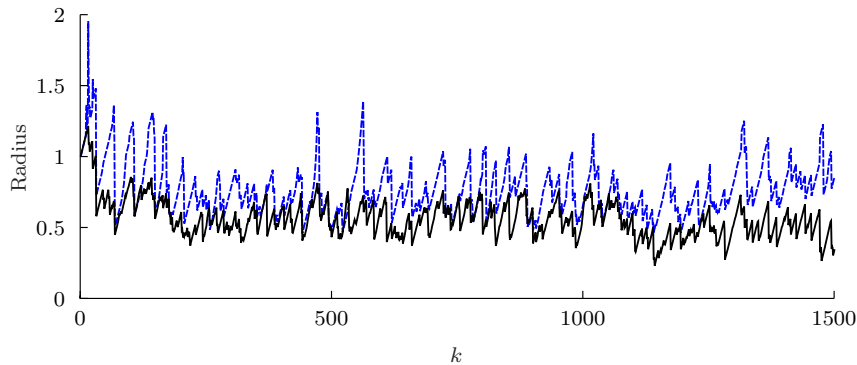


Figure 9.8: Radii of the update sets computed by CZFO (solid black line) and ZFO (dashed blue line) for the quadrotor UAV experiment.

9.3 Fault diagnosis of a quadrotor UAV

This section presents an example consisting in active fault isolation of a quadrotor UAV subject to actuator and sensor fault using the methods developed in Chapter 6.

Let $\zeta = [x \ y \ z \ u \ v \ w \ \phi \ \theta \ \psi \ p \ q \ r]^T$ be the system state, where $[x \ y \ z]^T$ is the position of the UAV with respect to the inertial frame, $[u \ v \ w]^T$ is the velocity vector expressed in the inertial frame, $[\phi \ \theta \ \psi]^T$ are Euler angles describing the orientation of the UAV, and $[p \ q \ r]^T$ is the angular velocity vector expressed in the body frame. Let $\mathbf{u} \in \mathbb{R}^4$ denote the thrust forces generated by the propellers. The equations of motion are¹⁴

¹⁴This is an adaptation of the model in Section 9.2 considering the thrust forces generated by the propellers as inputs, with inclination angle as described in Section 4.4.1 in Raffo (2011).

$$\dot{\boldsymbol{\zeta}} = \begin{cases} \dot{x} = u, & \dot{u} = \frac{1}{m}F_1 + \frac{1}{m}D_x, \\ \dot{y} = v, & \dot{v} = \frac{1}{m}F_2 + \frac{1}{m}D_y, \\ \dot{z} = w, & \dot{w} = -g + \frac{1}{m}F_3 + \frac{1}{m}D_z, \\ \dot{\phi} = p + q \sin \phi \tan \theta + r \cos \phi \tan \theta, & \dot{p} = \frac{I_{yy} - I_{zz}}{I_{xx}}qr + \frac{1}{I_{xx}}T_1, \\ \dot{\theta} = q \cos \phi - r \sin \phi, & \dot{q} = \frac{I_{zz} - I_{xx}}{I_{yy}}pr + \frac{1}{I_{yy}}T_2, \\ \dot{\psi} = q \sin \phi \sec \theta + r \cos \phi \sec \theta, & \dot{r} = \frac{I_{xx} - I_{yy}}{I_{zz}}pq + \frac{1}{I_{zz}}T_3, \end{cases} \quad (9.25)$$

where

$$\begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ T_1 \\ T_2 \\ T_3 \end{bmatrix} = \begin{bmatrix} \mathbf{I}_3 & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_{\mathcal{B}}^{\mathcal{I}} \end{bmatrix} \begin{bmatrix} 0 & l \cos(\alpha) & 0 & -l \cos(\alpha) \\ -l \cos(\alpha) & 0 & l \cos(\alpha) & 0 \\ \frac{k_{\tau}}{b} \cos(\alpha) & -\frac{k_{\tau}}{b} \cos(\alpha) & \frac{k_{\tau}}{b} \cos(\alpha) & -\frac{k_{\tau}}{b} \cos(\alpha) \\ -\sin(\alpha) & 0 & \sin(\alpha) & 0 \\ 0 & -\sin(\alpha) & 0 & \sin(\alpha) \\ \cos(\alpha) & \cos(\alpha) & \cos(\alpha) & \cos(\alpha) \end{bmatrix} \boldsymbol{\Omega}^{[i]} \mathbf{u},$$

with $\mathbf{R}_{\mathcal{B}}^{\mathcal{I}}$ defined equivalently as in (9.1) (Section 9.1.1), m , I_{xx} , I_{yy} , I_{zz} , l , g , α , k_{τ} , b are physical parameters given in Table 9.7, and $\mathbf{d} = [D_x \ D_y \ D_z]^T$ are disturbance forces applied to the UAV with $\|\mathbf{d}\|_{\infty} \leq 1$. The available measurement is given by

$$\mathbf{y}_k = \mathbf{s}^{[i]} + \mathbf{C}^{[i]} \mathbf{x}_k + \mathbf{D}_v^{[i]} \mathbf{v}_k, \quad (9.26)$$

with bounds on the measurement noise \mathbf{v}_k as described in Table 9.4. The actuator faults are modeled by $\boldsymbol{\Omega}^{[i]} \neq \mathbf{I}_4$, in which the faulty scenario corresponds to partial loss of potency of some of the thrusters. The sensor faults are modeled by $\mathbf{s}^{[i]} \neq \mathbf{0}$, which corresponds to measurement bias. This leads to a total of four possible scenarios, defined by

$$\begin{aligned} \boldsymbol{\Omega}^{[1]} = \boldsymbol{\Omega}^{[3]} = \mathbf{I}_4, \quad \boldsymbol{\Omega}^{[2]} = \boldsymbol{\Omega}^{[4]} = \text{diag}(1, 0.8, 0.3, 1), \\ \mathbf{s}^{[1]} = \mathbf{s}^{[2]} = \mathbf{0}_{9 \times 1}, \quad \mathbf{s}^{[3]} = \mathbf{s}^{[4]} = [\mathbf{0}_{1 \times 5} \ \pi/6 \ \mathbf{0}_{1 \times 3}]^T, \\ \mathbf{C}^{[1]} = \mathbf{C}^{[2]} = \mathbf{C}^{[3]} = \mathbf{C}^{[4]} = \begin{bmatrix} \mathbf{I}_3 & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 6} \\ \mathbf{0}_{6 \times 3} & \mathbf{0}_{6 \times 3} & \mathbf{I}_6 \end{bmatrix}, \\ \mathbf{D}_v^{[1]} = \mathbf{D}_v^{[2]} = \mathbf{D}_v^{[3]} = \mathbf{D}_v^{[4]} = \mathbf{I}_9. \end{aligned}$$

The nonlinear equations (9.25) were discretized by Euler approximation with sampling time 0.01 s. The initial state belongs to the constrained zonotope

$$X_0 = \left\{ [0.05 \cdot \mathbf{I}_{12} \ 0.05 \cdot \mathbf{1}_{12 \times 1}], \mathbf{0}_{12 \times 1}, [1 \ 0.5 \cdot \mathbf{1}_{1 \times 12}], 1 \right\}.$$

Table 9.7: Model parameters of the quadrotor UAV considered in the fault diagnosis experiment.

Parameter	m	l	I_{xx}, I_{yy}	I_{zz}
Value	2.24 kg	0.3332 m	0.0363 kg·m ²	0.0615 kg·m ²
Parameter	g	α	k_τ	b
Value	9.81 m/s ²	5°	1.7·10 ⁻⁷ N·m·s ²	9.5·10 ⁻⁶ N·s ²

For fault isolation, we consider an input sequence with length $N = 2$. The cost function is $J(\vec{\mathbf{u}}) = \|\vec{\mathbf{u}}\|_{\mathbf{R}}^2 = \sum_{j=0}^{N-1} \mathbf{u}_j^T \mathbf{R} \mathbf{u}_j$, with $\mathbf{R} = \mathbf{I}_4$. The complexity of the constrained zonotopes $\vec{Z}(i, j)$ is limited to 27 generators and 3 constraints using Methods 3.2 and 3.4, respectively. Moreover $\vec{\gamma}_w = \mathbf{0}$, and $\vec{\gamma}_u = (\mathbf{u}^{\text{eq}}, \mathbf{u}^{\text{eq}})$, with $\mathbf{u}^{\text{eq}} = (1/4)mg \sec(\alpha) \cdot \mathbf{1}_{4 \times 1}$. The vector \mathbf{h}_0 was chosen as $\gamma_{x,0} = \mathbf{0}$ (in this case $\mathbf{0} \in X_0$), with $\gamma_{x,k}$ obtained by propagating $(\gamma_{x,k-1}, \gamma_{u,k-1}, \gamma_{w,k-1})$ through the discretized version of the dynamics (9.25) for each model, $k \in \{1, 2\}$. The minimum separation threshold is $\varepsilon = 1.0 \cdot 10^{-8}$. The admissible input set is $U = \{3 \cdot \mathbf{1}_{4 \times 1}, \mathbf{u}^{\text{eq}}\}$.

The MIQP (6.29) was solved using MATLAB 9.1 and GUROBI 8.0.1, which resulted in the optimal separating input

$$\vec{\mathbf{u}} = \left(\begin{array}{c} \left[\begin{array}{c} 2.5146 \\ 2.5146 \\ 2.5146 \\ 2.5146 \end{array} \right] \\ \left[\begin{array}{c} 2.5146 \\ 2.5146 \\ 2.5146 \\ 2.5146 \end{array} \right] \end{array} \right). \quad (9.27)$$

This input sequence was found with execution time 3.58 s. The intersection (7.18) was verified to be empty for every $i, j \in \mathcal{M} = \{1, 2, 3, 4\}$, $i \neq j$, using the separating input (9.27). Moreover, the sequence (9.27) was injected into each model $i \in \mathcal{M}$, and $\vec{\mathbf{y}}^{[i]} \in \vec{\Psi}_{1:2}^{[i]}$ was satisfied for every $i \in \mathcal{M}$, with $\vec{\mathbf{y}}^{[i]} \notin \vec{\Psi}_{1:2}^{[j]}$ for every $i \neq j$. Therefore, the active model in (9.25)–(9.26) was guaranteed to be isolated by the injection of the separating input (9.27), allowing robust fault diagnosis of the quadrotor UAV when subject to actuator and sensor faults using the proposed active fault isolation method.

9.4 Final remarks

This chapter presented the application of set-based state estimation and fault diagnosis methods using constrained zonotopes to unmanned aerial vehicles.

The first application involved linear state estimation and state-feedback control for trajectory tracking of a suspended load using a tilt-rotor UAV. A constrained minimum-variance criterion was proposed for feedback connection with estimated states. The linear state estimation method based on constrained zonotopes was validated through numerical experiments carried out in the ProVANT simulator, a platform based on the

Gazebo simulator with a CAD model of the system, which demonstrated the reduced conservativeness of the proposed linear set-based state estimator, the performance of the optimal selection of estimated states, and the robustness of the designed controller.

The second application was a numerical experiment considering nonlinear state estimation of a quadrotor UAV, taking into account a realistic scenario in which measurements are provided by sensors located at the UAV. The third application considered a numerical experiment involving the optimal input design for nonlinear active fault diagnosis of a quadrotor UAV subject to actuator and sensor faults, which addressed scenarios of complete actuator failure and the existence of sensor bias.

10

Application: Water distribution networks

State estimation in water distribution networks (WDNs) is a challenging task due to the scarcity of measurements and the presence of several modeling uncertainties. In the literature, pseudo-measurements are often required to obtain an observable model for state estimation, which are highly uncertain and mostly represented by bounded sets. This chapter proposes two new methods for set-based state estimation in WDNs, considering unknown-but-bounded measurement and model uncertainties to calculate bounds on the system states. A new interval method is proposed based on iterative computation of tight enclosures of the nonlinear head-loss functions, and bounding the solution of the algebraic equations using rescaling. This method, referred to as Iterative Slope Approximation and Rescaling (ISAR), is mildly more conservative than the Iterative Hydraulic Interval State Estimation (IHISE) method previously proposed in [Vrachimis et al. \(2019\)](#), but significantly more computationally efficient. However, since intervals are not capable of capturing the dependencies between state variables, we propose the use of constrained zonotopes as an additional step to both IHISE and ISAR. This yields two new algorithms for set-based state estimation of WDNs, capable of capturing the dependencies between hydraulic states, which result in sets with significantly smaller volumes than intervals. The benefits of constrained zonotopes are also highlighted when the new enclosures are used for leakage detection. These provide higher leakage detection rates compared to intervals, as demonstrated in two case studies using benchmark WDNs. The content presented in this chapter has been submitted for publication in [Rego et al. \(2021e\)](#).

This chapter is organized as follows. Section [10.1](#) introduces the main topic of this

chapter. The topology of the water distribution networks considered in this chapter is described in Section 10.2 along with the corresponding mathematical formulation. Section 10.3 reviews the interval method proposed in [Vrachimis et al. \(2019\)](#), introduces the required mathematical background on CZs, describes a new interval method with reduced computational cost, and present the final enclosures described by CZs. The use of these enclosures for leakage detection in WDNs is discussed in Section 10.4. Numerical examples are presented in Section 10.5, and Section 10.6 concludes the chapter.

10.1 Introduction

A Water Distribution Network (WDN) is a system responsible for delivering water to consumers from water sources. The water industry is being modernized with the installation of sensors for monitoring WDNs, the use of supervisory control, and data acquisition (SCADA) systems. Integrated monitoring platforms can combine real-time sensor measurements with hydraulic models and geographical information systems. However, there is still the need for intelligent algorithms to process the acquired data, and use them for important tasks such as state estimation. State estimation algorithms in WDNs infer the internal system states, such as water flows in pipes and pressures at nodes, given the available measurements. A complete view of system states supports the operators in decision-making and enables the early detection of hydraulic faults, such as leakages. It is estimated that every day more than 45 million m^3 of treated water is lost due to leakages in developing countries, which could have served 200 million consumers ([Kingdom et al., 2006](#)).

State estimation in WDNs is a challenging task due to the scarcity of measurements compared to the system size. Indeed, observability in WDNs is only achievable in networks of large pipes which bring water from sources to water utilities, called the transport networks ([Vrachimis et al., 2018b](#)). In these networks, all the inflows and outflows are measured in real-time, which is a sensor configuration that guarantees observability ([Díaz et al., 2017](#)). However, most pipe failures in WDNs occur in urban distribution networks which may have hundreds of outlets that are not measured in real time. Water utilities typically sectorize urban networks into District Metered Areas (DMAs) and measure the inlet flow and pressure of each sub-network. A number of pressure sensors may exist in a DMA, typically used for pressure monitoring. To obtain hydraulic observability, groups of consumer nodes are modeled as a single outlet and a high resolution estimate of their consumption is determined using knowledge about population distributions and historical data ([Avni et al., 2015](#)). However, these so-called pseudo-measurements of consumer demands are highly uncertain with no statistical characterization of their error to the true consumption. The best way to characterize their uncertainty is with the use of error bounds ([Arsene et al., 2011](#)). Moreover, WDN hydraulic models are often uncertain in their

parameters, which clearly affects the quality of state estimation (Díaz et al., 2018). The most notable uncertain parameter is pipe roughness, which can be described as a measure of the impedance each pipe exhibits to the water flowing through it. This parameter cannot be determined experimentally for every pipe, thus an estimate based on the pipe material and age is usually adopted (Boulos et al., 2006).

Standard state estimation techniques used in WDNs, such as Kalman filtering and weighted least squares, require a measurement set that makes the system observable and the statistical characterization of sensor measurement error to give more weight to measurements originating from more-accurate sensors (Nagar & Powell, 2000; Díaz et al., 2016). Then, using a mathematical model of the system, they produce an estimated point and covariance (Kang & Lansey, 2010). Given the nature of uncertainties in WDNs, the approach of modeling uncertainties of input data as intervals has been proposed in Bargiela et al. (2003) and Vrachimis et al. (2019). This approach results in a set-based state estimate which is often more useful to an operator than providing point and covariance estimates, due to the lack of statistical characterization of all the uncertainties involved.

Set-based state estimation methods are often called Confidence Limit Analysis in the WDNs literature. The use of bounds for the representation of measurement uncertainty in WDNs was first introduced in Bargiela & Hainsworth (1989), where it was incorporated to estimate interval enclosures of the states. A common way to obtain set-based states in WDNs using a nonlinear model is through Monte-Carlo simulations (MCS) (Eliades et al., 2015). Nevertheless, this approach is computationally intensive, and the actual lower and upper bounds may not be reached even with a large number of simulations. On the other hand, bounds can be obtained without the need for extensive simulations using different methods such as the Error Maximization method (Arsene et al., 2011), the Ellipsoid method and Linear Programming (Bargiela et al., 2003). The bounds generated by these approaches are not guaranteed because of the linearization uncertainty introduced when using a linear approximation of the WDN hydraulic model, while they often neglect parameter uncertainty. An interesting approach when dealing with systems with bounded uncertainties is interval-based state estimation which has been studied extensively for linear systems (Efimov et al., 2013; Wang et al., 2015), and also used for specific classes of nonlinear systems (Raïssi et al., 2012; Vrachimis et al., 2019). In particular, a recent interval-based state estimation approach, specific for water systems, was proposed in Vrachimis et al. (2019), in which an optimization-based method calculates interval enclosures of the states using a nonlinear hydraulic model and considering parameter, input, and measurement uncertainties. It is shown that the provided bounds on the states approximate closely those obtained by performing a large number of MCS. Such bounds can be used not only for state estimation, but also for leak detection (Vrachimis et al., 2018a).

In contrast to intervals, zonotopes are capable of capturing potential dependencies between state variables, while mitigating the wrapping effect inherent to state estimation

methods based entirely on interval analysis (Kühn, 1998). Zonotopes have been used in parameter estimation and leak detection of WDNs in Blesa et al. (2010), and leak localization in Cugueró-Escofet et al. (2015). Specifically, the latter requires linearization of the output equations, and conversions between the generator representation of zonotopes to polytopes either in half-space representation or in vertex representation. These conversions have combinatorial and exponential complexities, respectively, and can lead to dramatic computational times in higher dimensions (Althoff et al., 2010). On the other hand, while maintaining the computational advantages of zonotopes and also being able to capture the dependency of state variables, constrained zonotopes incorporate equality constraints between the state variables into the mathematical description (see Chapters 5, 7, and 8). Nevertheless, despite these advantages, to the best of our knowledge the properties of constrained zonotopes have not been yet explored in methods for set-based state estimation and leak detection in WDNs.

This chapter proposes two new methods for set-based state estimation of WDNs based on constrained zonotopes, considering unknown-but-bounded water demands and model parameters. Specifically, we first formulate the mathematical model of a WDN from the laws of conservation of energy and mass, which is later approximated by linear algebraic constraints. Accordingly, we propose an interval method based on iterative computation of tight enclosures of the water flows by bounding the solution of the linear algebraic equations using rescaling (Scott et al., 2016), and subsequently re-approximating the nonlinear equalities for the new flow bounds. This method is referred to as Iterative Slope Approximation and Rescaling (ISAR), which is mildly more conservative but significantly faster than the previous approach in Vrachimis et al. (2019). However, as mentioned before, intervals are not capable of capturing the dependencies between state variables. Therefore, in addition we propose the use of CZs to obtain tight bounds of the solution of the nonlinear algebraic equations, with linear approximation based on the final interval enclosures provided by both the approach described in Vrachimis et al. (2019), and ISAR. These two new algorithms for set-based state estimation of WDNs, with a fair trade-off between efficiency and accuracy, are both capable of capturing the dependencies between hydraulic heads and water flows. This results in sets with significantly smaller volumes than intervals, and therefore with improved accuracy. The obtained enclosures are further used in passive leak detection of WDNs, in which, by being less conservative than intervals, allow a significant higher number of detections in several scenarios with different leak magnitudes. We investigate the key advantages of the proposed methods for set-based state estimation and leak detection in WDNs in numerical examples and also a realistic network, which highlight their improved efficiency and accuracy in comparison to intervals.

10.2 Problem formulation

The topology of a WDN is modeled by a directed graph denoted as $\mathcal{G} = (\mathcal{N}, \mathcal{L})$. Let $\mathcal{N} = \{1, \dots, n_h\}$ be the set of all nodes, where $|\mathcal{N}| = n_h$ is the total number of nodes. These represent junctions of pipes, consumer water demand locations, reservoirs and tanks. The hydraulic state associated with the nodes is the *hydraulic head*, indicated by h_j , $j \in \mathcal{N}$. The hydraulic head consists of the pressure at node j , whose minimum is the node elevation η_j with respect to a geodesic reference. Each node j is also associated with a water consumer demand at the node location, denoted by d_j . Demands drive the dynamics of a WDN and are an uncertain input to the system. Let $\mathcal{L} = \{1, \dots, n_q\}$ be the set of links, where $|\mathcal{L}| = n_q$ is the total number of links. These represent network pipes, water pumps and pipe valves. The hydraulic quantity associated with a link $i \in \mathcal{L}$ is the water flow, indicated by q_i . Moreover, u_i denotes the contribution of the level of water reservoirs and tanks to the i -th link.

The system of hydraulic algebraic equations which describe the behavior of the plant are derived from the laws of conservation of energy and mass for water networks. Consider a WDN with n_q links and n_h nodes, described by the nonlinear system of equations

$$\mathbf{f}(\mathbf{q}_k) + \mathbf{B}\mathbf{h}_k = \mathbf{u}_k, \quad (10.1a)$$

$$\mathbf{B}^T \mathbf{q}_k = \mathbf{d}_k, \quad (10.1b)$$

where $k \geq 0$ denotes the time instant, $\mathbf{q}_k \in \mathbb{R}^{n_q}$ denote the water flows at the links, $\mathbf{h}_k \in \mathbb{R}^{n_h}$ denote the node heads, $\mathbf{B} \in \mathbb{R}^{n_q \times n_h}$ denotes the incidence matrix (whose elements are -1 , 0 , or 1 depending on the connectivity of nodes with links), $\mathbf{u}_k \in \mathbb{R}^{n_q}$ is the contribution of the level of water reservoirs and tanks, and $\mathbf{d}_k \in \mathbb{R}^{n_h}$ is the water demand by the consumers at the nodes. The nonlinear mapping $\mathbf{f}(\mathbf{q}_k) : \mathbb{R}^{n_q} \rightarrow \mathbb{R}^{n_q}$ denotes the head loss at the links, with its i -th component $f_i(\mathbf{q}_k) \triangleq f_i(q_{i,k})$ being given by

$$f_i(q_{i,k}) = \begin{cases} r_i |q_{i,k}|^{(\nu-1)} q_{i,k} & \text{if link } i \text{ is a pipe,} \\ -(w_1 - w_2 q_{i,k}^{w_3}) & \text{if link } i \text{ is a pump,} \end{cases} \quad (10.2)$$

with ν , w_1 , w_2 , $w_3 \in \mathbb{R}$ being known parameters derived from physical properties of their corresponding elements (i.e., pipes and pumps) (Boulos et al., 2006). On the other hand, the parameter $\mathbf{r} \in \mathbb{R}^{n_q}$ is assumed to be uncertain and to belong to a known interval R .

Given the known information on \mathbf{u}_k and \mathbf{d}_k at instant k , for $k \geq 0$, the objective of this chapter is to obtain tight bounds Q_k and H_k satisfying $\mathbf{q}_k \in Q_k$ and $\mathbf{h}_k \in H_k$. It is assumed that interval bounds U_k and D_k of \mathbf{u}_k and \mathbf{d}_k , respectively, are known from historical data, such that $\mathbf{u}_k \in U_k$, $\mathbf{d}_k \in D_k$, for all $k \geq 0$. In addition, we assume that conservative interval bounds on the state $\mathbf{x}_k \triangleq (\mathbf{q}_k, \mathbf{h}_k)$ can be derived from physical constraints or available historical data.

10.3 Set-based state estimation of WDNs

10.3.1 Iterative Hydraulic Interval State Estimation

Iterative Hydraulic Interval State Estimation (IHISE) is a set-based state estimation methodology for WDNs first proposed in [Vrachimis et al. \(2018b\)](#) and further developed in [Vrachimis et al. \(2019\)](#). The methodology calculates bounds on the system states satisfying the nonlinear algebraic equations (10.1), given bounds on water demands and the model parameter \mathbf{r} . This is achieved using bounding linearization, a technique which restricts the nonlinearities within a convex set, thus converting the hydraulic equations in a form where state bounds can be calculated using convex optimization techniques. An iterative procedure minimizes the distance between upper and lower state bounds, by reducing the feasible set defined by bounding linearization at each step and converging to the tightest possible interval bounds. The IHISE is summarized in the following four steps.

10.3.1.1 Initial bounds on the state vector

The initial enclosure of the state $\mathbf{x}_k \in X_k$ is obtained by considering the physical properties of the network. The initial lower bounds $\mathbf{h}_k^{\text{L}(0)}$ on the unknown head vector \mathbf{h}_k are set as equal to the elevation of each node, while the initial upper bounds $\mathbf{h}_k^{\text{U}(0)}$ are equal to the sum of reservoir and pump heads, which is physically the maximum head that any node in the WDN can achieve. Given $[\mathbf{h}_k^{\text{L}(0)}, \mathbf{h}_k^{\text{U}(0)}]$, and the known bounds U_k of the reservoirs and tanks levels \mathbf{u}_k , by noting that each equation in (10.1a) is a function of only one $q_{i,k}$, and that the head-loss functions $f_i(q_{i,k})$ are inclusion isotonic ([Moore et al., 2009](#)) (meaning that $A \subseteq B \implies f(A) \subseteq f(B)$), we have that the initial bounds on the unknown water flows can be obtained using interval arithmetic. This is done by rearranging (10.1a) with respect to q_i , $i \in \mathcal{L}$, and applying the inverse of f_i . The initial bounds of the flow and head states are indicated by $Q_k^{(0)}$, and $H_k^{(0)}$, respectively.

10.3.1.2 Bounding linearization of the interval nonlinear terms

Let $m \geq 1$ denote the iteration index. This step encloses the uncertain nonlinear terms $\mathbf{f}(\mathbf{q}_k)$ by a convex set, for all¹ $\mathbf{q}_k \in Q_k^{(m-1)} = [\mathbf{q}_k^{\text{L}}, \mathbf{q}_k^{\text{U}}]$ and all the uncertain parameters $\mathbf{r} \in R$, using bounding linearization ([Vrachimis et al., 2019](#)). One way to achieve this is to enclose the image of each component of the nonlinear function $f_i(q_{i,k})$, for all $q_{i,k} \in [q_{i,k}^{\text{L}}, q_{i,k}^{\text{U}}]$ and $\mathbf{r} \in R$, between a lower line $\bar{f}_i^{\text{L}}(q_{i,k})$ and an upper line $\bar{f}_i^{\text{U}}(q_{i,k})$. These lines are defined as

$$\bar{f}_i^{\text{L}}(q_{i,k}) = \lambda_i^{\text{L}} q_{i,k} + \beta_i^{\text{L}}, \quad \forall i \in \mathcal{L}, \quad (10.3a)$$

$$\bar{f}_i^{\text{U}}(q_{i,k}) = \lambda_i^{\text{U}} q_{i,k} + \beta_i^{\text{U}}, \quad \forall i \in \mathcal{L}, \quad (10.3b)$$

¹For notational simplicity, we omit the dependence on the iteration index m of some variables, such as most of the lower and upper bounds of intervals.

with λ_i^L , λ_i^U , β_i^L , and β_i^U computed as explained in [Vrachimis et al. \(2019\)](#). The corresponding convex enclosure is defined component-wise by the linear inequalities

$$\bar{f}_i^L(q_{i,k}) \leq f_i(q_{i,k}) \leq \bar{f}_i^U(q_{i,k}), \forall q_{i,k} \in [q_{i,k}^L, q_{i,k}^U], i \in \mathcal{L} \quad (10.4)$$

10.3.1.3 Solution of the linear interval system of equations using linear programming

The procedure described in Step 2 yields linear inequality constraints that allow one to replace $f_i(q_{i,k})$ in [\(10.1a\)](#) with new variables ζ_i , thus allowing the transformation of all the algebraic equations in [\(10.1\)](#) into a set of linear inequalities. Let $Q_k^{(m-1)} = [\mathbf{q}_k^L, \mathbf{q}_k^U]$, $H_k^{(m-1)} = [\mathbf{h}_k^L, \mathbf{h}_k^U]$, $U_k = [\mathbf{u}_k^L, \mathbf{u}_k^U]$, $D_k = [\mathbf{d}_k^L, \mathbf{d}_k^U]$, $\mathbf{\Lambda}^L = \text{diag}(\boldsymbol{\lambda}^L)$, and $\mathbf{\Lambda}^U = \text{diag}(\boldsymbol{\lambda}^U)$, with $\boldsymbol{\lambda}^L = [\lambda_1^L \cdots \lambda_{n_q}^L]^T$, and $\boldsymbol{\lambda}^U = [\lambda_1^U \cdots \lambda_{n_q}^U]^T$. An enclosure of the state set satisfying [\(10.1a\)](#)–[\(10.1b\)](#), can be obtained using the following linear inequalities:

$$\mathbf{u}_k^L \leq \boldsymbol{\zeta} + \mathbf{B}\mathbf{h} \leq \mathbf{u}_k^U, \quad (10.5a)$$

$$\mathbf{d}_k^L \leq \mathbf{B}^T \mathbf{q} \leq \mathbf{d}_k^U, \quad (10.5b)$$

$$\mathbf{\Lambda}^L \mathbf{q} + \boldsymbol{\beta}^L \leq \boldsymbol{\zeta} \leq \mathbf{\Lambda}^U \mathbf{q} + \boldsymbol{\beta}^U, \quad (10.5c)$$

$$\mathbf{q}_k^L \leq \mathbf{q} \leq \mathbf{q}_k^U, \quad (10.5d)$$

$$\mathbf{h}_k^L \leq \mathbf{h} \leq \mathbf{h}_k^U, \quad (10.5e)$$

where $\boldsymbol{\beta}^L = [\beta_1^L \cdots \beta_{n_q}^L]^T$ and $\boldsymbol{\beta}^U = [\beta_1^U \cdots \beta_{n_q}^U]^T$. Note that all the interval sets have been replaced by their respective lower and upper bounds. Then, a total of $2(n_q + n_h)$ linear programs (LPs) are then formulated using [\(10.5a\)](#)–[\(10.5e\)](#), with decision variables given by $[\mathbf{q}^T \ \mathbf{h}^T \ \boldsymbol{\zeta}^T]^T = [\mathbf{x}^T \ \boldsymbol{\zeta}^T]^T$. Each pair of the following LPs provide the lower and upper bounds of the j -th state variable x_j , with $j = 1, \dots, n_h + n_q$:

$$\begin{cases} x_j^L = \min_{\mathbf{x}, \boldsymbol{\zeta}} x_j \\ \text{s.t. (10.5a)–(10.5e),} \end{cases} \quad \begin{cases} x_j^U = \max_{\mathbf{x}, \boldsymbol{\zeta}} x_j \\ \text{s.t. (10.5a)–(10.5e).} \end{cases} \quad (10.6)$$

10.3.1.4 Iterative solution of the nonlinear system of equations

The procedure in Step 3 leads to an interval solution of [\(10.1\)](#), denoted by $\tilde{X}_k^{(m)}$, which encloses the exact solution set denoted by X_k^* . An iterative method is then used to find a tight $X_k^{(m)}$ with respect to X_k^* , resulting from $X_k^{(m)} = X_k^{(m-1)} \cap \tilde{X}_k^{(m)}$, with $X^{(0)} = Q^{(0)} \times H^{(0)}$. At each iteration m , the previous refined set $X_k^{(m-1)}$ is used in the bounding linearization procedure of Step 2 to reduce the convex set defined by the inequalities [\(10.5\)](#). Step 3 is then repeated to produce a smaller set $\tilde{X}_k^{(m)}$. The iterations of the IHISE terminate when the relative change in the obtained enclosure $X_k^{(m)}$, given by the metric $e_k^{(m)}$, is smaller

than a chosen tolerance value ε , i.e., $e_k^{(m)} < \varepsilon$. This metric $e_k^{(m)}$ is given by

$$e_k^{(m)} \triangleq \|(\mathbf{x}_k^{\text{U}(m)} - \mathbf{x}_k^{\text{L}(m)}) - (\mathbf{x}_k^{\text{U}(m-1)} - \mathbf{x}_k^{\text{L}(m-1)})\|_1. \quad (10.7)$$

The overall procedure of the IHISE methodology is described in Algorithm 10.1.

Algorithm 10.1 IHISE (Vrachimis et al., 2019)

- 1: Assign initial bounds $H_k^{(0)}$ using physical constraints.
 - 2: Compute initial bounds $Q_k^{(0)}$ according to Step 1.
 - 3: Assign $X_k^{(0)} = Q_k^{(0)} \times H_k^{(0)}$, and $m = 1$.
 - 4: Perform bounding linearization of $\mathbf{f}(\mathbf{q}_k)$ for $\mathbf{q}_k \in Q_k^{(m-1)}$.
 - 5: Solve the pair of LPs (10.6) for $j = 1, \dots, n_q + n_h$, and obtain $\tilde{X}_k^{(m)}$.
 - 6: Assign $X_k^{(m)} = X_k^{(m-1)} \cap \tilde{X}_k^{(m)}$.
 - 7: If $e^{(m)} < \varepsilon$, return $X_k^{(m)}$ and terminate. Otherwise, assign $m = m + 1$ and go to 4.
-

Note that each iteration in Algorithm 10.1 requires the solutions of $2(n_h + n_q)$ LPs. Assuming that the solution of each LP has the average complexity reported in Kelner & Spielman (2006), i.e., it has average complexity of $O(N_d N_c^3)$, in which N_d and N_c are the number of decision variables and constraints, respectively, then the computational complexity of Algorithm 10.1 is $O(\kappa_m n^5)$, where $n = n_h + n_q$, and κ_m corresponds to the number of performed iterations.

10.3.2 Slope approximation of the nonlinear equalities

Consider the initial bounds $Q_k^{(0)}$ and $H_k^{(0)}$ computed as in Section 10.3.1.1. The exact solution set of (10.1) for all $\mathbf{u}_k \in U_k$, $\mathbf{d}_k \in D_k$, is the set of all $\mathbf{q}_k \in Q_k^{(0)}$ and $\mathbf{h}_k \in H_k^{(0)}$ such that

$$\begin{aligned} \mathbf{f}(\mathbf{q}_k) + \mathbf{B}\mathbf{h}_k &\in U_k, \\ \mathbf{B}^T \mathbf{q}_k &\in D_k. \end{aligned} \quad (10.8)$$

The following lemma presents an approximation of the nonlinear equalities (10.1) based on the bounding linearization procedure described in Section 10.3.1.2. This is referred to as *slope approximation*. Instead of the inequalities (10.5), Lemma 10.1 approximates (10.1) by set of linear algebraic equations.

Lemma 10.1. Let $\mathbf{q}_k \in [\mathbf{q}_k^{\text{L}}, \mathbf{q}_k^{\text{U}}] \subset \mathbb{R}^{n_q}$, $\mathbf{h}_k \in \mathbb{R}^{n_h}$, and $(\mathbf{d}_k, \mathbf{u}_k) \in D_k \times U_k$, such that the nonlinear algebraic equations (10.1) hold. Then, there exists $\boldsymbol{\delta} \in B_\infty^{n_q}$, $\boldsymbol{\Delta} = \text{diag}(\boldsymbol{\delta})$, $\boldsymbol{\Lambda}_+ \in \mathbb{R}^{n_q \times n_q}$, $\boldsymbol{\Lambda}_- \in \mathbb{R}^{n_q \times n_q}$, $\boldsymbol{\beta}_+ \in \mathbb{R}^{n_q}$, and $\boldsymbol{\beta}_- \in \mathbb{R}^{n_q}$ such that

$$\begin{cases} \boldsymbol{\Lambda}_+ \mathbf{q}_k + \mathbf{B}\mathbf{h}_k = \mathbf{u}_k - \boldsymbol{\beta}_+ - \boldsymbol{\Delta}(\boldsymbol{\beta}_- + \boldsymbol{\Lambda}_- \mathbf{q}_k), \\ \mathbf{B}^T \mathbf{q}_k = \mathbf{d}_k. \end{cases} \quad (10.9)$$

Proof. For a given interval bound $[\mathbf{q}_k^{\text{L}}, \mathbf{q}_k^{\text{U}}]$, there exist real vectors $\boldsymbol{\lambda}^{\text{L}}, \boldsymbol{\lambda}^{\text{U}}, \boldsymbol{\beta}^{\text{L}}, \boldsymbol{\beta}^{\text{U}} \in \mathbb{R}^{n_q}$ such that $\lambda_i^{\text{L}} q_i + \beta_i^{\text{L}} \leq f_i(q_i) \leq \lambda_i^{\text{U}} q_i + \beta_i^{\text{U}}$ for all $i = 1, 2, \dots, n_q$, and $\mathbf{q}_k \in [\mathbf{q}_k^{\text{L}}, \mathbf{q}_k^{\text{U}}]$ (see Section

10.3.1). Therefore,

$$\begin{aligned} \lambda_i^L q_{i,k} + \beta_i^L &\leq f_i(q_{i,k}) \leq \lambda_i^U q_{i,k} + \beta_i^U \\ \iff f_i(q_{i,k}) &\in [\lambda_i^L q_{i,k} + \beta_i^L, \lambda_i^U q_{i,k} + \beta_i^U] \\ \iff f_i(q_{i,k}) &\in \left(\frac{1}{2} (\beta_i^U + \beta_i^L) + \frac{1}{2} (\lambda_i^U + \lambda_i^L) q_{i,k} \right) \\ &\quad \oplus \left(\frac{1}{2} (\beta_i^U - \beta_i^L) + \frac{1}{2} (\lambda_i^U - \lambda_i^L) q_{i,k} \right) B_\infty. \end{aligned}$$

Let $\mathbf{\Lambda}_+ = (1/2)\text{diag}(\boldsymbol{\lambda}^U + \boldsymbol{\lambda}^L)$, $\mathbf{\Lambda}_- = (1/2)\text{diag}(\boldsymbol{\lambda}^U - \boldsymbol{\lambda}^L)$, $\boldsymbol{\beta}_+ = (1/2)(\boldsymbol{\beta}^U + \boldsymbol{\beta}^L)$, $\boldsymbol{\beta}_- = (1/2)(\boldsymbol{\beta}^U - \boldsymbol{\beta}^L)$. Then, there exists $\boldsymbol{\delta} \in B_\infty^{n_q}$ such that

$$\mathbf{f}(\mathbf{q}_k) = \mathbf{\Lambda}_+ \mathbf{q}_k + \boldsymbol{\beta}_+ + \boldsymbol{\Delta} (\boldsymbol{\beta}_- + \mathbf{\Lambda}_- \mathbf{q}_k), \quad (10.10)$$

where $\boldsymbol{\Delta} = \text{diag}(\boldsymbol{\delta})$. Substituting (10.10) in the nonlinear equations (10.1) results in the system of equalities (10.9). \blacksquare

10.3.3 Iterative Slope Approximation and Rescaling

In this section, we propose an iterative method to find interval bounds of the solution set of (10.9) with modest computational complexity, without requiring to solve linear programs as in Algorithm 10.1. This method is based on iteratively enclosing the solution set of the slope approximation (10.9) by a constrained box in CG-rep, followed by the use of the rescaling procedure described in Chapter 3, to obtain sharp interval bounds of the solution. The resulting method is referred to as Iterative Slope Approximation and Rescaling.

Let $s \geq 1$ denote the iteration index, and $\tilde{\mathbf{u}}_k \triangleq \mathbf{u}_k - \boldsymbol{\beta}_+ - \boldsymbol{\Delta} (\boldsymbol{\beta}_- + \mathbf{\Lambda}_- \mathbf{q}_k)$. Moreover, let $\tilde{U}_k^{(s)} \triangleq [\tilde{\mathbf{u}}_k^L, \tilde{\mathbf{u}}_k^U]$ be the interval obtained by evaluating the right-hand-side of $\tilde{\mathbf{u}}_k$ for all $\mathbf{u}_k \in [\mathbf{u}_k^L, \mathbf{u}_k^U]$, $\mathbf{q}_k \in [\mathbf{q}_k^L, \mathbf{q}_k^U]$, and $\Delta_{ii} \in B_\infty$, using interval arithmetic. By defining the augmented vector $\tilde{\mathbf{p}}_k = (\mathbf{q}_k, \mathbf{h}_k, \tilde{\mathbf{u}}_k, \mathbf{d}_k) \in \tilde{P}_k^{(s)} \in \mathbb{I}\mathbb{R}^{2n_q + 2n_h}$, with

$$\begin{aligned} \tilde{P}_k^{(s)} &= Q_k^{(s-1)} \times H_k^{(s-1)} \times \tilde{U}_k^{(s)} \times D_k, \\ \tilde{\mathbf{B}} &= \begin{bmatrix} \mathbf{\Lambda}_+ & \mathbf{B} & -\mathbf{I} & \mathbf{0} \\ \mathbf{B}^T & \mathbf{0} & \mathbf{0} & -\mathbf{I} \end{bmatrix}, \end{aligned} \quad (10.11)$$

the equalities (10.9) can be rewritten in the form $\tilde{\mathbf{B}}\tilde{\mathbf{p}}_k = \mathbf{0}$. In addition, the relation $\tilde{\mathbf{p}}_k \in \tilde{P}_k^{(s)}$ implies that there must exist a vector $\bar{\mathbf{p}} \in B_\infty^{2n_q + 2n_h}$ satisfying

$$\tilde{\mathbf{p}}_k = \text{mid}(\tilde{P}_k^{(s)}) + \text{diag}(\text{rad}(\tilde{P}_k^{(s)}))\bar{\mathbf{p}}. \quad (10.12)$$

Then, replacing (10.12) into the equalities $\tilde{\mathbf{B}}\tilde{\mathbf{p}}_k = \mathbf{0}$ results in $\tilde{\mathbf{B}}(\text{mid}(\tilde{P}_k^{(s)}) + \text{diag}(\text{rad}(\tilde{P}_k^{(s)}))\bar{\mathbf{p}}) = \mathbf{0}$, which, in turn, implies that $\tilde{\mathbf{B}}(\text{diag}(\text{rad}(\tilde{P}_k^{(s)})))\bar{\mathbf{p}} = -\tilde{\mathbf{B}}(\text{mid}(\tilde{P}_k^{(s)}))$. Therefore, by defin-

ing the variables $\tilde{\mathbf{c}}_k = \text{mid}(\tilde{P}_k^{(s)})$, $\tilde{\mathbf{G}}_k = \text{diag}(\text{rad}(\tilde{P}_k^{(s)}))$, $\tilde{\mathbf{A}} = \tilde{\mathbf{B}}(\text{diag}(\text{rad}(\tilde{P}_k^{(s)})))\bar{\mathbf{p}}$, and $\tilde{\mathbf{b}}_k = -\tilde{\mathbf{B}}(\text{mid}(\tilde{P}_k^{(s)}))$, we have from (10.12) that the following relation holds:

$$\tilde{\mathbf{p}}_k \in \{\tilde{\mathbf{c}}_k + \tilde{\mathbf{G}}_k \bar{\mathbf{p}} : \|\bar{\mathbf{p}}\|_\infty \leq 1, \tilde{\mathbf{A}}_k \bar{\mathbf{p}} = \tilde{\mathbf{b}}_k\}, \quad (10.13)$$

in which the right-hand side is a constrained box in CG-rep (note that $\tilde{\mathbf{G}}_k$ is diagonal). The rescaling procedure described by Method 3.3 in Chapter 3 is then applied to the right-hand side of (10.13) to obtain an approximated interval hull of the solution of $\tilde{\mathbf{A}}_k \bar{\mathbf{p}} = \tilde{\mathbf{b}}_k$ and, consequently, obtain a tight interval bound of $\tilde{\mathbf{p}}_k$. Sharp bounds can be obtained using the iterative procedure described in Algorithm 3.1 (Section 3.3), whose complexity, in the case of (10.13), is $O(n^3)$ (assuming that every variable is proportional to the state dimension $n = n_q + n_n$). In addition, further accurate results can be obtained if the constraints $\tilde{\mathbf{A}}_k \bar{\mathbf{p}} = \tilde{\mathbf{b}}_k$ are first preconditioned using Gauss-Jordan elimination.

The proposed ISAR algorithm involving the rescaling of (10.13) is illustrated in Algorithm 10.2. To obtain an interval $X_k^{(s)}$ to be used in the subsequent iteration, we neglect the equality constraints of the constrained box obtained in the rescaling procedure, which, in this case, is equivalent to computing its interval hull. Moreover, we use the metric $\varepsilon^{(s)}$ given by (10.7), with $m \triangleq s$, as stop criterion. Since Algorithm 3.1 has complexity $O(n^3)$, we have that ISAR has complexity $O(\kappa_s n^3)$, with κ_s being the number of performed iterations by Algorithm 10.2, while IHISE has average complexity $O(\kappa_m n^5)$ (see Section 10.3.1).

Algorithm 10.2 ISAR

- 1: Assign initial bounds $H_k^{(0)}$ using physical constraints.
 - 2: Compute initial bounds $Q_k^{(0)}$ according to Section 10.3.1.1.
 - 3: Assign $X_k^{(0)} = Q_k^{(0)} \times H_k^{(0)}$, $s = 1$.
 - 4: Compute the slope approximation (10.9) for $\mathbf{q}_k \in Q_k^{(s-1)}$.
 - 5: Compute $\tilde{P}_k^{(s)}$ as in (10.11).
 - 6: Rescale the constrained box in (10.13) using Method 3.3 and Algorithm 3.1.
 - 7: Retrieve the interval $X_k^{(s)}$ from the resulting CG-rep by neglecting its equality constraints.
 - 8: If $e^{(s)} < \varepsilon$, return $X_k^{(s)}$ and terminate. Otherwise, assign $s = s + 1$ and go to 4.
-

Remark 10.1. Step 7 in Algorithm 10.2 is equivalent to computing the interval hull of the set obtained by Method 3.3.

Remark 10.2. The interval provided by Algorithm 10.2 may not necessarily correspond to the interval hull of the solution of the linear algebraic equations (10.9). However, these are still accurate enclosures of the solution set, as illustrated in Section 10.5.

10.3.4 Final enclosure with constrained zonotopes

The interval bounds obtained by ISAR can be tight, but due to the wrapping effect inherent to intervals, these are not able to capture the dependencies between all the state variables. Therefore, we consider a final step in the state estimation method using constrained zonotopes to obtain a more accurate enclosure that takes into account these dependencies. The complete algorithm is denoted by ISAR-CZ. For the sake of completeness, we also consider applying this final step to the IHISE, with the corresponding algorithm denoted by IHISE-CZ.

Consider the slope approximation (10.9), the augmented vector $\mathbf{x}_k = (\mathbf{q}_k, \mathbf{h}_k)$ and the sets satisfying $(\mathbf{q}_k, \mathbf{h}_k) \in Q_k^{(\kappa_l)} \times H_k^{(\kappa_l)} = X_k^{(\kappa_l)}$, where $\kappa_l \triangleq \kappa_m$ or $\kappa_l \triangleq \kappa_s$ if the enclosure was obtained by IHISE or by ISAR, respectively. Let $\tilde{\mathbf{u}}_k = \mathbf{u}_k - \boldsymbol{\beta}_+ - \boldsymbol{\Delta}(\boldsymbol{\beta}_+ + \boldsymbol{\Lambda}_- \mathbf{q}_k)$, with $\mathbf{u}_k \in U_k$. Then, it is true that $\tilde{\mathbf{u}}_k \in \check{U}_k$, where \check{U}_k is a CZ given by

$$\check{U}_k \triangleq U_k \oplus (-\boldsymbol{\beta}_+) \oplus \triangleleft(\mathbf{D}, \boldsymbol{\beta}_- \oplus \boldsymbol{\Lambda}_- Q_k^{(\kappa_l)}),$$

with \mathbf{D} being a diagonal interval matrix satisfying $D_{ii} = B_\infty^{n_q}$ for $i = 1, 2, \dots, n_q$, such that $\boldsymbol{\Delta} \in \mathbf{D}$, and the operator $\triangleleft(\cdot, \cdot)$ is the CZ-inclusion operator defined in Theorem 4.1. The final enclosure used for state estimation is a CZ outer approximation of the solution set of (10.9), and is denoted by \hat{X}_k . For $\mathbf{u}_k \times \mathbf{d}_k \in U_k \times D_k$, this CZ is obtained by the generalized intersection

$$\hat{X}_k = \{\mathbf{x} \in X_k^{(\kappa_l)} : \mathbf{F}\mathbf{x} \in \check{U}_k \times D_k\} = X_k^{(\kappa_l)} \cap_{\mathbf{F}} (\check{U}_k \times D_k), \quad (10.14)$$

where $\mathbf{F} = \begin{bmatrix} \boldsymbol{\Lambda}_+ & \mathbf{B} \\ \mathbf{B}^T & \mathbf{0} \end{bmatrix}$.

In this chapter, we take advantage of the state coupling resulting from the nonlinear equality constraints (10.1), which are captured by the constrained zonotope \hat{X}_k given by (10.14). Moreover, since $Q_k^{(\kappa_l)}$, $H_k^{(\kappa_l)}$, D_k , and U_k , are all intervals (hence have n_q , n_h , n_h , and n_q generators, respectively), the resulting enclosure \hat{X}_k has $3n_q + 2n_h$ generators and $n_q + n_h$ constraints. Finally, to avoid numerical issues in \hat{X}_k when one or both the sets U_k and D_k are degenerate, we add tolerance intervals $\{\tau_u, 0\}$ and $\{\tau_d, 0\}$ to each of the corresponding degenerate dimensions in U_k and D_k , respectively.

Remark 10.3. Although all the sets $Q_k^{(\kappa_l)}$, $H_k^{(\kappa_l)}$, U_k and D_k , are intervals (and therefore zonotopes), constrained zonotopes are required to compute (10.14) because the generalized intersection (3.10) is conservative and difficult to enclose using zonotopes (Bravo et al., 2006).

Remark 10.4. In this chapter, the state estimation is assumed to be performed offline, using only the information obtained from physical constraints and historical data of the WDN. However, if measurements of \mathbf{q}_k and \mathbf{h}_k are available, these can be used to reduce the conservativeness of the initial bounds $Q_k^{(0)}$ and $H_k^{(0)}$ by intersecting them with the

measurement bounds. This procedure can be done prior to computing the IHISE-CZ and ISAR-CZ, and will be explored in future work.

10.4 Passive leak detection in WDNs

Consider a WDN in the presence of leakages, with n_q links and n_h nodes, described by the nonlinear algebraic equation (10.1a), and

$$\mathbf{B}^T \mathbf{q}_k = \mathbf{d}_k + \boldsymbol{\ell}_k, \quad (10.15)$$

where $\mathbf{q}_k \in \mathbb{R}^{n_q}$ are the water flows at the links, $\mathbf{B} \in \mathbb{R}^{n_q \times n_h}$ is the incidence matrix, and $\mathbf{d}_k \in \mathbb{R}^{n_h}$ is the water demand by the consumers at the nodes. In (10.15), $\boldsymbol{\ell}_k$ denotes the presence of pressure-dependent leakages modeled in this chapter at the nodes (Greyvenstein & van Zyl, 2007). The purpose of leak detection is to check whether $\ell_{i,k} \neq 0$ for at least one node i .

Consider the output equation

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{v}_k, \quad (10.16)$$

where $\mathbf{v}_k \in \mathbb{R}^{n_y}$ is the measurement uncertainty, and $Y_k \triangleq [\mathbf{y}_k^L, \mathbf{y}_k^U]$ is the obtained measurement interval, with $\text{mid}(Y_k) = \mathbf{y}_k$. In this chapter, this condition can be verified conservatively² by checking if the following relation holds:

$$\mathbf{C}\hat{X}_k \cap Y_k = \emptyset, \quad (10.17)$$

in which \hat{X}_k is the set obtained by state estimation IHISE-CZ and ISAR-CZ without considering leakage. In the case of IHISE and ISAR, \hat{X}_k is the interval provided by Algorithms 10.1 and 10.2, respectively.

Despite being conservative, the advantage in verifying if (10.17) holds is that no false positives can occur during leakage detection, given that the bounds on the uncertain parameters are valid, i.e., $\mathbf{r} \in R$, $\mathbf{u}_k \in U_k$ and $\mathbf{d}_k \in D_k$, for all $k \geq 0$. The resulting constrained zonotope from the left-hand-side in (10.17) can be computed using (3.8) and (3.10). Property 3.1 can be used to verify if this constrained zonotope is empty by solving one LP.

²Since \hat{X}_k is computed conservatively, (10.17) $\implies \ell_{i,k} \neq 0$ for at least one i , but the converse is not true in general.

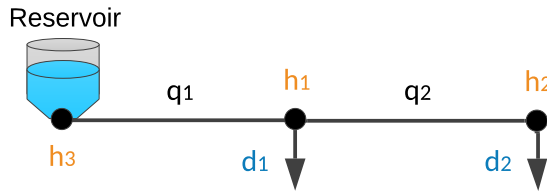


Figure 10.1: The two-pipe network example.

10.5 Numerical experiments

In this section, we show the efficiency and accuracy of the proposed CZ based methods (IHISE-CZ and ISAR-CZ) for state estimation of WDNs. In addition, we show how these methods can be employed for leak detection (we consider scenarios where the leakage is present for the entire simulation) and compare their performance with the interval-based methods IHISE and ISAR. For these latter, the convergence tolerance (10.7) was set to $\varepsilon = 1$.

10.5.1 Two-pipe example

In this first example we consider a network composed of 3 nodes and 2 pipes (see Figure 10.1). A 24-hour simulation was performed with EPANET (Rossman, 2000), using conditions (water demands and reservoir levels) that vary every half of an hour. We consider that the physical parameters $\mathbf{r} \in R$ have uncertainty of 5%, the demands $\mathbf{d}_k \in D_k$ are measured with uncertainty of 5%, and the reservoir level \mathbf{u}_k is precisely known. The tolerance values adopted in Section 10.3.4 are set to $\tau_u = 1.0 \times 10^{-4}$ m and $\tau_d = 1.0 \times 10^{-4}$ m³/h.

Figure 10.2 shows the volumes of the sets³ \hat{X}_k , obtained using IHISE, ISAR, IHISE-CZ, and ISAR-CZ, for time $k \in [0, 24]$ in hours, in intervals of 30 minutes each. Figure 10.2 also reports the volumes of ‘MCS interval’ and ‘MCS convex hull’. The former provides an interval approximation of the solution of (10.1) obtained performing 1000 MCS, considering parameters \mathbf{r} and demands \mathbf{d}_k drawn randomly from their corresponding uncertainty bounds. The latter is obtained by computing the convex hull of the same 1000 trajectories. Note that both results, which are based on a finite number of samples, represent only an inner approximation of the solution of (10.1) but are useful as a reference for the set-based state estimation methods.

Table 10.1 shows the execution times of IHISE, ISAR, IHISE-CZ, and ISAR-CZ, averaged over the time horizon $k \in [0, 24]$, together with the number of iterations performed

³For the sake of simplicity, in this section we use the slight abuse of notation $\hat{X}_k \triangleq X_k^{(\kappa_I)}$ for the sets obtained using IHISE and ISAR.

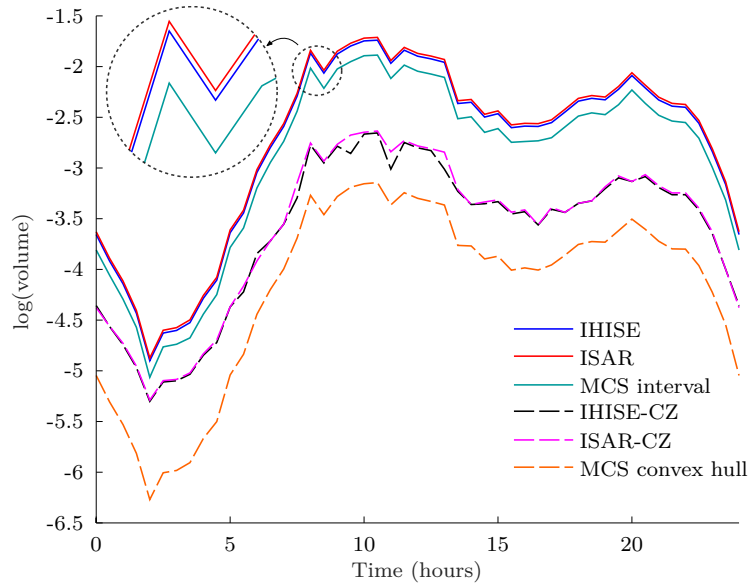


Figure 10.2: The volumes of the sets \hat{X}_k obtained using IHISE, IHISE-CZ, ISAR, ISAR-CZ, and also by the interval hull and convex hull of the trajectories resulting from 1000 Monte-Carlo simulations (MCS) using EPANET for the two-pipe network.

by the IHISE and ISAR for the given convergence tolerance ε . Note that the average execution time of ISAR is only 48.6% of the IHISE, even if a higher number of iterations is required. As expected, the volumes of the intervals/CZs provided by ISAR/ISAR-CZ are slightly bigger than the ones provided by IHISE/IHISE-CZ. Nevertheless, as shown in Figure 10.2, this increase is reasonable, demonstrating that the drawback of using ISAR/ISAR-CZ is mild while the advantage in terms of execution time is significant.

In addition, the volumes of both the outer convex approximations given by the CZs in IHISE-CZ and ISAR-CZ are smaller than the interval approximations provided by IHISE and ISAR (Figure 10.2), illustrating that both IHISE-CZ and ISAR-CZ can be significantly less conservative. Note that the volumes of the CZs are smaller even than MCS interval, demonstrating that CZs are indeed capable of capturing the existing coupling between the state variables. This effect is highlighted in Figure 10.3, which shows the projections of \hat{X}_0 onto (q_1, q_2, h_1) obtained using ISAR-CZ, the sets corresponding to MCS interval and MCS convex hull, and also the MCS trajectories.

Table 10.1: Average execution times and number of iterations per time step in the two-pipe example.

	Average execution times (seconds)	Number of iterations (#)
IHISE	0.0243	2
ISAR	0.0118	3
IHISE-CZ	0.0286	2
ISAR-CZ	0.0167	3

In the following, we consider a scenario with the presence of a leakage at node 1. For illustration purposes, it is considered that all the water flows and hydraulic heads are measured, i.e., \mathbf{C} in (10.16) is the identity matrix. Specifically, the flows at pipes 1

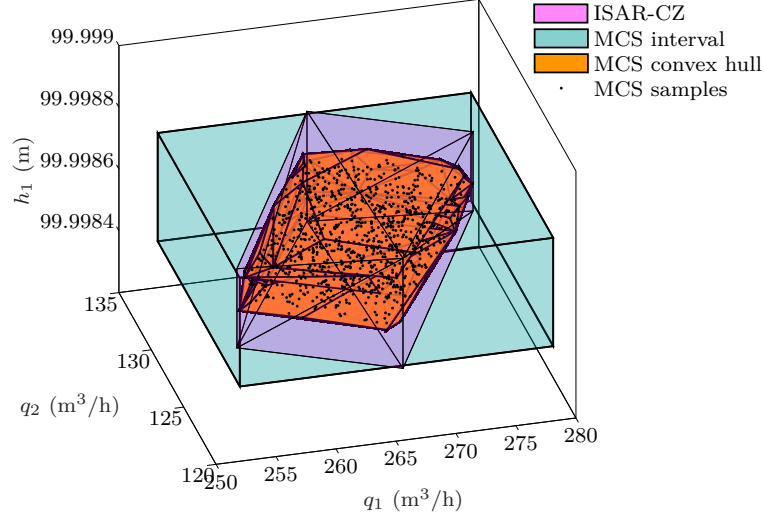


Figure 10.3: The projections of \hat{X}_0 onto (q_1, q_2, h_1) , obtained using ISAR-CZ, the intervals computed component-wise by maximum and minimum values obtained from 1000 MCS, as well as the convex hull and samples of the corresponding trajectories.

and 2 and the heads at nodes 1 and 2 are all measured with uncertainty of 2%, i.e., the measurement uncertainty in (10.16) is bounded by $\mathbf{v}_k \in V_k$, with $V_{k,i} = [-2\%|y_{k,i}|, 2\%|y_{k,i}|]$.

To generate a leakage at node i , this leakage is modelled by $\ell_{i,k} = \theta_i(h_{i,k} - \eta_i)^\phi$, where θ_i and ϕ denote the leak emitter coefficient and exponent respectively (Greyvenstein & van Zyl, 2007), which are completely unknown to the fault detection methodologies. The leakage emitter exponent for all the leakages is given by $\phi = 0.5$. The emitter coefficient θ_i for a leakage at node i is given based on the average system inflow of the WDN (i.e., the sum of all nominal demands $d_{i,k}$ averaged over a given time horizon $k \in [0, N]$), denoted by ρ . Then, to generate $\ell_i \neq 0$, the emitter coefficient θ_i is set to

$$\theta_i = \frac{\alpha\rho}{(\bar{h}_i - \eta_i)^\phi}$$

where \bar{h}_i is the nominal head of node i averaged over $k \in [0, N]$, and $\alpha \in [0, 100]\%$ is a parameter defining the magnitude of the leakage in comparison to the average total inflow ρ , referred as the leak magnitude.

Figures 10.4 and 10.5 show the leak detection rate (a successful individual detection is denoted by 1, while 0 corresponds to the absence of detection), averaged over 100 MCS. The detection rates shown in Figures 10.4 and 10.5 are further averaged over the leak magnitude range $\alpha = 1, 2, \dots, 15\%$, and the time interval $k = 0, 1, \dots, 24$ hours, respectively. In this example, IHISE-CZ and IHISE provided exactly the same detection rates of ISAR-CZ and ISAR, respectively. Moreover, IHISE-CZ and ISAR-CZ provide up to about 20% more detections than IHISE and ISAR, demonstrating the improvements achieved in leak detection when using a set representation capable of taking into account the dependencies between water flows and hydraulic heads. This feature is highlighted in Figure 10.6, which

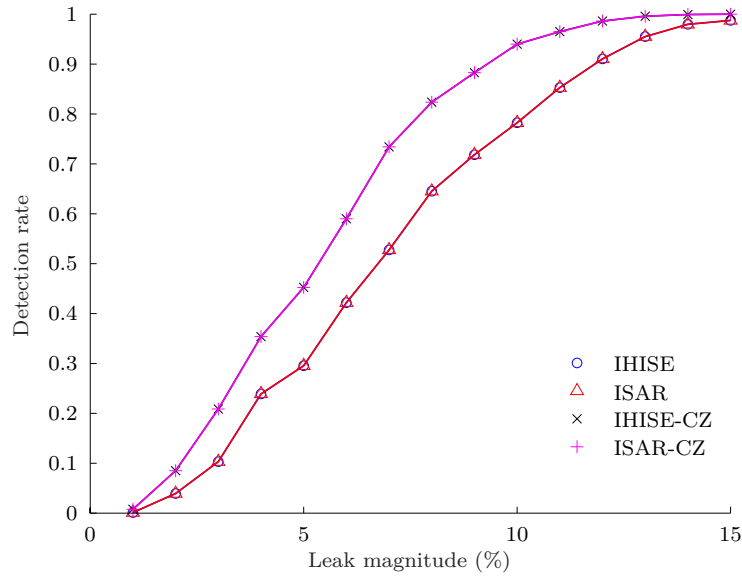


Figure 10.4: Leak detection rate of the IHISE, ISAR, IHISE-CZ, and ISAR-CZ for the two-pipe network (leak at node 1), considering different leak magnitudes, averaged over 100 random simulations and the time interval $k \in [0, 24]$.

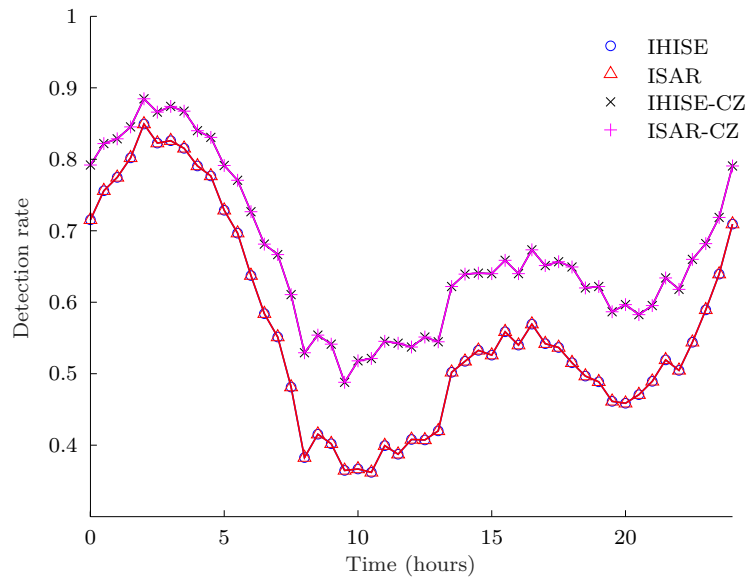


Figure 10.5: Leak detection rate of the IHISE, ISAR, IHISE-CZ, and ISAR-CZ for the two-pipe network (leak at node 1) over time, averaged over 100 random simulations and the leak magnitudes in the interval $[1, 15]\%$.

shows the projection of the sets \hat{X}_k provided by IHISE, ISAR-CZ, onto the water flows $(q_{1,k}, q_{2,k})$, as well as the corresponding projection of the measurement set Y_k , for time $k = 2$, in one of the 100 MCS considering leakage magnitude of 4%. This illustrates a case in which the leak has been successfully detected by the ISAR-CZ but not by the IHISE, which can be verified by the fact that the intersection of Y_k with the CZ provided by ISAR-CZ is empty, while the intersection with the interval provided by IHISE is not empty.

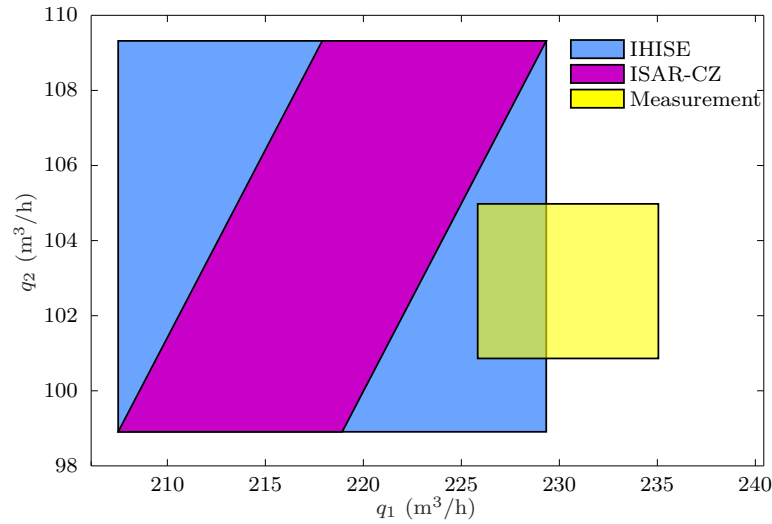


Figure 10.6: The projection onto $\mathbf{q}_k \in \mathbb{R}^2$ of the sets \hat{X}_k obtained using IHISE and ISAR-CZ, together with the measured flow interval for \mathbf{q}_k at time $k = 2$ in one of the 100 random simulations considering leakage at node 1 of 4% of the average nominal inflow in the two-pipe example.

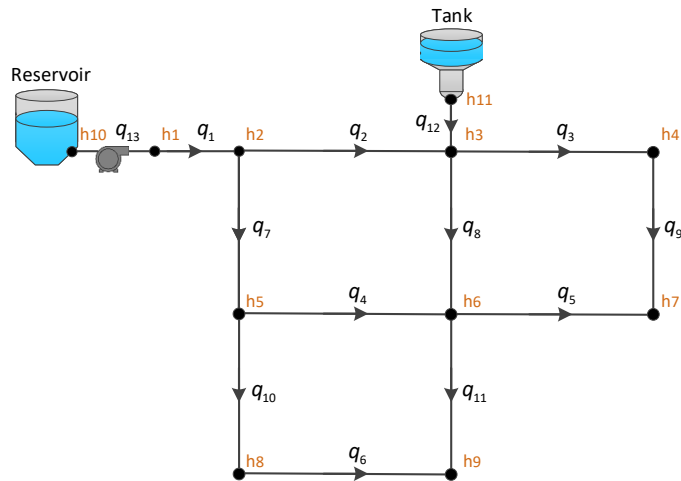


Figure 10.7: The benchmark network “Net1” used here as an example of a transport network.

10.5.2 Transport network

The second example considers the benchmark network ‘Net1’, which is provided by EPANET, and is illustrated in Figure 10.7. This network is composed of 9 nodes and 13 pipes, and it is used here as an example of a real transport network. In the following, we consider that pipe parameters $\mathbf{r} \in R$ have uncertainty of 5%, the water demands at nodes $\mathbf{d}_k \in D_k$, which are inflows to DMAs, are measured with uncertainty of 5%, and the reservoir and tank levels \mathbf{u}_k are precisely known. The tolerance values adopted in Section 10.3.4 are set to $\tau_u = 0.1$ m and $\tau_d = 0.1$ m³/h.

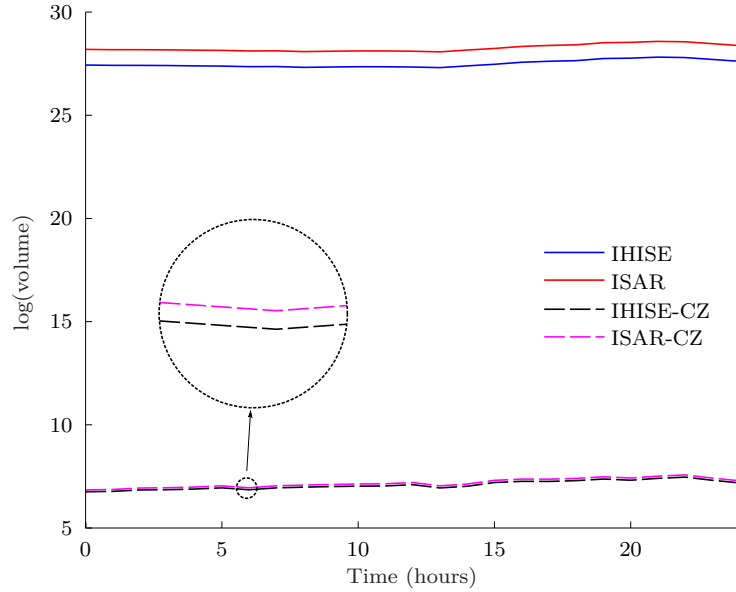


Figure 10.8: The volumes of the sets \hat{X}_k obtained using IHISE, IHISE-CZ, ISAR, and ISAR-CZ for the transport network ‘Net1’.

Figure 10.8 shows the volumes of the sets⁴ \hat{X}_k obtained using IHISE, ISAR, IHISE-CZ, ISAR-CZ, for time $k \in [0, 24]$. As in the previous example, for this transport network the volumes of the CZs generated by both IHISE-CZ and ISAR-CZ are significantly smaller than the volumes of the intervals provided by IHISE and ISAR, demonstrating that IHISE-CZ and ISAR-CZ can be significantly less conservative than IHISE and ISAR. Note also that the volumes of enclosures provided by IHISE-CZ and ISAR-CZ are very similar, showing that, also in this case, there is almost no drawback in terms of accuracy when using ISAR instead of IHISE. The corresponding execution times averaged over the time horizon $k \in [0, 24]$ are shown in Table 10.2, together with the number of iterations performed by the IHISE and ISAR. In this case, the average execution time of ISAR, which requires a higher number of iterations than IHISE, is only 7.4% of the time of IHISE.

Table 10.2: Average execution times and number of iterations per time step for the ‘Net1’ transport network.

	Average execution times (seconds)	Number of iterations (#)
IHISE	0.6364	7–8
ISAR	0.0470	9
IHISE-CZ	0.6410	7–8
ISAR-CZ	0.0516	9

We now assume that the flows at pipes 12 and 13, and the pressure at node 6 are measured, i.e., $\mathbf{y}_k = (q_{k,12}, q_{k,13}, h_{k,6}) + \mathbf{v}_k$, where the measurement uncertainty is bounded

⁴In this example, the fact that the sets \hat{X}_k are in \mathbb{R}^{22} results in intractable computational cost in obtaining the exact volumes of the CZs. Therefore, we show instead the volumes of the parallelotopes obtained by eliminating all the constraints in \hat{X}_k followed by generator reduction, using the complexity reduction methods presented in Section 3.3. The volumes are computed using the formula in Chisci et al. (1996).

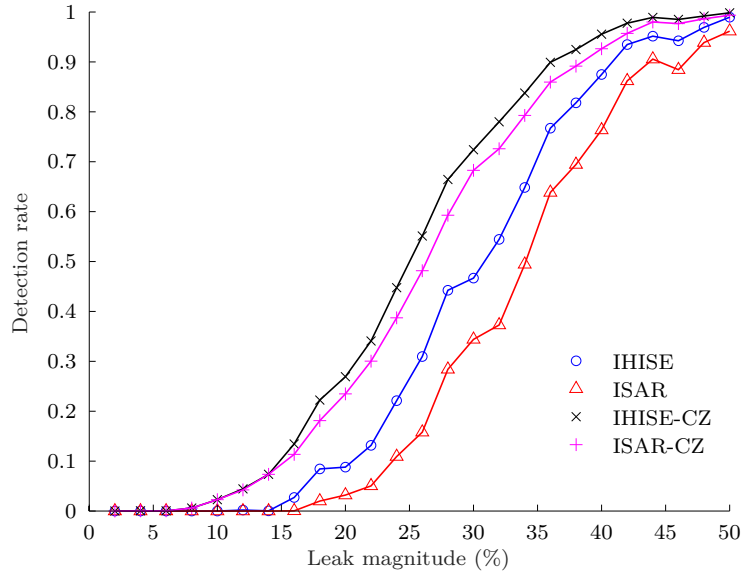


Figure 10.9: Leak detection rate of the IHISE, ISAR, IHISE-CZ, and ISAR-CZ for the transport network ‘Net1’ (leak at node 8), considering different leak magnitudes, averaged over 100 random simulations and the time interval $k \in [0, 24]$.

by $\mathbf{v}_k \in V_k$, with $V_{k,i} = [-2\%|y_{k,i}|, 2\%|y_{k,i}|]$. Two leakage scenarios are considered, with the presence of leakage at node 8 and node 4, respectively.

Figures 10.9 and 10.10 show the leak detection rate for the first scenario, averaged over 100 MCS, and further over the leak magnitude range $\alpha = 2, 4, \dots, 50\%$, and the time interval $k = 0, 1, \dots, 24$ hours, respectively. In this example, IHISE-CZ provides up to about 20% more detections than IHISE, demonstrating once again the advantages of using CZs instead of intervals for leak detection of WDNs. In addition, Figures 10.11 and 10.12 show the leak detection rate for the second scenario, averaged over 100 MCS, and further over the leak magnitude range $\alpha = 2, 4, \dots, 50\%$, and the time interval $k = 0, 1, \dots, 24$ hours, respectively. In the second scenario, once again the IHISE-CZ provides up to about 20% more detections than IHISE.

10.5.3 CY-DMA network

The third example considers a realistic DMA network from a large water utility in Cyprus, denoted by ‘CY-DMA’ (Vrachimis et al., 2020). The network is composed of 90 nodes and 130 pipes and is illustrated in Figure 10.13. In this example, we consider that the physical parameters $\mathbf{r} \in R$ have uncertainty of 5%, the node demands $\mathbf{d}_k \in D_k$, which represent groups of residential consumers, are measured with uncertainty of 10%, and the reservoir level \mathbf{u}_k is precisely known. The tolerance values adopted in Section 10.3.4 are set to $\tau_u = 1.0 \times 10^{-4}$ m and $\tau_d = 1.0 \times 10^{-4}$ m³/h. Figure 10.14 shows the volumes of the sets⁵ \hat{X}_k obtained using IHISE, ISAR, IHISE-CZ, ISAR-CZ, for time $k \in [0, 24]$. For this

⁵As in the previous example, for the CZs we show the volumes of the parallelotopes obtained by eliminating all the constraints in \hat{X}_k followed by generator reduction.

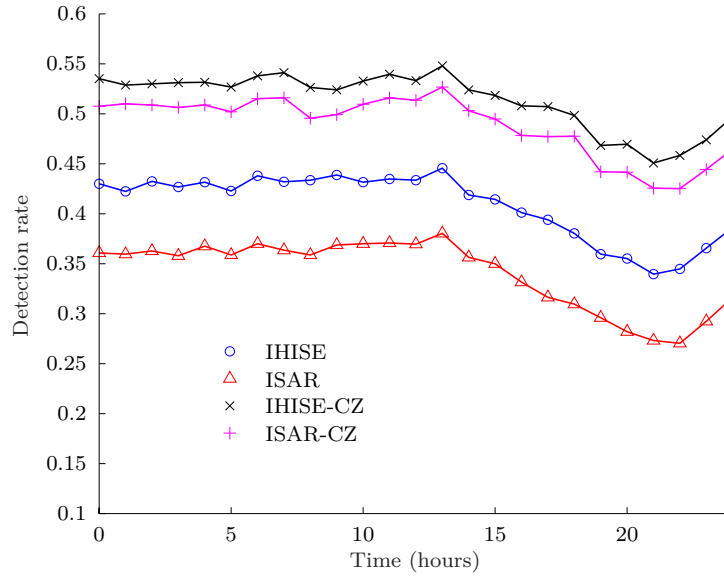


Figure 10.10: Leak detection rate of the IHISE, ISAR, IHISE-CZ, and ISAR-CZ for the transport network ‘Net1’ (leak at node 8) over time, averaged over 100 random simulations and the leak magnitudes in the interval $[2, 50]\%$.

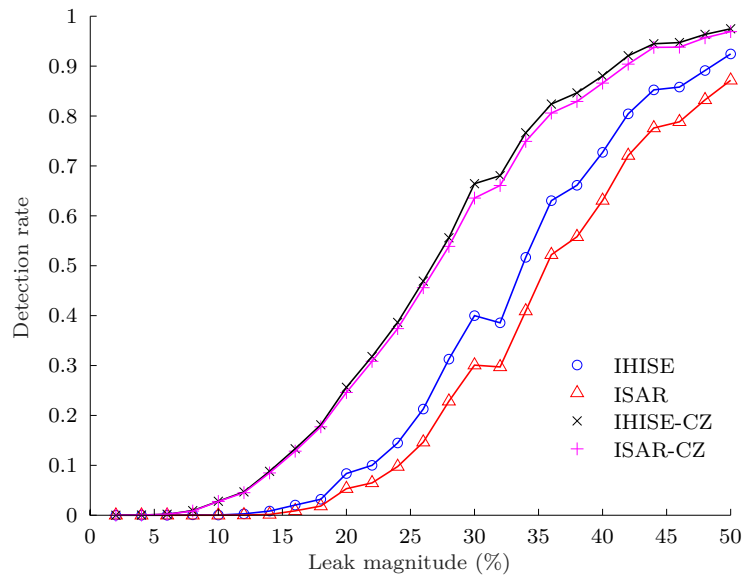


Figure 10.11: Leak detection rate of the IHISE, ISAR, IHISE-CZ, and ISAR-CZ for the transport network ‘Net1’ (leak at node 4), considering different leak magnitudes, averaged over 100 random simulations and the time interval $k \in [0, 24]$.

realistic network, the volumes of the constrained zonotopes generated by both IHISE-CZ and ISAR-CZ are, once again, significantly smaller than the bounds provided by IHISE and ISAR. Note that, due to the dimensionality of the CY-DMA network, considering, for instance, a rectangular set with uncertainty of 0.1 would result in $\log(\text{volume}) = -220$. Taking this into account, in particular the IHISE-CZ is capable of providing sets with about $\log(\text{volume}) = -500$. The corresponding execution times averaged over $k \in [0, 24]$ are shown in Table 10.3, together with the number of iterations performed by the IHISE and ISAR. Note that the average execution time of ISAR, which requires a higher number

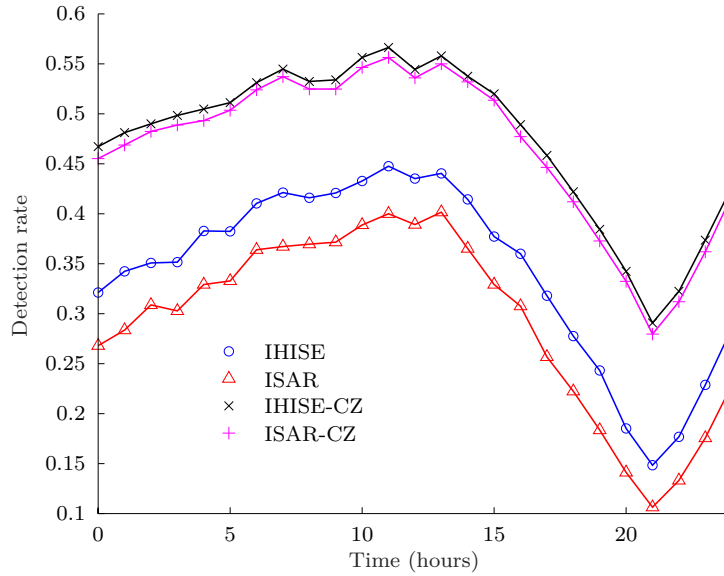


Figure 10.12: Leak detection rate of the IHISE, ISAR, IHISE-CZ, and ISAR-CZ for the transport network ‘Net1’ (leak at node 4) over time, averaged over 100 random simulations and the leak magnitudes in the interval $[2, 50]\%$.

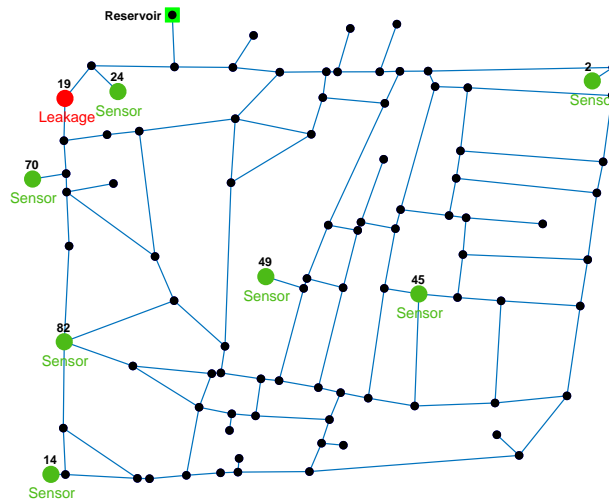


Figure 10.13: The ‘CY-DMA’ network, a realistic example of a district metered area network.

of iterations than IHISE, is only 10.7% of the time of IHISE.

Table 10.3: Average execution times and number of iterations per time step for the CY-DMA network.

	Average execution times (seconds)	Number of iterations (#)
IHISE	54.5293	15–16
ISAR	5.8429	34–37
IHISE-CZ	54.5549	15–16
ISAR-CZ	5.8670	34–37

We now consider a leakage scenario, with the presence of leakage at node 19. It is

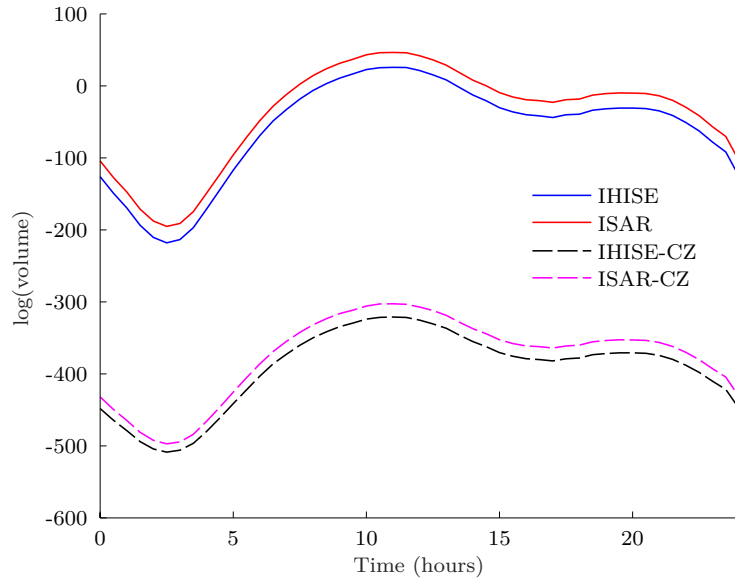


Figure 10.14: The volumes of the sets \hat{X}_k obtained using IHISE, IHISE-CZ, ISAR, and ISAR-CZ for the CY-DMA network.

considered that the inlet flow (pipe 14) and pressure (node ‘Reservoir’) are measured. Moreover it is considered that the water utility has installed pressures sensors at nodes 2, 14, 24, 45, 49, 70, and 82, with a total of 7 pressure sensors. The sensors’ positions were chosen using a sensor placement procedure which calculates the sensitivity matrix of the WDN and, using a heuristic optimization approach, computes the sensor locations that maximize the collective sensitivity of all sensors to all possible leakages (Casillas et al., 2013). The measured flows and pressures have uncertainties of 2% and 0.35%, respectively.

Figures 10.15 and 10.16 show the leak detection rate, averaged over 100 MCS, and further over the leak magnitude range $\alpha = 1, 2, \dots, 25\%$, and the time interval $k = 0, 1, \dots, 24$ hours, respectively. In this scenario, IHISE-CZ could provide up to about 10% more detections than IHISE. In this scenario, it is interesting to note that the advantage of using CZs instead of intervals is more expressive for leakage detection during midday (12:00-13:00) rather than night hours (3:00-4:00). This realistic network uses typical residential demand patterns with peak during midday, and minimum flow demands during night. The high pressure conditions resulting from the low demands during night hours cause the leakage magnitude to increase, which makes the leakage more evident to the leakage detection methodologies. The ability of CZs to consider the dependencies between state variables results in the interesting capability of providing higher detection rates even at peak demand hours, which reduces the time of detection and water loss. Finally, a few different leakage scenarios have been simulated taking into account several additional pressure sensors in the CY-DMA. However, due to the optimal placement of pressure sensors, the improvement in leak detection of all the methods was mild, which do not cope with the cost of placing additional sensors in a realistic network, and for this reason were not reported.

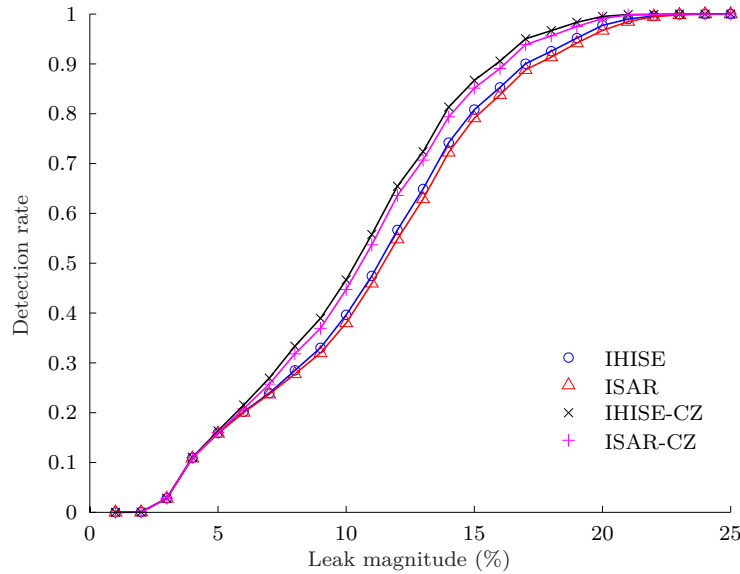


Figure 10.15: Leak detection rate of the IHISE, ISAR, IHISE-CZ, and ISAR-CZ for the CY-DMA network (leak at node 19), considering 7 pressure sensors, different leak magnitudes, averaged over 100 random simulations and the time interval $k \in [0, 24]$.

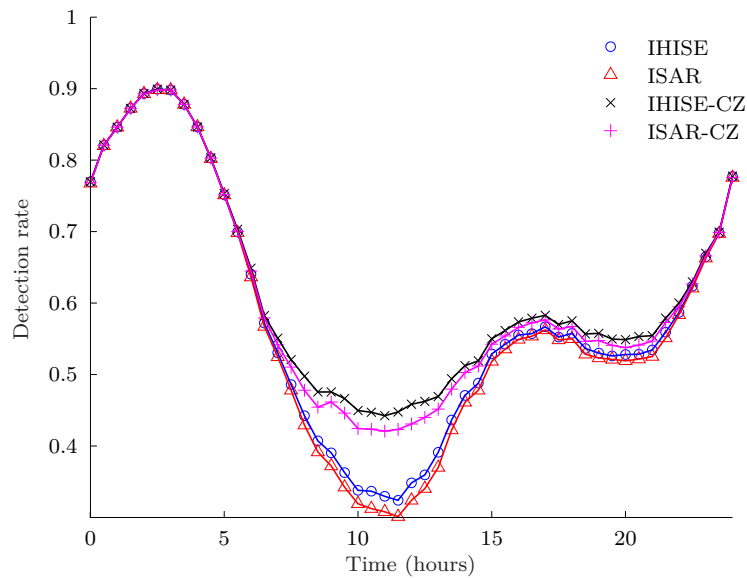


Figure 10.16: Leak detection rate of the IHISE, ISAR, IHISE-CZ, and ISAR-CZ for the CY-DMA network (leak at node 19) over time, considering 7 pressure sensors, averaged over 100 random simulations and the leak magnitudes in the interval $[1, 25]\%$.

10.6 Final remarks

This chapter proposed two new methods for set-based state estimation of WDN based on CZs, namely IHISE-CZ and ISAR-CZ. With respect to interval-based methods, CZ-based ones are able to capture the dependencies between hydraulic states, resulting in sets with significantly smaller volumes. The benefits of IHISE-CZ and ISAR-CZ were also demonstrated for leakage detection in WDNs. In particular, the simulation results on a benchmark transport network and on a realistic DMA network show an increase in detection

rates of up to 20% when using the proposed methods, while also requiring significantly less computational resources to compute the enclosures. Moreover, it is observed that the proposed methods perform well during peak demand hours, thus reducing significantly the time-to-detection and the cost of water loss for water utilities.

11

Application: Li-Ion batteries

In this chapter, we propose an interval state estimation of a lithium-ion cell when using an Equivalent Circuit Model (ECM), taking into account bounded parametric uncertainty and measurement noise. Battery management systems (BMSs) are responsible for controlling and monitoring battery operations. In order to be effective, BMSs rely on mathematical models. The parameters of these latter are usually obtained through an identification procedure which provides limited accuracy. In addition, only cell current and voltage are usually measurable, and therefore, the model states need to be estimated. Simulation results show the suitability of the proposed interval estimator to provide a tight enclosure of the states, which is essential for fault detection and model-based control of a lithium-ion cell. The results of this chapter were obtained in collaboration with the PhD candidate Diego Locatelli, supervised by Prof. Davide M. Raimondo. The content has been presented in [Locatelli et al. \(2021\)](#).

This chapter is organized as follows. In Section 11.2, the Single Particle Model with electrolyte dynamics (SPMe) and ECM models are described. The identification procedure and set-based state estimation are presented in Sections 11.3 and 11.4, respectively. The results are reported in Section 11.5 and, finally, Section 11.6 concludes the chapter.

11.1 Introduction

Lithium-ion batteries have become of particular interest due to their use in many commercial sectors, including portable electronic devices, hybrid and electric vehicles, and grid energy

storage. High cell voltage, cycling durability, high energy, and power density, are the main key factors that make Lithium-ion the best trade-off between costs, performance and efficiency, when compared to other battery types on the market.

Battery management systems aim to make the battery safer, reliable and efficient. As shown in [Lu et al. \(2013\)](#), violation of temperature, current and voltage restrictions, may result in lower performance as well as safety issues. Advanced BMSs rely on mathematical models in order to improve efficiency and monitor the battery cell, by estimating fundamental quantities such as the State of Charge (SOC). Mathematical models for Lithium-ion battery dynamics fall within two main categories: Electrical Circuit Models and Electrochemical Models (EMs). EM are very accurate and, therefore, very useful when high fidelity is required, such as in simulations. They are generally very complex models, and this affects the computational cost. On the other hand, ECMs are simpler models, and they are used in real-time control system applications especially for online SOC and state estimation.

Look-up table methods are used for SOC estimation exploiting its relations with other measurable parameters as the Open Circuit Potential (OCP), but they are not suitable for run-time operations ([Xiong et al., 2017](#)). This is due to the fact that the OCP can be measured only in stationary conditions. Coulomb counting is also one of the main tools used for SOC estimation ([Piller et al., 2001](#)), but it is not completely accurate because of drift errors due to disturbances on the current measurement and uncertainty on the cell capacity. Moreover, while SOC estimation is important, in order to employ a model-based controller, full state estimation is required.

Several stochastic approaches on state estimation have been applied in the battery field, including Extended Kalman Filtering ([Di Domenico et al., 2010](#); [Bizeray et al., 2015](#)), sliding-mode observer ([Kim, 2009](#)), Sigma-point Kalman Filter ([Plett, 2006](#)), Particle filtering ([Tulsyan et al., 2016](#)), and Moving Horizon Estimator ([Hu et al., 2018](#)). The stochastic estimation approaches assume that probabilistic distributions of the uncertainties are known. On the other side, set-based estimation considers unknown-but-bounded uncertainties. When physical bounds are available (as for the case of ECM parameters), this second option is more reliable since knowing the exact distribution of the uncertainties is rarely the case in practice. To the best of our knowledge, set-based estimation was used within the battery context only in [Zhang et al. \(2020a\)](#). This latter proposes a continuous-time ECM-based interval observer for SOC estimation in a parallel connected Li-ion battery pack. The proposed observer assumes that continuous-time measurements are available, which is not true in practice since sensors have finite sampling rates. Moreover, the coupling between parameters is not considered, which is usually present when an identification procedure is performed.

In this chapter, we develop a discrete-time interval observer for a single Li-ion cell based on the forward-backward method described in [Jaulin et al. \(2001a\)](#). In particular,

parametric uncertainties are considered and obtained by performing an identification procedure on data collected on a well known EM, i.e. the SPMe. The contributions of this chapter are: (i) the identification of ECM parameter bounds based on the Fisher Information Matrix, (ii) a discrete-time interval state estimation method based on inclusion functions and constraint propagation, which handles discrete-time measurement. Numerical experiments show that the proposed methodology is efficient and provides accurate enclosures for both the SOC and the electric state variables of the ECM.

11.2 Lithium-ion cell modeling

This section introduces the two models considered in this chapter. The SPMe, which is a well known EM, is used as the “real plant”. Its complete parameterization is obtained according to [Ecker et al. \(2015\)](#) and represents a Kokam SLPB 75106100 cell. Since the SPMe is computational expensive, for estimation purposes we adopt the Thévenin ECM instead. Its parametrization and uncertainty quantification are obtained relying on input-output data (current-voltage) collected from the SPMe.

11.2.1 Single Particle Model With Electrolyte dynamics

The SPMe is an approximation of the Doyle-Fuller-Newman (DFN) model proposed in [Doyle et al. \(1993\)](#), which predicts the charge/discharge behaviour of dual insertion cells. The model is general for cells utilizing two composite electrodes with active insertion material, electrolyte and inert conducting material. Cathode, separator and anode are the three sections composing the battery cell and are here denoted by $\{p, s, n\}$. In this chapter, the model described by [Pozzi et al. \(2020\)](#) is taken as a reference, but we assume the cell is kept in a climatic chamber at constant temperature (isothermal process). The single cell dynamics is described by a system of Ordinary Differential Equations (ODEs). A polynomial approximation of the solid ions concentration along the particle radius is used to reduce the Fick’s law for each electrode to an ODE (see [Subramanian et al. \(2005\)](#)). Moreover, the Finite Volume Method (FVM) is used to spatially discretize the Partial Differential Equations (PDEs) describing the electrolyte ions diffusion.

The dynamics of the SPMe is described by $3 + 3N$ electric state variables $[\bar{\theta}_p(t) \ \bar{q}_p(t) \ \bar{q}_n(t) \ c_{e,p}^{[1]}(t) \ \dots \ c_{e,n}^{[N]}(t)]$, where N is the number of discretization volumes for each section of the cell, $\bar{\theta}_p(t)$ is the cathode average stoichiometry, $\bar{q}_i(t)$ is the volume-averaged concentration flux for the i -th electrode, and $c_{e,j}^{[k]}(t)$ is the electrolyte concentration in the k -th volume of the j -th cell section. Note that indices j and i will be used to refer to the three cell sections $\{p, s, n\}$ and to the two electrodes $\{p, n\}$, respectively.

The dynamics of the cathodic average stoichiometry can be expressed as

$$\dot{\bar{\theta}}_p(t) = -\frac{\Delta\theta_p}{C_{\text{batt}}^S} I(t),$$

where C_{batt}^S is the nominal cell capacity, $\Delta\theta_i = \theta_i^{100\%} - \theta_i^{0\%}$ is the difference between the values of the stoichiometry at battery fully charged and fully discharged, and I is the current, input of the system¹. In order to avoid a further differential equation, the anodic average stoichiometry is computed according to the algebraic condition (see [Di Domenico et al. \(2010\)](#))

$$\bar{\theta}_n(t) = \theta_n^{0\%} + \frac{\bar{\theta}_p(t) - \theta_p^{0\%}}{\Delta\theta_p} \Delta\theta_n.$$

For each electrode, the volume-averaged concentration flux dynamics is described by

$$\dot{\bar{q}}_i = -30 \frac{D_{s,i}}{R_{p,i}^2} \bar{q}_i(t) - \frac{45\Delta\theta_i c_{s,i}^{\text{max}}}{6R_{p,i} C_{\text{batt}}^S} I(t),$$

where $D_{s,i}$ is the solid diffusion coefficient, $c_{s,i}^{\text{max}}$ is the maximum solid concentration, $R_{p,i}$ is the solid particle radius, and θ_i is the surface stoichiometry given by

$$\theta_i(t) = \bar{\theta}_i(t) + \frac{8R_{p,i} \bar{q}_i(t)}{35c_{s,i}^{\text{max}}} - \frac{R_{p,i}^2 \Delta\theta_i}{105D_{s,i} C_{\text{batt}}^S} I(t).$$

The ODEs describing the diffusion of the electrolyte concentration derive from the finite volume discretization of the original PDEs proposed in [Moura et al. \(2016\)](#). Each section of the cell is divided into N finite volumes. The k -th volume in the j -th section is centered in the spatial coordinate $x_{j,k}$ and ranges within the interval $[x_{j,k-\frac{1}{2}}, x_{j,k+\frac{1}{2}}]$, with width $\Delta x_j = L_j/N$, where L_j is the section thickness. For each volume, the electrolyte concentration dynamics is given by

$$\epsilon_j \frac{\partial c_{e,j}^{[k]}(t)}{\partial t} = \left[\frac{\tilde{D}_e(x)}{\Delta x_j} \frac{\partial c_{e,j}(x,t)}{\partial x} \right]_{x_{j,k-\frac{1}{2}}}^{x_{j,k+\frac{1}{2}}} + K(j), \quad (11.1)$$

with ϵ_j the material porosity, A the contact area between solid and electrolyte phases, F the Faraday constant, t_+ the transference number, and $\tilde{D}_e(x)$ the electrolyte diffusion coefficient, whose computation is explained more in detail in [Pozzi et al. \(2020\)](#). The term $K(j)$ is null for the separator section and $\mp(1-t_+)I(t)/FAL_n$ for cathode and anode respectively. The state of charge $z^S(t)$ can be computed according to the cathodic or anodic convention, as shown by the following relation

$$z^S(t) = \frac{\bar{\theta}_p(t) - \theta_p^{0\%}}{\Delta\theta_p} = \frac{\bar{\theta}_n(t) - \theta_n^{0\%}}{\Delta\theta_n}. \quad (11.2)$$

¹In this chapter, we adopt the convention that the battery is being charged when a negative current is applied.

Finally, the output terminal voltage is given by

$$V^s(t) \triangleq V_{oc}(t) + \bar{\eta}_p(t) - \bar{\eta}_n(t) + \Delta\Phi_e(t), \quad (11.3)$$

where $V_{oc}(t)$ is the overall open circuit potential defined as

$$V_{oc}(t) \triangleq U_p(t) - U_n(t), \quad (11.4)$$

with $U_p(t)$ and $U_n(t)$ the cathodic and anodic open circuit potentials, respectively. The expression of these latter is obtained using a fitting procedure on the experimental data provided in [Ecker et al. \(2015\)](#), as

$$U_p = 4.571 + 0.02414\theta_p - 7.8370\theta_p^2 + 8.07\theta_p^3 + 20.94\theta_p^4 - 40.7\theta_p^5 + 18.45\theta_p^6, \quad (11.5)$$

$$U_n = \frac{0.00694 + 0.1261\theta_n}{\theta_n^2 + 0.6995\theta_n + 0.00405}. \quad (11.6)$$

The overpotentials $\bar{\eta}_i(t)$ are defined, for each electrode, as $\bar{\eta}_i(t) = \frac{2RT}{F} \sinh^{-1}\left(\frac{\Delta\theta_i R_{p,i}}{6\bar{i}_{0,i}(t)C_{batt}^S} I(t)\right)$, in which the quantity $\bar{i}_{0,i}(t)$ is given by $\bar{i}_{0,i}(t) = Fk_i \sqrt{\bar{c}_{e,i}(t)\theta_i(t)(1-\theta_i(t))}$, where k_i is the rate reaction constant, and $\bar{c}_{e,i}(t)$ is the average electrolyte concentration in the i -th section. Returning to (11.3), $\Delta\Phi_e(t)$ is evaluated by

$$\Delta\Phi_e(t) = \Phi_e^{drop}(t) + \frac{2RT(t)}{F}(1-t_+) \log\left(\frac{c_{e,p}^{[1]}(t)}{c_{e,n}^{[N]}(t)}\right), \quad (11.7)$$

whose terms are already defined except for $\Phi_e^{drop}(t)$, which is approximated by $\Phi_e^{drop}(t) \simeq -(I(t)/2N)(\phi_p(t) + 2\phi_s(t) + \phi_n(t))$. The quantities $\phi_j(t)$, $j \in \{p, s, n\}$ depend on the electrolyte conductivity and are computed assuming a trapezoidal shape of the ionic current $i_e(x, t)$ along the spatial domain. We refer to [Pozzi et al. \(2020\)](#) for further details. The SPMe parameters used in this chapter are taken from [Ecker et al. \(2015\)](#) and are summarized in [Table 11.1](#).²

11.2.2 Equivalent Circuit Model

The ECM considered in this chapter for state estimation purposes is the Thévenin model depicted in [Figure 11.1](#). It consists of three parts: (i) the open circuit potential $V_{oc}(t)$; (ii) two internal resistors R_0 and R_1 ; and (iii) the capacitor C_1 . This latter is useful to describe the charge and discharge transient of the cell. The electrical behavior of the ECM

²Activation energies are used in Arrhenius's law for temperature-dependent parameters as $\psi(T) = \psi_0 e^{-\frac{E_{\alpha,\psi}}{RT}}$.

Table 11.1: SPMe cell section varying parameters.

Parameter	Value	Parameter	Value
$\theta_p^{0\%}$	0.86	L_p (m)	$54.5 \cdot 10^{-6}$
$\theta_n^{0\%}$	0.04	L_s (m)	$19 \cdot 10^{-6}$
$\theta_p^{100\%}$	0.26	L_n (m)	$73.7 \cdot 10^{-6}$
$\theta_n^{100\%}$	0.75	ϵ_p	1.5443
$R_{p,p}$ (m)	$6.49 \cdot 10^{-6}$	ϵ_s	1.7572
$R_{p,n}$ (m)	$8.7 \cdot 10^{-6}$	ϵ_n	1.6369
$c_{s,p}^{\max}$ ($\frac{\text{mol}}{\text{m}^3}$)	48580	k_p ($\frac{\text{m}^{5/2}}{\text{mol}^{1/2}}$)	$3.0496 \cdot 10^{-11}$
$c_{s,n}^{\max}$ ($\frac{\text{mol}}{\text{m}^3}$)	31920	k_n ($\frac{\text{m}^{5/2}}{\text{mol}^{1/2}}$)	$1.4656 \cdot 10^{-11}$
$D_{s,i}$ ($\frac{\text{m}^2}{\text{s}}$)	10^{-14}	$E_{\alpha_{k,p}}$ ($\frac{\text{J}}{\text{mol}}$)	43600
$E_{\alpha_{D_s,p}}$ ($\frac{\text{J}}{\text{mol}}$)	80600	$E_{\alpha_{k,n}}$ ($\frac{\text{J}}{\text{mol}}$)	53400
$E_{\alpha_{D_s,n}}$ ($\frac{\text{J}}{\text{mol}}$)	40800	$C_{\text{batt}}^{\text{S}}$ (A·h)	7.5
$D_{e,j}$ ($\frac{\text{m}^2}{\text{s}}$)	$2.4 \cdot 10^{-10}$	t_+	0.26
$E_{\alpha_{D_e,j}}$ ($\frac{\text{J}}{\text{mol}}$)	17100	A (m^2)	0.4121
R ($\frac{\text{J}}{\text{mol} \cdot \text{K}}$)	8.314472	F ($\frac{\text{s} \cdot \text{A}}{\text{mol}}$)	96485

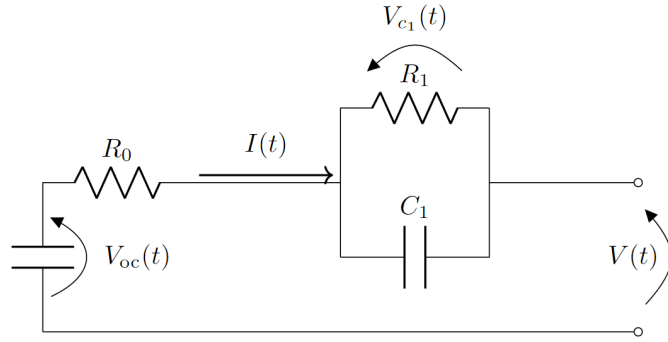


Figure 11.1: Thévenin equivalent circuit model of the Lithium-ion cell.

is derived from Kirchhoff and Ohm's laws as described by

$$\begin{cases} z^{\text{E}}(t) = -\frac{I(t)}{C_{\text{batt}}^{\text{E}}}, \\ \dot{V}_{c_1}(t) = -\frac{V_{c_1}(t)}{R_1(t)C_1(t)} + \frac{I(t)}{C_1(t)}, \end{cases} \quad (11.8)$$

where $z^{\text{E}}(t)$ is the cell SOC, $I(t)$ is the input current³, $V_{c_1}(t)$ is the voltage drop across the parallel $R_1 - C_1$, and $C_{\text{batt}}^{\text{E}}$ is the nominal cell capacity.

Remark 11.1. Since the ECM will only approximate the SPMe dynamics and its parameters will be estimated, there is no guarantee that battery capacity, SOC and OCP will have the exact same value/time behavior. For this reason, we distinguish them by using the superscripts E and S, respectively.

³Note that, also in this case, we adopt the convention that the battery is charged by a negative current.

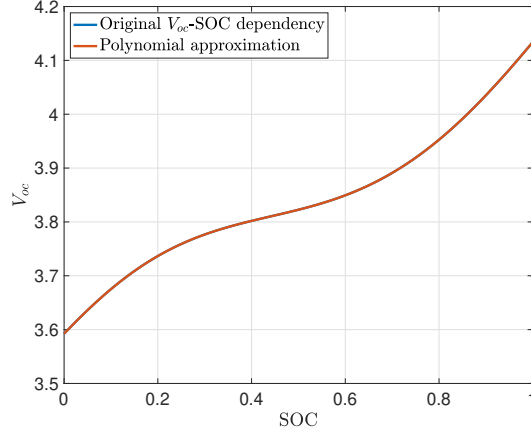


Figure 11.2: Polynomial approximation of OCP-SOC dependency.

In accordance with [Perez et al. \(2017\)](#), in order the ECM to better approximate the SPMe, R_1 and C_1 can be expressed as nonlinear functions of the SOC, given by

$$R_1(t) = R_{1,0} + R_{1,1}z^E(t) + R_{1,2}z^E(t)^2, \quad (11.9)$$

$$C_1(t) = C_{1,0} + C_{1,1}z^E(t) + C_{1,2}z^E(t)^2. \quad (11.10)$$

Finally, the output voltage equation can be computed as

$$V^E(t) = V_{oc}^E(t) - R_0 I(t) - V_{c_1}(t), \quad (11.11)$$

where $V_{oc}^E(t)$ is the overall open circuit potential. Unlike (11.4), this latter is here expressed as a function of the SOC, i.e. $z^E(t)$. This is necessary since θ_p and θ_n are not states of (11.8). However, since the OCP represents the potential difference between the negative and positive electrodes when no current is applied, then, once the transient is over, $\bar{\theta}_i = \theta_i$. This, together with equation (11.2), allows to express θ_p and θ_n , and consequently (11.5), (11.6) and the OCP, in terms of SOC. Nevertheless, rather than using such an expression, we fit on it a polynomial function of the SOC

$$\begin{aligned} V_{oc}^E(t) = & 3.592 + 0.9082 z^E(t) - 0.57 z^E(t)^2 - 2.979 z^E(t)^3 + 6.56 z^E(t)^4 - 4.238 z^E(t)^5 \\ & + 0.8608 z^E(t)^6 - 1.676 \cdot 10^{-10} z^E(t)^7 + 1.143 \cdot 10^{-10} z^E(t)^8 - 2.982 \cdot 10^{-11} z^E(t)^9, \end{aligned} \quad (11.12)$$

which is simpler (and therefore may lead to less conservatism when evaluated through interval arithmetic), but still very accurate (see Figure 11.2).

The identification of the ECM parameters, besides the V_{oc} which is assumed known a priori, relies on the procedure described in Section 11.3.

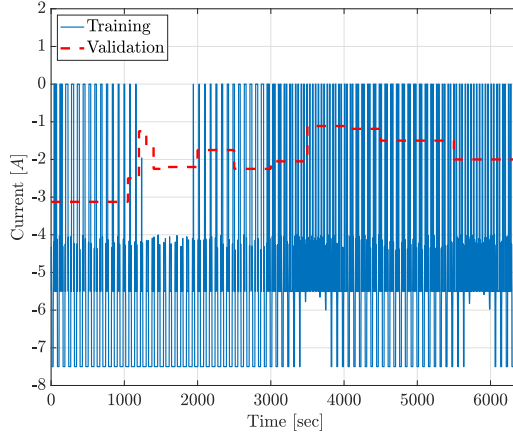


Figure 11.3: Training and validation current input profiles.

11.3 Identification of the ECM parameters

The ECM described in Section 11.2.2 is formally written as

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), \boldsymbol{\phi}), \\ \mathbf{y}(t) = \mathbf{g}(\mathbf{x}(t), \mathbf{u}(t), \boldsymbol{\phi}), \\ \mathbf{x}(t_0) = \mathbf{x}_0, \end{cases} \quad (11.13)$$

with

$$\boldsymbol{\phi} \triangleq [R_0 \ R_{1,0} \ R_{1,1} \ R_{1,2} \ C_{1,0} \ C_{1,1} \ C_{1,2} \ C_{\text{batt}}^E]. \quad (11.14)$$

In order to apply the interval state estimation method proposed in Section 11.4, the ECM is discretized according to the Euler's method, thus obtaining

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{x}_k + T_s \mathbf{f}(\mathbf{x}_k, u_k, \boldsymbol{\phi}), \\ \mathbf{y}_k = \mathbf{g}(\mathbf{x}_k, u_k, \boldsymbol{\phi}), \\ \mathbf{x}_{k_0} = \mathbf{x}_0, \end{cases} \quad (11.15)$$

where T_s is the sampling time, and $k \geq k_0$ is the time step.

According to [Perez et al. \(2017\)](#), R_1 and C_1 exhibit a different SOC relation for the charging and discharging phase. In this chapter, we consider the charging phase only. The parameter estimation is performed starting from data collected on the “real plant”, here assumed to be the SPMe. We consider as initial condition an almost completely discharged cell at rest, i.e. $\mathbf{x}_0 = (0.017, 0)$, and as input signal, defined over a time horizon \bar{k} , the one reported in Figure 11.3. Note that a sufficiently exciting input profile helps the identifiability and reduces parameters uncertainties.

Remark 11.2. Even though the ECM parameters still need to be estimated, \mathbf{x}_0 can be

easily obtained when a cell is at rest. In this case, the voltage across the parallel $R_1 - C_1$ is zero, V^E coincides with V_{oc}^E , and $z^E(0)$ can be computed from equation (11.12)⁴.

The system is affected by measurement noise that ranges within the interval $[-3mV, 3mV]$. In order to use the following identification procedure, in this section we approximate it as i.i.d. Gaussian noise $\mathbf{d}_k \sim \mathcal{N}(0, \sigma_y^2)$ with $\sigma_y = 1mV$. In particular, for the true value of the parameter vector ϕ^* , one has $\hat{\mathbf{y}}_k(\phi^*) = g(\mathbf{x}_k, \mathbf{u}_k, \phi^*) + \mathbf{d}_k$. Let $\hat{\mathbf{Y}}(\phi^*) \in \mathbb{R}^{\bar{k}+1}$ denote the vector of observed output data over the time horizon $k = k_0, \dots, k_0 + \bar{k}$. Then, it holds that $\hat{\mathbf{Y}}(\phi^*) \sim \mathcal{N}(\mathbf{Y}(\phi^*), \mathbf{C}_y)$, where $\mathbf{Y}(\phi^*)$ stands for the output vector in the absence of measurement noise and $\mathbf{C}_y \in \mathbb{R}^{(\bar{k}+1) \times (\bar{k}+1)}$ for the diagonal measurement covariance matrix given by $\mathbf{C}_y = \sigma_y^2 \mathbf{I}_{\bar{k}+1}$.

Once the training data is collected, the parameters $\hat{\phi}$ can be estimated by solving the following maximum likelihood optimization problem:

$$\min_{\phi} (\hat{\mathbf{Y}}(\phi^*) - \mathbf{Y}(\phi))^T \mathbf{C}_y^{-1} (\hat{\mathbf{Y}}(\phi^*) - \mathbf{Y}(\phi))$$

$$\text{s.t. model dynamics (11.15),}$$

$$\mathbf{h}(\mathbf{x}_k, \mathbf{u}_k, \phi) \geq \mathbf{0}, \quad (11.16a)$$

$$\phi^{\min} \leq \phi \leq \phi^{\max}, \quad (11.16b)$$

where $\mathbf{Y}(\phi)$ stands for the output vector obtained by solving, for a given ϕ , system (11.15) over the time horizon $k = k_0, \dots, k_0 + \bar{k}$. Equation (11.16b) allows to account for physical bounds on the parameters, e.g. $R_0 \geq 0$, $C_{\text{batt}}^E \geq 0$. Similarly, equation (11.16a) is required to bound SOC-dependent variables such as R_1 and C_1 . In practice, assuming functions (11.9) and (11.10) have the same concavity as in Perez et al. (2017), one can force the positivity of these quantities by setting constraints on their values for $z^E = 1$ and $z^E = 0$ (concave case), or for the z^E corresponding to the minimum value of the function (convex case). This latter can be found analytically. Further constraints can be added when prior knowledge on the parametric bounds is available. The value of the estimated parameters, obtained by solving problem (11.16), is reported in Table 11.2.

Table 11.2: Estimated ECM parameters.

$R_0(\Omega)$	$R_{1,0}(\Omega)$	$R_{1,1}(\Omega)$	$R_{1,2}(\Omega)$
0.093	0.0221	-0.07	0.0672
$C_{1,0}(\text{As})$	$C_{1,1}(\text{As})$	$C_{1,2}(\text{As})$	$C_{\text{batt}}^E(\text{As})$
235.52	$7.7613 \cdot 10^4$	$-7.0974 \cdot 10^4$	$2.6963 \cdot 10^4$

The ECM model with the estimated parameters has been tested in validation against the SPMe, using the validation profile in Figure 11.3 and the same initial condition of the training phase. As shown in Figure 11.4, the results are good but some discrepancy

⁴The OCP is assumed known a priori.

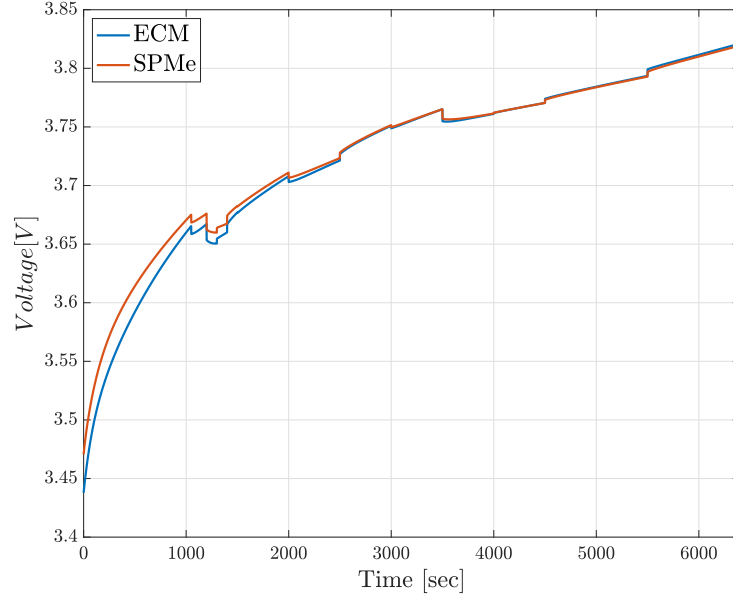


Figure 11.4: Validation profile of the terminal voltage.

is present. For state-estimation and control purposes, it is important to quantify how accurate the obtained parameter values are. Since $\hat{\mathbf{Y}}(\phi^*)$ is a random variable, one has that also the estimated $\hat{\phi}$ is a random variable with covariance matrix $\mathbf{C}_{\hat{\phi}} \in \mathbb{R}^{8 \times 8}$. Similarly to [Pozzi et al. \(2018\)](#), we rely on the Fisher Information Matrix (FIM) to obtain an estimate of $\mathbf{C}_{\hat{\phi}}$. According to the Cramer-Rao bound, the FIM provides a lower bound on the parameters covariance matrix as follows:

$$\mathbf{C}_{\hat{\phi}} \geq (\mathbf{F}(\hat{\phi}))^{-1} = ((\mathbf{S}(\hat{\phi}))^T \mathbf{C}_y^{-1} \mathbf{S}(\hat{\phi}))^{-1}, \quad (11.17)$$

where $\mathbf{F}(\hat{\phi}) \in \mathbb{R}^{8 \times 8}$ is the FIM, and $\mathbf{S}(\hat{\phi})$ is the sensitivity matrix computed as $\mathbf{S}(\hat{\phi}) \triangleq \nabla_{\hat{\phi}} \mathbf{Y}(\hat{\phi})$.

Remark 11.3. Despite the FIM inverse being commonly used in the literature to quantify the parametric uncertainty, this method can be inaccurate for several reasons: (i) it provides only a lower bound of $\mathbf{C}_{\hat{\phi}}$; (ii) it relies on local parameter sensitivity (which assumes a linear relationship between model parameter variations and simulation results); (iii) it is not able to capture the non-Gaussian case since it estimates only the covariance matrix; (iv) it assumes that the model structure used in the identification phase is correct. Future work will consider better estimates by using global sensitivity analysis and take into account the error committed when approximating a high-fidelity model with a reduced one (see e.g. [Weber et al. \(2019\)](#)).

The estimated correlation matrix $\mathbf{R}_{\hat{\phi}}$ ⁵, obtained using the FIM, is reported in Figure

⁵Given $\mathbf{D} \triangleq \sqrt{\text{diag}(\mathbf{C}_{\hat{\phi}})}$, the correlation matrix can be derived by $\mathbf{R}_{\hat{\phi}} = \mathbf{D}^{-1} \mathbf{C}_{\hat{\phi}} \mathbf{D}^{-1}$.

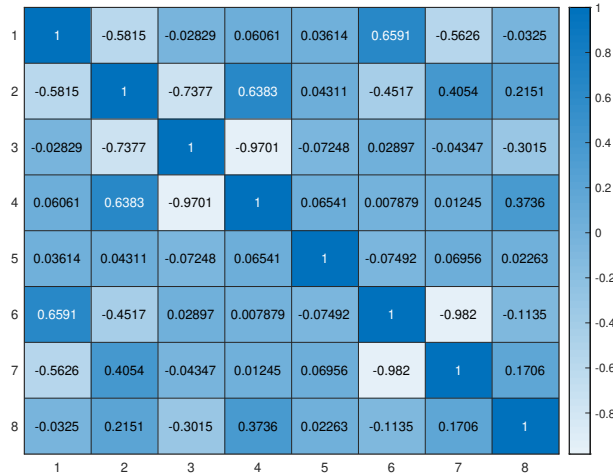


Figure 11.5: Parametric correlation matrix.

11.5. Non-zero off-diagonal values testify a correlation between the parameters in (11.14). In order to take into account this feature when using the interval state estimation of the next section, a linear change of variable is made $\hat{\psi} = \mathbf{T}\hat{\phi}$, where \mathbf{T} is chosen as the inverse of the eigenvector matrix of $\mathbf{C}_{\hat{\phi}}$ ⁶. By doing so, one obtains that the covariance matrix $\mathbf{C}_{\hat{\psi}}$ of the new parameter set $\hat{\psi}$ is diagonal.

The use of an unbounded distribution to describe the parametric uncertainty is overly conservative since physical electric parameters cannot be negative nor exceed meaningful values (it does not make any sense for the Kokam SLPB 75106100 cell we consider to have a resistance of $M\Omega$). For this reason, in the following section, we restrict each element of the parametric uncertainty $\hat{\psi}_i$ to the following domain $[\hat{\psi}_i - 3\sqrt{C_{\hat{\psi}}(i, i)}, \hat{\psi}_i + 3\sqrt{C_{\hat{\psi}}(i, i)}]$, thus truncating to ± 3 times the standard deviation of the parameter. These sets are further reduced by considering the physical limits of the original parameters $\hat{\phi}$ (e.g. positivity for each SOC value). Note that, considering a reference system with a diagonal covariance matrix is very important since intervals are not able to capture any coupling between variables.

The identification phase has been implemented using CasADi (Andersson et al., 2019). The optimization problem has been solved with IPOPT, a primal-dual interior point method interfaced with CasADi.

11.4 Set-based state estimation method

The set-based state estimation method proposed in this chapter relies on intervals. See Section 3.4 for basic operations and notations of interval analysis.

⁶Since $\mathbf{C}_{\hat{\phi}}$ is symmetric, then it has a complete set of orthogonal eigenvectors, and therefore, \mathbf{T} is invertible.

11.4.1 Problem formulation

Consider the discrete-time nonlinear model written below

$$\begin{cases} \mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}), \\ \mathbf{y}_k = \mathbf{g}(\mathbf{x}_k, \mathbf{u}_k, \mathbf{v}_k), \end{cases} \quad (11.18)$$

for $k \geq 1$, with $\mathbf{y}_0 = \mathbf{g}(\mathbf{x}_0, \mathbf{u}_0, \mathbf{v}_0)$, $\mathbf{f} : \mathbb{R}^n \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_w} \rightarrow \mathbb{R}^n$, $\mathbf{g} : \mathbb{R}^n \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_v} \rightarrow \mathbb{R}^{n_y}$, where $\mathbf{w} \in W$ and $\mathbf{v} \in V$ are the unknown-but-bounded process and measurement disturbances, respectively, with W and V being intervals.

The objective is to obtain accurate interval enclosures \hat{X}_k of the state variables \mathbf{x}_k which are consistent with the nonlinear system (11.18) and the measurement \mathbf{y}_k . Given an initial set \bar{X}_0 such that $\mathbf{x}_0 \in \bar{X}_0$, in this chapter we proceed through the well-known prediction-update algorithm, which is based on computing intervals \bar{X}_k and \hat{X}_k such that

$$\bar{X}_k \supseteq \{\mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}) : \mathbf{x}_{k-1} \in \bar{X}_{k-1}, \mathbf{w}_{k-1} \in W\}, \quad (11.19)$$

$$\hat{X}_k \supseteq \{\mathbf{x}_k \in \bar{X}_k : \mathbf{g}(\mathbf{x}_k, \mathbf{u}_k, \mathbf{v}_k) = \mathbf{y}_k, \mathbf{v}_k \in V\}, \quad (11.20)$$

where (11.19) is referred to as the *prediction step*, and (11.20) as the *update step*. We assume that an interval enclosure \bar{X}_0 of the initial state \mathbf{x}_0 and the current measurement \mathbf{y}_k for $k \geq 0$ are known.

In this chapter, the guaranteed state estimation is performed in an efficient way by combining inclusion functions with *forward-backward constraint propagation* (FBCP) (Jaulin et al., 2001a). For each time k , given the previous state set \hat{X}_{k-1} , the prediction step (11.19) is performed using inclusion functions (Moore et al., 2009), resulting in the predicted interval \bar{X}_k such that $\mathbf{x}_k \in \bar{X}_k$. On the other hand, the update step (11.20) is computed by solving a Constraint Satisfaction Problem (CSP).

To obtain $\hat{X}(k)$, the general problem is to refine \bar{X}_k and \bar{Y}_k by removing values in the respective domains that are not consistent with each other. This corresponds to a CSP \mathcal{H} , which is formulated as

$$\mathcal{H} : \mathbf{y}_k = \mathbf{g}(\mathbf{x}_k, \mathbf{u}_k, \mathbf{v}_k), (\mathbf{x}_k, \mathbf{y}_k, \mathbf{v}_k) \in \bar{X}_k \times \bar{Y}_k \times V,$$

whose solution set is defined as $\mathcal{S} \triangleq \{(\mathbf{y}_k, \mathbf{x}_k) \in \bar{Y}_k \times \bar{X}_k : \mathbf{y}_k = \mathbf{g}(\mathbf{x}_k, \mathbf{u}_k, \mathbf{v}_k), \mathbf{v}_k \in V\}$. To solve \mathcal{H} , during the forward constraint propagation phase, the state set \bar{X}_k is first propagated through \mathbf{g} using inclusion functions, considering intermediate variables, yielding an output interval \bar{Y}_k which is refined by intersecting it with the measurement \mathbf{y}_k . During the backward propagation, the nonlinear mapping \mathbf{g} is swept backwards and the interval \bar{Y}_k obtained in forward propagation phase is used to refine \bar{X}_k . The proposed FBCP algorithm for the ECM is detailed in the next section.

11.4.2 State estimation of the Lithium-ion cell using FBCP

The FBCP algorithm proposed in [Jaulin et al. \(2001a\)](#) is composed of three intermediate steps (*contractor decomposition*, *forward update*, and *backward update*), here applied to the ECM example.

We consider (11.15) describing the ECM with output function \mathbf{g} given by (11.11). The process disturbance $\mathbf{w}_k = \mathbf{T}\hat{\phi}$ is determined by the parameters uncertainties in the ECM by means of the transformation matrix in Section 11.3, bounded by $\mathbf{w}_k \in W \triangleq \mathbf{T}[\hat{\phi}]$. On the other hand, the output disturbance \mathbf{v}_k comprises both the parameters uncertainties and output additive measurement noise $\mathbf{d}_k \in D \subset \mathbb{R}$, with \mathbf{d} defined in Section 11.3, as $\mathbf{v}(k) \triangleq [\mathbf{w}(k)^T \ \mathbf{d}(k)]^T$. Note that, since R_0 is correlated to the other parameters appearing in (11.14) (see Figure 11.5), \mathbf{v}_k depends also on \mathbf{w}_k .

In order to mitigate the dependency effect that arises when considering the polynomial function (11.12), $V_{\text{oc}}^{\text{E}}(k)$ is rewritten using the centered form suggested in [Hansen & Walster \(2003\)](#), as

$$V_{\text{oc}}^{\text{E}}(x_{1,k}) = p_{0,k} + \sum_{i=1}^9 p_{i,k} (x_{1,k} - c)^i, \quad (11.21)$$

where $c \triangleq \text{mid}(\bar{X}_{1,k})$, with $x_{1,k} \triangleq z^{\text{E}}(k) \in \bar{X}_{1,k}$ being the first component of \bar{X}_k , p_i being auxiliary variables given by $p_{0,k} \triangleq V_{\text{oc}}^{\text{E}}(c)$, $p_{i,k} \triangleq (1/i!)V_{\text{oc}}^{\text{E},[i]}(c)$, $i \in \{1, \dots, 9\}$, and $V_{\text{oc}}^{\text{E},[i]}(c)$ denoting the i -th derivative of (11.12) with respect to $z^{\text{E}}(k)$, evaluated at c .

1) *Contractor decomposition.* The function \mathbf{g} derived from (11.11) is first decomposed into a ‘‘primitive’’ form comprised of simplified expressions in which only one function or one elementary operation is present. This is achieved by introducing intermediate variables $h_{i,k} \in H_{i,k}$ given by

$$h_{i,k} \triangleq (x_{1,k} - c)^i, \quad i \in \{1, \dots, 9\}, \quad (11.22)$$

which allow to rewrite the polynomial function (11.12) as

$$V_{\text{oc}}^{\text{E}}(k) = p_{0,k} + \sum_{i=1}^9 p_{i,k} h_{i,k}. \quad (11.23)$$

Therefore, intervals $H_{i,k}$ and $\mathcal{V}_k^{\text{E}} \triangleq \{V_{\text{oc}}^{\text{E}}(k) \text{ given by (11.23) : } h_{i,k} \in H_{i,k}\}$ are computed by evaluating (11.22) for all $x_{1,k} \in \bar{X}_{1,k}$, and (11.23) for all $h_{i,k} \in H_{i,k}$, respectively, using interval arithmetic.

2) *Forward update.* Given the measurement $\mathbf{y}(k)$, we first subtract the noise interval $[\mathbf{d}]$, obtaining $\bar{Y} \triangleq \mathbf{y}(k) - \mathbf{d}$, for all $\mathbf{d} \in D$. Then, we compute the intersection

$$\bar{Y}_k \leftarrow \bar{Y}_k \cap (\mathcal{V}_k^{\text{E}} \oplus \bar{X}_{2,k} \oplus R_0 u_k), \quad (11.24)$$

in order to refine \bar{Y}_k (Minkowski sums are obtained using interval addition). Note that if no measurement is available at time k , this set is initialized as $\bar{Y}_k \leftarrow (-\infty, +\infty)$.

3) *Backward update.* The decomposed function \mathbf{g} is swept backwards. Every variable appearing on the right-hand-side of \mathbf{g} is made an explicit function of the other variables appearing in the forward update, as

$$\mathcal{V}_k^E \leftarrow \mathcal{V}_k^E \cap (\bar{Y}_k \oplus (-\bar{X}_{2,k}) \oplus (-R_0 u_k)), \quad (11.25a)$$

$$\hat{X}_{2,k} \leftarrow \bar{X}_{2,k} \cap (\bar{Y}_k \oplus (-\mathcal{V}_k^E) \oplus (-R_0 u_k)). \quad (11.25b)$$

For assessing the state of charge $\hat{X}_{1,k}$, equation (11.23) is swept backwards and decomposed into multiple expressions, passing first through the intermediate variables $H_{i,k}$, for $i = 1, \dots, 9$, as

$$H_{i,k} \leftarrow H_{i,k} \cap \left[\frac{1}{p_{i,k}} \left(\mathcal{V}_k^E \oplus (-p_{0,k}) \bigoplus_{j \neq i} (-p_{j,k} H_{j,k}) \right) \right]. \quad (11.26)$$

The interval $H_{1,k}$ is further refined as follows. Since $H_{1,k}$ is zero centered by definition (see (11.22)), the following logic is necessary for retrieving both positive and negative even roots to $H_{1,k}$:

$$H_{1,k} \leftarrow H_{1,k} \cap_{i \in \{2,4,6,8\}} H_{i,k}^{\frac{1}{i}} = H_{1,k} \cap_{i \in \{2,4,6,8\}} [-|\bar{h}_{i,k}|^{\frac{1}{i}}, |\bar{h}_{i,k}|^{\frac{1}{i}}].$$

For odd roots,

$$H_{1,k} \leftarrow H_{1,k} \cap_{i \in \{3,5,7,9\}} ([\underline{\alpha}_{i,k}, \bar{\alpha}_{i,k}]),$$

where (i) $\underline{\alpha}_{i,k} \triangleq \underline{h}_{i,k}^{\frac{1}{i}}$ if $h_{i,k} > 0$ for all $h_{i,k} \in H_{i,k}$, $\underline{\alpha}_{i,k} \triangleq -(|\underline{h}_i|^{\frac{1}{i}})$ otherwise, and (ii) $\bar{\alpha}_{i,k} \triangleq -(|\bar{h}_i|^{\frac{1}{i}})$ if $h_{i,k} < 0$ for all $h_{i,k} \in H_{i,k}$, $\bar{\alpha}_{i,k} \triangleq \bar{h}_i^{\frac{1}{i}}$ otherwise, with $\underline{h}_{i,k}$ and $\bar{h}_{i,k}$ being the lower and upper bounds of $H_{i,k}$, respectively. Finally, the refinement of the SOC interval is done through the intersection $\hat{X}_{1,k} \leftarrow \bar{X}_{1,k} \cap (H_{1,k} \oplus c)$. The interval enclosure \hat{X}_k for the update step (11.20) is then given by $\hat{X}_k \leftarrow \hat{X}_{1,k} \times \hat{X}_{2,k}$.

Remark 11.4. Steps 2 and 3 can be repeated iteratively to obtain a more refined enclosure \hat{X}_k (Jaulin et al., 2001a). Nevertheless, for the ECM, one iteration demonstrated to provide sufficiently accurate enclosures, as illustrated in Section 11.5.

Remark 11.5. Since intervals cannot capture the fact that \mathbf{w}_k is constant, as well as the dependencies between \mathbf{v}_k and \mathbf{w}_k , the application of the FBCP algorithm to the ECM is then conservative. Future work will explore other set representations such as zonotopes, which allow to effectively manage this dependency and to reduce conservatism.

11.5 Numerical results

This section presents the results obtained by applying the interval state estimation method to the considered Li-ion cell during charging phase with input profile in Figure 11.6.

The initial state set \bar{X}_0 is centered in the initial state condition $(z^E(0), V_{c1}(0)) = (0.3, 0)$.

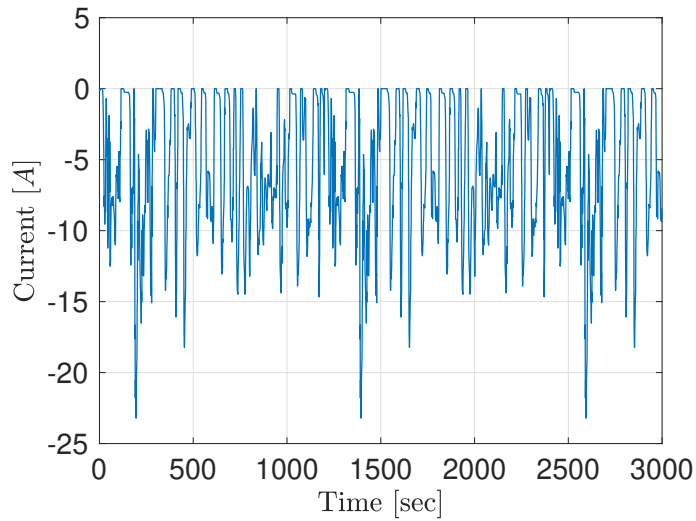


Figure 11.6: Current input profile $I(t)$ applied to the ECM.

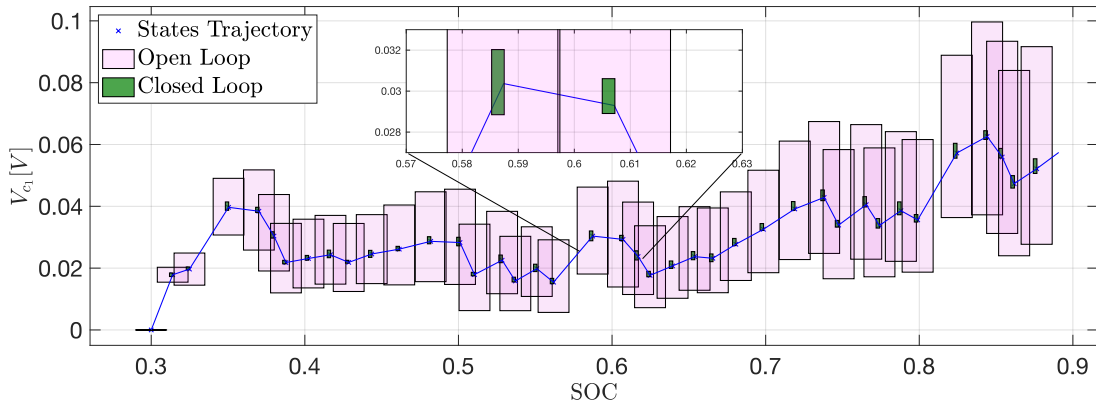


Figure 11.7: Enclosures obtained using interval state estimation.

We consider an initial state uncertainty of 1% for the SOC and a much lower uncertainty on the voltage V_{c1} of 0.0001% V . This is justified by assuming that the battery is in stationary condition at $t = 0$ after a rest period (see Remark 11.2). The open loop state estimation is computed using only the prediction step (11.19) with natural inclusion function (Moore et al., 2009), at each time step k . Figure 11.7 depicts a comparison between states set computed in open loop (pink) and closed loop (green) respectively, highlighting the result obtained after the refinement procedure related to FBCP.

A further comparison between open and closed loop estimation is also shown in Figure 11.8, in which the areas of the enclosures are calculated at each time k . Note that, despite the improved accuracy, the sets mildly increase over time. This is due to the increase of the SOC with time due to charging, which results in more conservative interval enclosure of the polynomial (11.23). As it can be noticed, the use of the FBCP approach (closed-loop) significantly reduces the size of the obtained intervals and still guarantees that the real state trajectory \mathbf{x}_k (depicted in blue in Figure 11.7) belongs to \hat{X}_k . The simulation has been

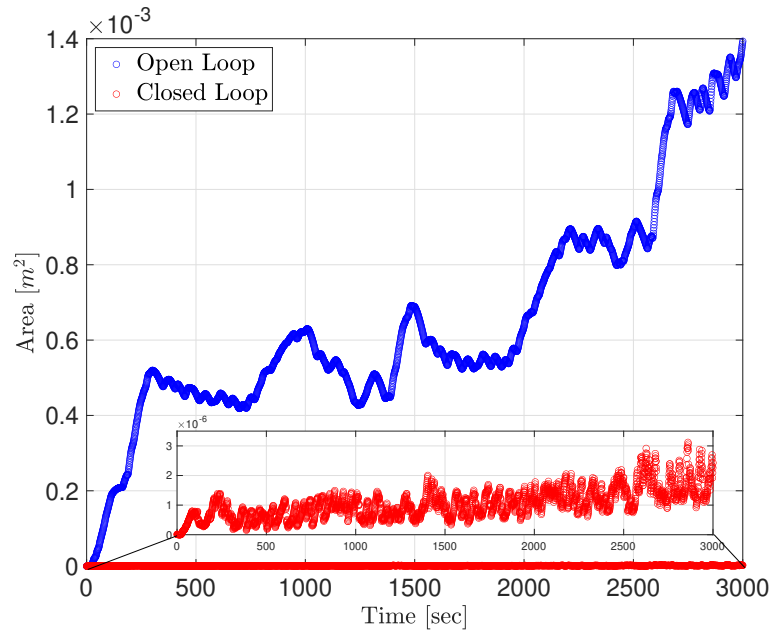


Figure 11.8: Area of the estimated enclosures over time.

done with MATLAB 2020a, while all interval operations were performed using INTLAB 9 (Rump, 1999). The plots in Figure 11.7 has been done using MPT (Kvasnica et al., 2004).

11.6 Final remarks

This chapter developed a discrete-time interval observer for a single cell Li-ion battery based on Jaulin et al. (2001b). Parameters have been identified by mean of least squares method using voltage data collected on an SPMe. Parametric uncertainties have been derived exploiting FIM, whose inverse approximates the covariance of the parameters. A variable transformation of the parameters have been performed to take their correlation into account. Simulation results demonstrated tight enclosures on the states over time, which is corroborated by comparing open-loop and closed-loop state estimation.

12

Conclusions

This chapter summarizes the main contributions obtained in this doctoral thesis, and concludes the text. Future work proposals are also presented and detailed at the end of this chapter.

12.1 Overview and contributions

This doctoral thesis developed new methods for set-based state estimation and active fault diagnosis of nonlinear discrete-time systems with unknown-but-bounded uncertainties, discrete-time nonlinear systems whose trajectories are known to satisfy nonlinear equality constraints (called invariants), set-based state estimation and active fault diagnosis of linear descriptor systems, and joint state and parameter estimation of nonlinear descriptor systems. Set-based estimation aims to compute tight enclosures of the possible system states in each time step subject to unknown-but-bounded uncertainties. Most existing methods employ a standard prediction-update framework with set-based prediction and update steps based on various set representations and techniques. However, achieving accurate enclosures for nonlinear systems remains a significant challenge. When these enclosures are represented by simple sets such as intervals, ellipsoids, parallelotopes, and zonotopes, certain set operations can be very conservative. Yet, using general convex polytopes is much more computationally demanding.

To address this challenge, this thesis presented new methods for efficiently propagating constrained zonotopes through nonlinear mappings:

- Chapter 4 developed three new different approaches for computing the prediction step (4.2) based on constrained zonotopes. The first method (Section 4.2.1) combined properties of interval arithmetic and constrained zonotopes, leading to a highly tunable and accurate state estimation algorithm. The other methods (Sections 4.2.2 and 4.2.3) expanded the tools proposed in Scott et al. (2016) to the class of nonlinear discrete-time systems (4.1).
- These methods extended, in a consistent way, two existing approaches for propagating zonotopes through nonlinear mappings. The key advantage of these new extensions is that they allowed the entire state estimation procedure to be done using constrained zonotopes in CG-rep. Therefore, the update step could be done by exact intersection (with linear measurements), which is known to generate highly asymmetrical sets that cannot be accurately enclosed by ellipsoids, intervals, parallelotopes, and zonotopes.
- Using the methods proposed in Chapter 4, such sets could be directly propagated to the next time step without prior simplification to a symmetric set. This overcomes a major source of conservatism in existing methods based on the aforementioned enclosures, while largely retaining the efficiency of computations with zonotopes.

In addition, this thesis improved the standard prediction-update framework for systems with invariants by adding a consistency step. This new step used invariants to reduce conservatism and was enabled by new algorithms for refining constrained zonotopes based on nonlinear constraints. Also, new update algorithms were developed allowing nonlinear measurement equations in CZ-based state estimation for the first time:

- Chapter 5 presented new methods for set-valued state estimation of discrete-time nonlinear systems whose trajectories are known to satisfy nonlinear equality constraints, called invariants.
- The new consistency step used invariants to reduce conservatism and is enabled by new algorithms for refining constrained zonotopes based on nonlinear constraints.
- Chapter 5 also presented significant improvements to prediction and update steps proposed in Chapter 4. Specifically, new update algorithms were developed that allow nonlinear measurement equations, and existing prediction methods based on conservative approximation techniques were modified to allow a more flexible choice of the approximation point, which led to tighter enclosures.
- Numerical results demonstrated that the proposed methods can provide significantly tighter enclosures than existing zonotope-based methods while maintaining comparable efficiency.

This thesis also introduced a new approach for set-based active fault diagnosis of a class of nonlinear discrete-time systems. The new enclosures based on constrained zonotopes are employed for passive fault detection of a class of nonlinear discrete-time systems:

- Chapter 6 proposed an adaptation of the first-order Taylor extension (Section 4.2.3) to obtain an affine parametrization of the reachable sets of systems with nonlinear dynamics. Based on this affine parametrization of reachable sets, a new method for input design aiming active fault diagnosis of nonlinear discrete-time systems has been proposed. The method was based on the open-loop fault diagnosis approach proposed in Scott et al. (2014) for linear systems using zonotopes.
- Chapter 6 also extended the active fault diagnosis method to constrained zonotopes, based on the approach proposed in Raimondo et al. (2016).
- The resulting problem consisted in solving an MIQP to find the optimal input sequence that allows guaranteed fault diagnosis in a given time horizon. The performance of the proposed active fault diagnosis method was demonstrated in a numerical experiment, which consisted in fault diagnosis of a robotic manipulator subject to actuator faults. The diagnosis of the current possible faults was guaranteed in the experiment by injecting the designed optimal input sequence into the nonlinear system.

In addition, this thesis presented new methods based on constrained zonotopes for set-valued state estimation and active fault diagnosis of linear descriptor systems, besides a new set representation, called mixed mixed zonotope:

- Chapter 7 proposed novel algorithms for set-based state estimation and AFD of linear descriptor systems with unknown-but-bounded uncertainties. The first methods used constrained zonotopes. In contrast to other sets representations, linear static constraints on the state variables, typical of descriptor systems, can be directly incorporated in the mathematical description of constrained zonotopes. Thanks to this feature, set-based methods based on CZs could provide less conservative enclosures for the trajectories of linear descriptor systems.
- Chapter 7 proposed also a new representation for unbounded sets based on zonotopes, called mixed zonotopes. These are a generalization of zonotopes capable of describing unbounded sets, which allowed to develop methods for state estimation and active fault diagnosis of linear descriptor systems without assuming the knowledge of an admissible set that encloses all the possible trajectories of the system.
- The new methods based on constrained zonotopes and mixed zonotopes led to significantly tighter results than zonotope approaches. In addition, active fault

diagnosis was enabled without assuming rank properties on the structure of the system. The effectiveness of the new methods was corroborated by numerical examples.

This thesis also presented a new method for set-based joint state and parameter estimation of nonlinear discrete-time systems:

- Chapter 8 developed a new method for set-based joint state and parameter estimation of nonlinear discrete-time systems with algebraic variables and unknown model parameters. This method extended the nonlinear state estimation methods using constrained zonotopes proposed in Chapter 5 to include the estimation of both algebraic variables and unknown parameters in a unified framework.
- Unlike other methods in the literature, the proposed method maintained existing dependencies between states, algebraic variables, and unknown parameters, which resulted in the accuracy of both state and parameter estimation being significantly improved. The effectiveness of the new methodology was highlighted in two numerical examples, with the second example being a nonlinear descriptor system.

Lastly, this thesis presented the application of the proposed set-based state estimation and fault diagnosis methods using constrained zonotopes to unmanned aerial vehicles, water distribution networks, and a lithium-ion cell:

- Chapter 9, initially, dealt with the problem of path tracking of a suspended load using a tilt-rotor UAV. Such task requires the knowledge of the load position, which may not be provided by available sensors. Therefore, to accomplish the path tracking, a linear set-based state estimator based on constrained zonotopes was proposed to solve the problem of estimating the load position and orientation, considering sensors with different sampling times, and unknown-but-bounded disturbances. In order to provide feedback to the controller using the estimated state set, an optimal choice of the estimated point was given according to a proposed constrained minimum-variance criterion. The path tracking was then solved by a discrete-time mixed $\mathcal{H}_2/\mathcal{H}_\infty$ controller with pole-placement constraints. Results from numerical experiments, performed in a platform based on the Gazebo simulator with a CAD model of the system were presented to corroborate the performance of the linear set-based state estimator along with the designed control strategy. The second application was a numerical experiment considering nonlinear state estimation of a quadrotor UAV, taking into account a realistic scenario in which measurements are provided by sensors located at the UAV. The third application considered a numerical experiment involving the optimal input design for nonlinear active fault diagnosis of a quadrotor UAV subject to actuator and sensor faults, which addressed scenarios of complete actuator failure and the existence of sensor bias.

- Chapter 10 proposes two new methods for set-based state estimation in WDNs, considering unknown-but-bounded measurement and model uncertainties to calculate bounds on the system states. A new interval method was proposed based on iterative computation of tight enclosures of the nonlinear head-loss functions, and bounding the solution of the algebraic equations using rescaling. This method, denoted by ISAR, was mildly more conservative than the IHISE method previously proposed in Vrachimis et al. (2019), but significantly more computationally efficient. However, since intervals are not capable of capturing the dependencies between state variables, Chapter 10 also proposed the use of constrained zonotopes as an additional step to both IHISE and ISAR. This led to two new algorithms for set-based state estimation of WDNs, capable of capturing the dependencies between hydraulic states, which result in sets with significantly smaller volumes than intervals. The benefits of constrained zonotopes were also highlighted when the new enclosures were used for leakage detection. These provided higher leakage detection rates compared to intervals, as demonstrated in two case studies using benchmark WDNs.
- Chapter 11 developed a discrete-time interval state estimator for a single Li-ion cell, based on the forward-backward method described in Jaulin et al. (2001a). In particular, parametric uncertainties were considered and obtained by performing an identification procedure on data collected on a well known EM, i.e. the SPMe. Among the contributions of the chapter, were: (i) the identification of ECM parameter bounds based on the Fisher Information Matrix, which was performed by the PhD candidate Diego Locatelli, being supervised by Prof. Davide M. Raimondo; and (ii) a discrete-time interval state estimation method based on inclusion functions and constraint propagation, which handles discrete-time measurement, and is the contribution concerning this doctoral thesis. Numerical experiments showed that the proposed methodology is efficient and provided accurate enclosures for both the SOC and the electric state variables of the ECM.

12.2 Future work

This section describes some possible directions for future work derived from the contributions presented in this doctoral thesis.

- *Efficient implementation of the nonlinear state estimation methods via parallelization.* The methods developed in Chapters 4 and 5 were able to provide more accurate enclosures than existing zonotopic methods in the literature. However, the proposed methods have an increased computational burden, which may prevent their application in some practical systems. To circumvent this issue, an immediate future work is the implementation of the proposed algorithms using parallelization procedures

in Graphics Processing Units (GPUs). This would allow for reduced computation times, and therefore the application of the proposed methods in an experimental setup for unmanned aerial vehicles, for instance.

- *Active fault diagnosis of discrete-time systems with nonlinear dynamics with reduced conservatism.* Chapter 6 developed an active fault diagnosis method for discrete-time systems with nonlinear dynamics based on the affine parameterization of reachable sets in the inputs, using the first-order Taylor extension proposed in Chapter 4. However, the affine parameterization obtained requires the evaluation of the reachable sets over the entire admissible input set, which may result in severe conservatism. Future work would seek different parameterizations of the reachable sets such that this conservatism is reduced.
- *Active fault diagnosis of discrete-time systems with nonlinear dynamics, nonlinear measurement and nonlinear invariants.* The active fault diagnosis method proposed in Chapter 6 was developed for discrete-time systems assuming nonlinear dynamics and linear measurement. In this way, the reachable sets could be approximated using the first-order Taylor extension proposed in Chapter 4. A direct continuation of this research is to rather consider the first-order Taylor extension developed in Chapter 5, which allows also for nonlinear measurement and takes into account the existence of nonlinear invariants. In addition, the choice of approximation point would be improved with respect to the current active fault diagnosis method, since it would be allowed to belong to the interval hull of the previous enclosure, as proposed in Chapter 5.
- *Closed-loop active fault diagnosis of nonlinear discrete-time systems.* The active fault diagnosis method proposed in Chapter 6 is an open-loop structure, i.e., it does not make use of measurements obtained by injecting the optimal input into the system to improve the diagnosis procedure. After achieving the results suggested in the two future work proposals above, a continuation of this research is to extend the developed fault diagnosis method to a closed-loop structure as proposed in [Raimondo et al. \(2016\)](#), using a moving horizon scheme and nonlinear observers to shorten the input sequence required to perform the fault diagnosis.
- *Closed-loop fault diagnosis of linear descriptor systems.* Similarly to the proposal above, the active fault diagnosis methods developed in Chapter 7 using constrained zonotopes and mixed zonotopes are open-loop structures. A direct continuation of this research is to develop closed-loop methods for linear descriptor systems by using a moving-horizon scheme and set-based observers to take into account the measurements obtained during the fault diagnosis procedure.

- *State estimation and fault diagnosis of nonlinear systems using mixed zonotopes.* A new representation for unbounded sets has been proposed in Chapter 7, referred to as mixed zonotopes. This new set representation allowed for state estimation and fault diagnosis of linear descriptor systems without assuming the knowledge of an admissible state set. Moreover, in the state estimation case, it allowed also for obtaining accurate enclosures of the system states without assuming the knowledge of a bounded initial set. A future work proposal is to investigate the development of state estimation and fault diagnosis methods for nonlinear systems using mixed zonotopes. The main challenge in the development of such methods is the evaluation of the remainder set in the enclosure approximations, since the previous set may be unbounded.
- *Investigate the conditions in which mixed zonotopes are able to provide bounded enclosures for linear descriptor systems.* The results obtained in Chapter 7 showed that it was possible to obtain bounded enclosures for the trajectories of linear descriptor systems using mixed zonotopes. However, this boundedness property is not guaranteed by the proposed methods. A direct continuation of this research is to investigate which are the necessary conditions and properties of the linear descriptor system which result in bounded enclosures for the trajectories of the states.

Bibliography

- Alamo, T., Bravo, J., & Camacho, E. (2005). Guaranteed state estimation by zonotopes. *Automatica*, 41(6), 1035–1043.
- Alamo, T., Bravo, J. M., Redondo, M. J., & Camacho, E. F. (2008). A set-membership state estimation algorithm based on DC programming. *Automatica*, 44(1), 216–224.
- Almeida, M. M. & Raffo, G. V. (2015). Nonlinear control of a tiltrotor UAV for load transportation. In *11th IFAC Symposium on Robot Control* (pp. 234–239).
- Althoff, M. (2013). Reachability analysis of nonlinear systems using conservative polynomialization and non-convex sets. *Hybrid Systems: Computation and Control*, (pp. 173–182).
- Althoff, M. & Krogh, B. H. (2011). Zonotope bundles for the efficient computation of reachable sets. In *Proc. of the 50th IEEE Conference on Decision and Control* (pp. 6814–6821).
- Althoff, M., Stursberg, O., & Buss, M. (2008). Reachability analysis of nonlinear systems using conservative linearization. In *Proc. of the 47th IEEE Conference on Decision and Control* (pp. 4042–4048).
- Althoff, M., Stursberg, O., & Buss, M. (2010). Computing reachable sets of hybrid systems using a combination of zonotopes and polytopes. *Nonlinear Analysis: Hybrid Systems*, 4(2), 233–249.
- Alwi, H., Edwards, C., Stroosma, O., & Mulder, J. A. (2008). Fault tolerant sliding mode control design with piloted simulator evaluation. *Journal of Guidance, Control and Dynamics*, 31(5), 1186–1201.
- Amiri, N., Ramirez-Serrano, A., & Davies, R. (2011). Modelling of opposed lateral and longitudinal tilting dual-fan unmanned aerial vehicle. In *Proc. of the 18th IFAC World Congress* (pp. 2054–2059).

- Andersson, J. A. E., Gillis, J., Horn, G., Rawlings, J. B., & Diehl, M. (2019). CasADi – A software framework for nonlinear optimization and optimal control. *Mathematical Programming Computation*, 11(1), 1–36.
- Arsene, C. T., Al-Dabass, D., & Hartley, J. (2011). Confidence limit analysis of water distribution systems based on a least squares loop flows state estimation technique. In *2011 UKSim 5th European Symposium on Computer Modeling and Simulation* (pp. 94–101).: IEEE.
- Avni, N., Fishbain, B., & Shamir, U. (2015). Water consumption patterns as a basis for water demand modeling. *Water Resources Research*, 51(10), 8165–8181.
- Bargiela, A. & Hainsworth, G. D. (1989). Pressure and flow uncertainty in water systems. *Journal of Water Resources Planning and Management*, 115(2), 212–229.
- Bargiela, A., Pedrycz, W., & Tanaka, M. (2003). A study of uncertain state estimation. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 33(3), 288–301.
- Bars, F. L., Sliwka, J., Jaulin, L., & Reynet, O. (2012). Set-membership state estimation with fleeting data. *Automatica*, 48(2), 381–387.
- Bender, D. J. & Laub, A. J. (1987). The linear-quadratic optimal regulator for descriptor systems: Discrete-time case. *Automatica*, 23, 71–85.
- Bernard, M., Kondak, K., Maza, I., & Ollero, A. (2011). Autonomous transportation and deployment with aerial robots for search and rescue missions. *Journal of Field Robotics*, 28(6), 914–931.
- Bertsekas, D. P. (1999). *Nonlinear Programming*. Athena Scientific, 2nd edition.
- Bisgaard, M. (2008). *Modeling, Estimation and Control of Helicopter Slung Load System*. PhD thesis, Aalborg University.
- Bisgaard, M., Cour-Harbo, A. I., & Bendtsen, J. D. (2010). Adaptive control system for autonomous helicopter slung load operations. *Control Engineering Practice*, 18(7), 800–811.
- Bisgaard, M., la Cour-Harbo, A., & Bendtsen, J. D. (2009). Swing damping for helicopter slung load systems using delayed feedback. In *AIAA Guidance, Navigation, and Control Conference*.
- Bisgaard, M., la Cour-Harbo, A., Johnson, E. N., & Bendtsen, J. D. (2007a). Full state estimation for helicopter slung load system. In *AIAA Guidance, Navigation and Control Conference and Exhibit*.

- Bisgaard, M., la Cour-Harbo, A., Johnson, E. N., & Bendtsen, J. D. (2007b). Vision aided state estimator for helicopter slung load system. In *17th IFAC Symposium on Automatic Control in Aerospace* (pp. 425–430).
- Bizeray, A. M., Zhao, S., Duncan, S. R., & Howey, D. A. (2015). Lithium-ion battery thermal-electrochemical model-based state estimation using orthogonal collocation and a modified extended Kalman filter. *Journal of Power Sources*, 296, 400–412.
- Blanchini, F., Casagrande, D., Giordano, G., Miani, S., Oлару, S., & Reppa, V. (2018). Fault isolation for large scale discrete-time systems based on implicit set representation. In *Proc. of 2018 European Control Conference* (pp. 685–690).
- Blanchini, F. & Miani, S. (2015). *Set-theoretic methods in control*. Springer, 2nd edition.
- Blesa, J., Puig, V., & Saludes, J. (2012). Robust fault detection using polytope-based set-membership consistency test. *IET Control Theory and Applications*, 6(12), 1767–1777.
- Blesa, J., Puig, V., Saludes, J., & Vento, J. (2010). Leak detection, isolation and estimation of pressurized water pipe networks using lpv models and zonotopes. In *Proc. of the 8th IFAC Symposium on Nonlinear Control Systems* (pp. 36–41).
- Bocquet, M. & Sakov, P. (2013). Joint state and parameter with an iterative ensemble kalman smoother. *Nonlinear Processes in Geophysics*, 20, 803–818.
- Boulos, P. F., Lansey, K. E., & Karney, B. W. (2006). *Comprehensive Water Distribution Systems Analysis Handbook for Engineers and Planners*. MWH Soft, Incorporated.
- Boyd, S. & Vandenberghe, L. (2004). *Convex Optimization*. Cambridge.
- Bravo, J. M., Alamo, T., & Camacho, E. F. (2006). Bounded error identification of systems with time-varying parameters. *IEEE Transactions on Automatic Control*, 51(7), 1144–1150.
- Cardoso, D. N., Esteban, S., & Raffo, G. V. (2021). A new robust adaptive mixing control for trajectory tracking with improved forward flight of a tilt-rotor UAV. *ISA Transactions*, 110, 86–104.
- Cardoso, D. N., Raffo, G. V., & Esteban, S. (2016). A robust adaptive mixing control for improved forward flight of a tilt-rotor UAV. In *Proc. of 19th International Conference on Intelligent Transportation Systems* (pp. 1432–1437).
- Casillas, M., Puig, V., Garza-Castañón, L., & Rosich, A. (2013). Optimal Sensor Placement for Leak Location in Water Distribution Networks Using Genetic Algorithms. *Sensors*, 13(11), 14984–15005.

- Castaldi, P., Mimmo, N., & Simani, S. (2014). Differential geometry based active fault tolerant control for aircraft. *Control Engineering Practice*, 32, 227–235.
- Chabane, S. B., Maniu, C. S., Alamo, T., Camacho, E., & Dumur, D. (2014a). Improved set-membership estimation approach based on zonotopes and ellipsoids. In *Proc. of the 2014 European Control Conference* (pp. 993–998).
- Chabane, S. B., Maniu, C. S., Alamo, T., Camacho, E. F., & Dumur, D. (2014b). Ellipsoidal state estimation for systems with interval uncertainties. In *Proc. of the 53rd IEEE Conference on Decision and Control* (pp. 2603–2608).
- Chabane, S. B., Maniu, C. S., Alamo, T., Camacho, E. F., & Dumur, D. (2014c). Improved set-membership estimation approach based on zonotopes and ellipsoids. In *Proc. of the 2014 European Control Conference* (pp. 993–998).
- Chabane, S. B., Maniu, C. S., Alamo, T., Camacho, E. F., & Dumur, D. (2014d). A new approach for guaranteed ellipsoidal state estimation. In *Proc. of the 19th IFAC World Congress* (pp. 6533–6538).
- Chabert, G. & Jaulin, L. (2009). Contractor programming. *Artificial Intelligence*, 173, 1079–1100.
- Chen, K. & Brdys, M. (1995). Set membership estimation of state and parameters of large water supply and distribution systems?integrated quantity and quality models. *IFAC Proceedings Volumes*, 28(10), 543–548.
- Chen, S. J., Hein, B., & Wörn, H. (2007). Swing attenuation of suspended objects transported by robot manipulator using acceleration compensation. In *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems* (pp. 2919–2924).
- Chen, X., Abraham, E., & Sankaranarayanan, S. (2012). Taylor model flowpipe construction for non-linear hybrid systems. In *Proc. of the IEEE 33rd Real-Time Systems Symposium* (pp. 183–192).
- Chilali, M. & Gahinet, P. (1996). \mathcal{H}_∞ design with pole placement constraints: an lmi approach. *IEEE Transactions on Automatic Control*, 41(3), 358–367.
- Chisci, L., Garulli, A., & Zappa, G. (1996). Recursive state bounding by parallelotopes. *Automatica*, 32(7), 1049–1055.
- Chizeck, H. J. & Willsky, A. S. (1978). Towards fault-tolerant optimal control. In *Proc. of the 1978 IEEE Conference on Decision and Control* (pp. 19–20).
- Combastel, C. (2003). A state bounding observer based on zonotopes. In *2003 European Control Conference* (pp. 2589–2594).

- Combastel, C. (2005). A state bounding observer for uncertain non-linear continuous-time systems based on zonotopes. In *Proc. of the 44th IEEE Conference on Decision and Control, and 2005 European Control Conference* (pp. 7228–7234).
- Combastel, C. (2015a). Merging Kalman filtering and zonotopic state bounding for robust fault detection under noisy environment. In *Proc. of the 9th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes* (pp. 289–295).
- Combastel, C. (2015b). Zonotopes and Kalman observers: Gain optimality under distinct uncertainty paradigms and robust convergence. *Automatica*, 55, 265–273.
- Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). *Introduction to Algorithms*. Cambridge, 3rd edition.
- Cugueró-Escofet, P., Blesa, J., Pérez, R., Cugueró-Escofet, M. A., & Sanz, G. (2015). Assessment of leak localization algorithm in water networks under demand uncertainty. In *Proc. of the 9th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes* (pp. 226–231).
- Cui, T., Ding, F., Jin, X.-B., Alsaedi, A., & Hayat, T. (2020). Joint multi-innovation recursive extended least squares parameter and state estimation for a class of state-space systems. *International Journal of Control, Automation and Systems*, 18(6), 1412–1424.
- D’Amato, A. M., Ali, A., Springmann, J. C., Cutler, J. W., Ridley, A. J., & Bernstein, D. S. (2011). Adaptive state estimation for nonminimum-phase systems with uncertain harmonic inputs. In *Proc. of 2011 AIAA Guidance, Navigation and Control Conference* (pp. 1–18).
- Dang, T. & Testylier, R. (2012). Reachability analysis for polynomial dynamical systems using the bernstein expansion. *Reliable Computing*, 17, 128–152.
- Denis-Vidal, L., Jauberthie, C., & Kieffer, M. (2019). Optimal experiment design for bounded-error estimation of nonlinear models. In *Proc. of the 58th Conference on Decision and Control* (pp. 4147–4154).
- Di Domenico, D., Stefanopoulou, A., & Fiengo, G. (2010). Lithium-ion battery state of charge and critical surface charge estimation using an electrochemical model-based extended kalman filter. *Journal of Dynamic Systems, Measurement, and Control*, 132(6).
- Díaz, S., González, J., & Mínguez, R. (2016). Uncertainty evaluation for constrained state estimation in water distribution systems. *Journal of Water Resources Planning and Management*, 142(12), 06016004.

- Díaz, S., Mínguez, R., & González, J. (2017). Topological observability analysis in water distribution systems. *Journal of Water Resources Planning and Management*, 143(5), 06017001.
- Díaz, S., Mínguez, R., González, J., & Savic, D. (2018). Explicit expressions for state estimation sensitivity analysis in water systems. *Journal of Water Resources Planning and Management*, 144(4).
- Dobkin, D., Hershberger, J., Kirkpatrick, D., & Suri, S. (1993). Computing the intersection-depth of polyhedra. *Algorithmica*, 9, 518–533.
- Doyle, M., Fuller, T. F., & Newman, J. (1993). Modeling of galvanostatic charge and discharge of the lithium/polymer/insertion cell. *Journal of the Electrochemical Society*, 140(6), 1526–1533.
- Durieu, C., Walter, E., & Polyak, B. (2001). Multi-input multi-output ellipsoidal state bounding. *Journal of Optimization Theory and Applications*, 111(2), 273–303.
- Ecker, M., Käbitz, S., Laresgoiti, I., & Sauer, D. U. (2015). Parameterization of a physico-chemical model of a lithium-ion battery: II. model validation. *Journal of the Electrochemical Society*, 162(9), A1849.
- Efimov, D., Perruquetti, W., Raissi, T., & Zolghadri, A. (2013). Interval Observers for Time-Varying Discrete-Time Systems. *IEEE Transactions on Automatic Control*, 58(12), 3218–3224.
- Efimov, D., Polyakov, A., & Richard, J.-P. (2015). Interval observer design for estimation and control of time-delay descriptor systems. *European Journal of Control*, 23, 26–35.
- Eliades, D. G., Stavrou, D., Vrachimis, S. G., Panayiotou, C. G., & Polycarpou, M. M. (2015). Contamination event detection using multi-level thresholds. In *Computing and Control for the Water Industry CCWI2015* (pp. 1429–1438).
- Eras-Herrera, W. Y., Mesquita, A. R., & Teixeira, B. O. S. (2019). Equality-constrained state estimation for hybrid systems. *IET Control Theory and Applications*, 13(13), 2018–2028.
- Fekih, A. (2014). Fault diagnosis and fault tolerant control design for aerospace systems: a bibliographical review. In *Proc. of the American Control Conference* (pp. 1286–1291).
- Frehse, G., Bogomolov, S., & Greitschus, M. (2015). Eliminating spurious transitions in reachability with support functions. In *Proc. of the 18th International Conference on Hybrid Systems: Computation and Control* (pp. 149–158).

- Ghorbal, K., Goubault, E., & Putot, S. (2010). A logical product approach to zonotope intersection. In *Proc. of International Conference on Computer Aided Verification* (pp. 212–226).
- Girard, A. & Guernic, C. L. (2008). Efficient reachability analysis for linear systems using support functions. In *Proc. of the 17th IFAC World Congress* (pp. 8966–8971).
- Girard, A., Guernic, C. L., & Maler, O. (2006). Efficient computation of reachable sets of linear time-invariant systems with inputs. In *Proc. of the 2006 International Workshop on Hybrid Systems: Computation and Control* (pp. 257–271).
- Goodarzi, F. A. & Lee, T. (2017). Global formulation of an extended Kalman filter on SE(3) for geometric control of a quadrotor UAV. *Journal of Intelligent and Robotic Systems*, 88(2), 395–413.
- Greyvenstein, B. & van Zyl, J. E. (2007). An experimental investigation into the pressure - leakage relationship of some failed water pipes. *Journal of Water Supply: Research and Technology-Aqua*, 56(2), 117–124.
- Guerra, P., Puig, V., & Witczak, M. (2008). Robust fault detection with unknown-input interval observers using zonotopes. In *Proc. of 17th IFAC World Congress* (pp. 5557–5562).
- Hagemann, W. (2015). Efficient geometric operations on convex polyhedra, with an application to reachability analysis of hybrid systems. *Mathematics in Computer Science*, 9(3), 283–325.
- Hamdi, H., Rodrigues, M., Mechmeche, C., & Braiek, N. B. (2012). Robust fault detection and estimation for descriptor systems based on multi-models concept. *International Journal of Control, Automation, and Systems*, 10(6), 1260–1266.
- Hansen, E. & Walster, G. W. (2003). *Global optimization using interval analysis: revised and expanded*. CRC Press.
- Hatami, E., Vosoughi, N., & Salaireh, H. (2016). Design of a fault tolerated intelligent control system for load following operation in a nuclear power plant. *Electrical Power and Energy Systems*, 78, 864–872.
- Herceg, M., Kvasnica, M., Jones, C., & Morari, M. (2013). Multi-Parametric Toolbox 3.0. In *Proc. of the 2013 European Control Conference* (pp. 502–510). <http://control.ee.ethz.ch/~mpt>.
- Herrero, P., Delaunay, B., Jaulin, L., Georgiou, P., Oliver, N., & Toumazou, C. (2016). Robust set-membership parameter estimation of the glucose minimal model. *International Journal of Adaptive Control and Signal Processing*, 30(2), 173–185.

- Hsieh, C.-S. (2002). Performance gain margins of the two-stage lq reliable control. *Automatica*, 38(11), 1985–1990.
- Hu, Q. (2010). Robust adaptive sliding-mode fault-tolerant control with L_2 -gain performance for flexible spacecraft using redundant reaction wheels. *IET Control Theory and Applications*, 4(6), 1055–1070.
- Hu, X., Cao, D., & Egardt, B. (2018). Condition monitoring in advanced battery management systems: Moving horizon estimation using a reduced electrochemical model. *IEEE/ASME Transactions on Mechatronics*, 23(1), 167–178.
- IBM (2018). ILOG CPLEX V12.8 user’s manual for CPLEX.
- Jain, R. P. K. (2015). Transportation of a cable suspended load using unmanned aerial vehicles. Master’s thesis, Delft University of Technology.
- Janschek, K. (2011). *Mechatronic systems design: methods, models, concepts*. Springer Science & Business Media.
- Jaulin, L. (2009a). A nonlinear set membership approach for the localization and map building of underwater robotics. *IEEE Transactions on Robotics*, 25(1), 88–98.
- Jaulin, L. (2009b). Robust set-membership state estimation; application to underwater robotics. *Automatica*, 45(1), 202–206.
- Jaulin, L. (2012). Combining interval analysis with flatness theory for state estimation of sailboat robots. *Mathematics in Computer Science*, 6(4), 347–359.
- Jaulin, L. (2016). Inner and outer set-membership state estimation. *Reliable Computing*, 22, 47–55.
- Jaulin, L. & Bars, F. L. (2013). An interval approach for stability analysis: Application to sailboat robotics. *IEEE Transactions on Robotics*, 29(1), 282–287.
- Jaulin, L., Braems, I., Kieffer, M., & Walter, E. (2001a). Nonlinear state estimation using forward-backward propagation of intervals in an algorithm. In *Scientific Computing, Validated Numerics, Interval Methods* (pp. 191–201). Springer.
- Jaulin, L., Kieffer, M., Didrit, O., & Éric Walter (2001b). *Applied Interval Analysis*. Springer.
- Johnson, M. A. & Grimble, M. J. (1987). Recent trends in linear optimal quadratic multivariable control system design. In *IEE Proceedings D - Control Theory and Applications*, volume 134 (pp. 53–71).

- Jonckheere, E. (1988). Variational calculus for descriptor systems. *IEEE Transactions on Automatic Control*, 33(5), 491–495.
- Julier, S. J. & LaViola, J. J. (2010). On kalman filtering with nonlinear equality constraints. *IEEE Transactions on Signal Processing*, 55(6), 2774–2784.
- Kang, D. & Lansey, K. (2010). Real-time optimal valve operation and booster disinfection for water quality in water distribution systems. *Journal of Water Resources Planning and Management*, 136(4), 463–473.
- Kelner, J. A. & Spielman, D. A. (2006). A randomized polynomial-time simplex algorithm for linear programming. In *Proc. of the 38th Annual ACM Symposium on Theory of Computing* (pp. 51–60).
- Kieffer, M., Jaulin, L., & Walter, E. (1998). Guaranteed nonlinear state estimation using interval analysis. In *Proc. of the 37th Conference on Decision and Control* (pp. 3966–3971).
- Kim, I.-S. (2009). A technique for estimating the state of health of lithium batteries through a dual-sliding-mode observer. *IEEE Transactions on Power Electronics*, 25(4), 1013–1022.
- Kingdom, B., Liemberger, R., & Marin, P. (2006). *The Challenge of Reducing Non-Revenue Water (NRW) in Developing Countries—How the Private Sector Can Help: A Look at Performance-Based Service Contracting*. Dec. WSS Sector Board Discussion Paper 8, World Bank.
- Kochdumper, N. & Althoff, M. (2020a). Constrained Polynomial Zonotopes. <https://arxiv.org/abs/2005.08849>.
- Kochdumper, N. & Althoff, M. (2020b). Reachability analysis for hybrid systems with nonlinear guard sets. In *Proc. of the 23rd International Conference on Hybrid Systems: Computation and Control* (pp. 1–10).
- Kochdumper, N. & Althoff, M. (2020c). Sparse polynomial zonotopes: A novel set representation for reachability analysis. *IEEE Transactions on Automatic Control*. In press.
- Kühn, W. (1998). Rigorously computed orbits of dynamical systems without the wrapping effect. *Computing*, 61(1), 47–67.
- Kurzanski, A. B. & Varaiya, P. (2000). Ellipsoidal techniques for reachability analysis. In *Proc. of the 2000 International Workshop on Hybrid Systems: Computation and Control* (pp. 202–214).

- Kvasnica, M., Grieder, P., Baotić, M., & Morari, M. (2004). Multi-parametric toolbox (MPT). In *International Workshop on Hybrid Systems: Computation and Control* (pp. 448–462).: Springer.
- Lara, A. V., Rego, B. S., Raffo, G. V., & Arias-Garcia, J. (2017). Desenvolvimento de um ambiente de simulação de VANTs tilt-rotor para testes de estratégias de controle. In *Anais do XIII Simpósio Brasileiro de Automação Inteligente* (pp. 2135–2141).
- Le, V. T. H., Stoica, C., Alamo, T., Camacho, E. F., & Dumur, D. (2013). Zonotopic guaranteed state estimation for uncertain systems. *Automatica*, 49(11), 3418–3424.
- Le, V. T. H., Stoica, C., Dumur, D., Alamo, T., & Camacho, E. F. (2011). Robust tube-based constrained predictive control via zonotopic set-membership estimation. In *Proc. of the 50th IEEE Conference on Decision and Control and 2011 European Control Conference* (pp. 4580–4585).
- Lefferts, E. J., Markley, F. L., & Shuster, M. D. (1982). Kalman filtering for spacecraft attitude estimation. *Journal of Guidance, Control, and Dynamics*, 5(5), 417–429.
- Li, H., Liu, H., Gao, H., & Shi, P. (2012). Reliable fuzzy control for active suspension systems with actuator delay and fault. *IEEE Transactions on Fuzzy Systems*, 20(2), 342–357.
- Locatelli, D., Rego, B. S., Raffo, G. V., & Raimondo, D. M. (2021). Interval state estimation based on constraint propagation for a lithium-ion cell using an equivalent circuit model. In *Proc. of the 11th IFAC International Symposium on Advanced Control of Chemical Processes*. Accepted.
- Lu, L., Han, X., Li, J., Hua, J., & Ouyang, M. (2013). A review on the key issues for lithium-ion battery management in electric vehicles. *Journal of Power Sources*, 226, 272–288.
- Marseglia, G. R. & Raimondo, D. M. (2017). Active fault diagnosis: A multi-parametric approach. *Automatica*, 79, 223–230.
- Marseglia, G. R., Raimondo, D. M., Magni, L., & Mesbah, A. (2017). A probabilistic framework for reference design for guaranteed fault diagnosis under closed-loop control. In *Proc. of IEEE 56th Conference on Decision and Control* (pp. 5739–5744).
- Mazenc, F., Dinh, T. N., & Niculescu, S. (2013). Robust interval observers and stabilization design for discrete-time systems with input and output. *Automatica*, 49(11), 3490–3497.
- Merhy, D., Maniu, C. S., Alamo, T., & Camacho, E. F. (2019). Ellipsoidal set-membership state estimation for descriptor systems. In *Proc. of the 23rd International Conference on System Theory, Control and Computing* (pp. 1–6).

- Mistler, V., Benallegue, A., & M'Sirdi, N. K. (2001). Exact linearization and noninteracting control of a 4 rotors helicopter via dynamic feedback. In *Proc. of the 10th IEEE International Workshop on Robot and Human Interactive Communication* (pp. 586–593).
- Mitchell, I. M., Bayen, A. M., & Tomlin, C. J. (2005). A time-dependent hamilton?jacobi formulation of reachable sets for continuous dynamic games. *IEEE Transactions on Automatic Control*, 50(7), 947–957.
- Montgomery, R. C. & Caglayan, A. K. (1976). Failure accommodation in digital flight control systems by bayesian decision theory. *Journal of Aircraft*, 13(2), 69–75.
- Montgomery, R. C. & Price, D. B. (1976). Failure accommodation in digital flight control systems accounting for nonlinear aircraft dynamics. *Journal of Aircraft*, 13(2), 76–82.
- Moore, R. E. (1966). *Interval Analysis*. Prentice-Hall Englewood Cliffs.
- Moore, R. E., Kearfott, R. B., & Cloud, M. J. (2009). *Introduction to Interval Analysis*. Philadelphia, PA, USA: SIAM.
- Moura, S. J., Argomedeo, F. B., Klein, R., Mirtabatabaei, A., & Krstic, M. (2016). Battery state estimation for a single particle model with electrolyte dynamics. *IEEE Transactions on Control Systems Technology*, 25(2), 453–468.
- Mukkula, A. R. G. & Paulen, R. (2017). Model-based design of optimal experiments for nonlinear systems in the context of guaranteed parameter estimation. *Computers and Chemical Engineering*, 99, 198–213.
- Nagar, A. K. & Powell, R. S. (2000). Observability analysis of water distribution systems under parametric and measurement uncertainty. In *Joint Conference on Water Resource Engineering and Water Resources Planning and Management 2000: Building Partnerships* (pp. 1–10). Reston, VA: American Society of Civil Engineers.
- Neuland, R., Meffei, R., Jaulin, L., Prestes, E., & Kolberg, M. (2014). Improving the precision of auvs localization in a hybrid interval-probabilistic approach using a set-inversion strategy. *Unmanned Systems*, 2(4), 361–375.
- Ngo, Q. H. & Hong, K.-S. (2012). Sliding-mode antisway control of an offshore container crane. *IEEE/AMSE Transactions on Mechatronics*, 17(2), 201–209.
- Park, S., Bae, J., Kim, Y., & Kim, S. (2013). Fault tolerant flight control system for the tilt-rotor uav. *Journal of the Franklin Institute*, 350(9), 2535–2559.
- Paulson, J. A., Raimondo, D. M., Findeisen, R., Braatz, R. D., & Streif, S. (2014). Guaranteed active fault diagnosis for uncertain nonlinear systems. In *Proc. of the 2014 European Control Conference* (pp. 926–931).

- Perez, H. E., Hu, X., Dey, S., & Moura, S. J. (2017). Optimal charging of Li-ion batteries with coupled electro-thermal-aging dynamics. *IEEE Transactions on Vehicular Technology*, 66(9), 7761–7770.
- Piller, S., Perrin, M., & Jossen, A. (2001). Methods for state-of-charge determination and their applications. *Journal of Power Sources*, 96(1), 113–120.
- Ping, X. & Sun, N. (2012). Dynamic output feedback robust model predictive control via zonotopic set-membership estimation for constrained quasi-lpv systems. *Journal of Applied Mathematics*, 2015.
- Platzer, A. & Clarke, E. M. (2007). The image computation problem in hybrid systems model checking. In *Proc. of the International Workshop on Hybrid Systems: Computation and Control* (pp. 473–486).
- Plett, G. L. (2006). Sigma-point kalman filtering for battery management systems of lipb-based hev battery packs: Part 2: Simultaneous state and parameter estimation. *Journal of Power Sources*, 161(2), 1369–1384.
- Polyak, B. T., Nazin, S. A., Durieu, C., & Walter, E. (2004). Ellipsoidal parameter or state estimation under model uncertainty. *Automatica*, 40(7), 1171–1179.
- Polyakov, A., Poznyak, A., & Richard, J.-P. (2013). Robust output stabilization on time-varying input delay systems using attractive ellipsoid method. In *Proc. of the 52nd IEEE Conference on Decision and Control* (pp. 934–939).
- Ponsini, O., Michel, C., & Rueher, M. (2012). Refining abstract interpretation based value analysis with constraint programming techniques. In *Proc. of International Conference on Principles and Practice of Constraint Programming* (pp. 593–607).
- Pozzi, A., Ciaramella, G., Volkwein, S., & Raimondo, D. M. (2018). Optimal design of experiments for a lithium-ion cell: parameters identification of an isothermal single particle model with electrolyte dynamics. *Industrial & Engineering Chemistry Research*, 58(3), 1286–1299.
- Pozzi, A., Zambelli, M., Ferrara, A., & Raimondo, D. M. (2020). Balancing-aware charging strategy for series-connected lithium-ion cells: A nonlinear model predictive control approach. *IEEE Transactions on Control Systems Technology*, 28(5), 1862–1877.
- Raffo, G. V. (2011). Robust control strategies for a quadrotor helicopter: An underactuated mechanical system.
- Raffo, G. V. & Almeida, M. M. (2016). Nonlinear robust control of a quadrotor UAV for load transportation with swing improvement. In *American Control Conference* (pp. 3156–3162).

- Raimondo, D. M., Braatz, R. D., & Scott, J. K. (2013). Active fault diagnosis using moving horizon input design. In *Proc. of the 2013 European Control Conference* (pp. 3131–3136).
- Raimondo, D. M., Marseglia, G. R., Braatz, R. D., & Scott, J. K. (2016). Closed-loop input design for guaranteed fault diagnosis using set-valued observers. *Automatica*, 74, 107–117.
- Raimondo, D. M., Rivero, S., Summers, S., Jones, C. N., Lygeros, J., & Morari, M. (2012). A set-theoretic method for verifying feasibility of a fast explicit nonlinear model predictive controller. In R. Johansson & A. Rantzer (Eds.), *Distributed Decision Making and Control* (pp. 289–311). London: Springer London.
- Raïssi, T., Efimov, D., & Zolghadri, A. (2012). Interval state estimation for a class of nonlinear systems. *IEEE Transactions on Automatic Control*, 57(1), 260–265.
- Raïssi, T., Ramdani, N., & Candau, Y. (2004). Set membership state and parameter estimation for systems described by nonlinear differential equations. *Automatica*, 40(10), 1771–1777.
- Rego, B. S., Locatelli, D., Raimondo, D. M., & Raffo, G. V. (2021a). Set-based joint state and parameter estimation of discrete-time systems using constrained zonotopes. To be submitted.
- Rego, B. S. & Raffo, G. V. (2016). Suspended load path tracking control based on zonotopic state estimation using a tilt-rotor UAV. In *Proc. of the IEEE 19th International Conference on Intelligent Transportation Systems* (pp. 1445–1451).
- Rego, B. S. & Raffo, G. V. (2019). Suspended load path tracking control using a tilt-rotor UAV based on zonotopic state estimation. *Journal of the Franklin Institute*, 356(4), 1695–1729.
- Rego, B. S., Raffo, G. V., Scott, J. K., & Raimondo, D. M. (2020a). Guaranteed methods based on constrained zonotopes for set-valued state estimation of nonlinear discrete-time systems. *Automatica*, 111, 108614.
- Rego, B. S., Raimondo, D. M., & Raffo, G. V. (2018a). Path tracking control with state estimation based on constrained zonotopes for aerial load transportation. In *Proc. of the 57th IEEE Conference on Decision and Control* (pp. 1979–1984).
- Rego, B. S., Raimondo, D. M., & Raffo, G. V. (2018b). Set-based state estimation of nonlinear systems using constrained zonotopes and interval arithmetic. In *Proc. of the 2018 European Control Conference* (pp. 1584–1589).

- Rego, B. S., Raimondo, D. M., & Raffo, G. V. (2020b). Set-based state estimation and fault diagnosis of linear discrete-time descriptor systems using constrained zonotopes. In *Proc. of the 1st IFAC-V World Congress* (pp. 4291–4296).
- Rego, B. S., Raimondo, D. M., & Raffo, G. V. (2021b). Set-based joint state and parameter estimation of nonlinear systems with nonlinear algebraic constraints using constrained zonotopes. To be submitted.
- Rego, B. S., Raimondo, D. M., & Raffo, G. V. (2021c). Set-based state estimation and active fault diagnosis of linear descriptor systems using mixed zonotopes. To be submitted.
- Rego, B. S., Scott, J. K., Raimondo, D. M., & Raffo, G. V. (2021d). Set-valued state estimation of nonlinear discrete-time systems with nonlinear invariants based on constrained zonotopes. *Automatica*, 129, 109638.
- Rego, B. S., Vrachimis, S. F., Polycarpou, M. M., Raffo, G. V., & Raimondo, D. M. (2021e). State estimation and leakage detection in water distribution networks using constrained zonotopes. *IEEE Transactions on Control Systems Technology*. Minor revision after first round.
- Rohou, S., Jaulin, L., Mihaylova, L., Bars, F. L., & Veres, S. M. (2018). Reliable non-linear state estimation involving time uncertainties. *Automatica*, 93, 379–388.
- Rossman, L. A. (2000). Epanet 2: users manual.
- Rotella, N., Bloesch, M., Righetti, L., & Schaal, S. (2014). State estimation for a humanoid robot. In *Proc. of the 2014 IEEE/RSJ International Conference Intelligent Robots and Systems* (pp. 952–958).
- Rump, S. M. (1999). INTLAB — INTerval LABoratory. In T. Csendes (Ed.), *Developments in Reliable Computing* (pp. 77–104). Dordrecht: Springer Netherlands.
- Rumschinski, P., Borchers, S., Bosio, S., Weismantel, R., & Findeisen, R. (2010). Set-based dynamical parameter estimation and model invalidation for biochemical reaction networks. *BMC Systems Biology*, 4(69), 1–14.
- Saeedi, S., Trentini, M., Seto, M., & Li, H. (2016). Multiple-robot simultaneous localization and mapping: A review. *Journal of Field Robotics*, 33(1), 3–46.
- Santos, M. A., Cardoso, D. N., Rego, B. S., Raffo, G. V., & Esteban, S. (2017a). A discrete robust adaptive control of a tilt-rotor UAV for an enlarged flight envelope. In *Proc. of the 56th IEEE Conference on Decision and Control* (pp. 5208–5214).

- Santos, M. A., Rego, B. S., Raffo, G. V., & Ferramosca, A. (2017b). Suspended load path tracking control strategy using a tilt-rotor UAV. *Journal of Advanced Transportation*, 2017, 1–22.
- Schweppe, F. (1968). Recursive state estimation: Unknown but bounded errors and system inputs. *IEEE Transactions on Automatic Control*, 13(1), 22–28.
- Scott, J. K., Findeisen, R., Braatz, R. D., & Raimondo, D. M. (2013a). Design of active inputs for set-based fault diagnosis. In *Proc. of the 2013 American Control Conference* (pp. 3561–3566).
- Scott, J. K., Findeisen, R., Braatz, R. D., & Raimondo, D. M. (2014). Input design for guaranteed fault diagnosis using zonotopes. *Automatica*, 50(6), 1580–1589.
- Scott, J. K., Marseglia, G. R., Magni, L., Braatz, R. D., & Raimondo, D. M. (2013b). A hybrid stochastic-deterministic input design method for active fault diagnosis. In *Proc. of the 52nd IEEE Conference on Decision and Control* (pp. 5656–5661).
- Scott, J. K., Raimondo, D. M., Marseglia, G. R., & Braatz, R. D. (2016). Constrained zonotopes: a new tool for set-based estimation and fault detection. *Automatica*, 69, 126–136.
- Shamma, J. S. & Tu, K.-Y. (1999). Set-valued observers and optimal disturbance rejection. *IEEE Transactions on Automatic Control*, 44(2), 253–264.
- Shen, K. & Scott, J. K. (2017). Rapid and accurate reachability analysis for nonlinear dynamic systems by exploiting model redundancy. *Computers & Chemical Engineering*, 106, 596–608.
- Shen, Q., Wang, D., Zhu, S., & Poh, E. K. (2015). Integral-type sliding mode fault-tolerant control for attitude stabilization of spacecraft. *IEEE Transactions on Control Systems Technology*, 23(3), 1131–1138.
- Shi, F. & Patton, R. (2014). Active fault tolerant control of lpv descriptor systems based on extended state observers. In *Proc. of the 2014 American Control Conference* (pp. 4458–4463).
- Simon, D. (2006). *Optimal State Estimation: Kalman, H_∞ and Nonlinear Approaches*. John Wiley & Sons, Inc.
- Simon, D. (2010). Kalman filtering with state constraints: a survey of linear and nonlinear algorithms. *IET Control Theory and Applications*, 4(8), 1303–1318.

- Sjöberg, J. & Glad, T. (2005). Power series solution of the Hamilton-Jacobi-Bellman equation for descriptor systems. In *Proc. of the 44th Conference on Decision and Control* (pp. 6869–6874).
- Sreenath, K., Lee, T., & Kumar, V. (2013). Geometric control and differential flatness of a quadrotor UAV with a cable-suspended load. In *52nd IEEE Conference on Decision and Control* (pp. 2269–2274).
- Subramanian, V. R., Diwakar, V. D., & Tapriyal, D. (2005). Efficient macro-micro scale coupled modeling of batteries. *Journal of the Electrochemical Society*, 152(10), A2002.
- Tabatabaeipour, S. M. (2015). Active fault detection and isolation of discrete-time linear time-varying systems: a set-membership approach. *International Journal of Systems Science*, 46(11), 1917–1933.
- Tao, Y., Shen, D., Wang, Y., & Ye, Y. (2015). Reliable \mathcal{H}_∞ control for uncertain nonlinear discrete-time systems subject to multiple intermittent faults in sensors and/or actuators. *Journal of the Franklin Institute*, 352(11), 4721–4740.
- Teixeira, B. O. S., Chandrasekar, J., Tôrres, L. A. B., Aguirre, L. A., & Bernstein, D. S. (2009). State estimation for linear and non-linear equality-constrained systems. *International Journal of Control*, 82(5), 918–936.
- Tulsyan, A., Tsai, Y., Gopaluni, R. B., & Braatz, R. D. (2016). State-of-charge estimation in lithium-ion batteries: A particle filter approach. *Journal of Power Sources*, 331, 208–223.
- Vento, J., Blesa, J., Puig, V., & Sarrate, R. (2015). Set-membership parity space hybrid system diagnosis. *International Journal of Systems Science*, 46(5), 790–807.
- Vicino, A. & Zappa, G. (1996). Sequential approximation of feasible parameter sets for identification with set membership uncertainty. *IEEE Transactions on Automatic Control*, 41(6), 774–785.
- Vrachimis, S. G., Eliades, D. G., & Polycarpou, M. M. (2018a). Leak Detection in Water Distribution Systems Using Hydraulic Interval State Estimation. In *IEEE Conference on Control Technology and Applications (CCTA)* (pp. 565–570).
- Vrachimis, S. G., Eliades, D. G., & Polycarpou, M. M. (2018b). Real-time hydraulic interval state estimation for water transport networks: a case study. *Drinking Water Engineering and Science*, 11(1), 19–24.
- Vrachimis, S. G., Lifshitz, R., Eliades, D. G., Polycarpou, M. M., & Ostfeld, A. (2020). Active Contamination Detection in Water-Distribution Systems. *Journal of Water Resources Planning and Management*, 146(4), 04020014.

- Vrachimis, S. G., Timotheou, S., Eliades, D. G., & Polycarpou, M. M. (2019). Iterative hydraulic interval state estimation for water distribution networks. *Journal of Water Resources Planning and Management*, 145(1), 04018087.
- Walter, E. & Piet-Lahanier, H. (1989). Exact recursive polyhedral description of the feasible parameter set for bounded-error models. *IEEE Transactions on Automatic Control*, 34(8), 911–915.
- Wan, J., Sharma, S., & Sutton, R. (2018). Guaranteed state estimation for nonlinear discrete-time systems via indirectly implemented polytopic set computation. *IEEE Transactions on Automatic Control*, 63(12), 4317–4322.
- Wang, H., Kolmanovsky, I., & Sun, J. (2017). Zonotope-based set-membership parameter identification of linear systems with additive and multiplicative uncertainties: A new algorithm. In *Proc. of the 2017 American Control Conference* (pp. 1481–1486).
- Wang, Y., Bevly, D. M., & Rajamani, R. (2015). Interval observer design for LPV systems with parametric uncertainty. *Automatica*, 60, 79–85.
- Wang, Y., Puig, V., & Cembrano, G. (2018a). Set-membership approach and kalman observer based on zonotopes for discrete-time descriptor systems. *Automatica*, 93, 435–443.
- Wang, Y., Puig, V., Xu, F., & Cembrano, G. (2019). Robust fault detection and isolation based on zonotopic unknown input observers for discrete-time descriptor systems. *Journal of the Franklin Institute*, 356, 5293–5314.
- Wang, Y., Wang, Z., Puig, V., & Cembrano, G. (2018b). Zonotopic set-membership state estimation for discrete-time descriptor LPV systems. *IEEE Transactions on Automatic Control*. In press.
- Weber, R. M., Spragg, R., Hoffmann, K., & Onori, S. (2019). Process noise quantification in kalman filters with application to electrochemical lithium-ion battery state estimation. In *Proc. of 2019 IEEE International Symposium on Industrial Electronics* (pp. 1995–2000).
- Wu, C., Qi, J., Song, D., Qi, X., Liu, T., & Han, J. (2015). Development of an unmanned helicopter automatic barrels transportation system. In *IEEE International Conference on Robotics and Automation* (pp. 4686–4691).
- Xiong, R., Cao, J., Yu, Q., He, H., & Sun, F. (2017). Critical review on the battery state of charge estimation methods for electric vehicles. *Ieee Access*, 6, 1832–1843.
- Xu, F., Puig, V., Ocampo-Martinez, C., Oлару, S., & Niculescu, S. (2014). Robust mpc for actuator-fault tolerance using set-based passive fault detection and active fault isolation. In *Proc. of the 2014 53rd Conference on Decision and Control* (pp. 4959–4964).

- Xu, F., Tan, J., Wang, X., Puig, V., Liang, B., & Yuan, B. (2016). A novel design of unknown input observers using set-theoretic methods for robust fault detection. In *Proc. of the 2016 American Control Conference* (pp. 5957–5961).
- Yang, C. & Blash, E. (2009). Kalman filtering with nonlinear state constraints. *IEEE Transactions on Aerospace and Electronic Systems*, 45(1), 70–84.
- Yang, F. & Li, Y. (2009). Set-membership filtering for discrete-time systems with nonlinear equality constraints. *IEEE Transactions on Automatic Control*, 54(10), 2480–2486.
- Yang, S., Xu, F., Wang, X., Yang, J., & Liang, B. (2019). Set-based active fault diagnosis for discrete-time linear descriptor systems. In *Proc. of the 2019 American Control Conference* (pp. 5045–5051).
- Yang, X. & Scott, J. K. (2018). Accurate set-based state estimation for nonlinear discrete-time systems using differential inequalities with model redundancy. In *Proc. of the 57th IEEE Conference on Decision and Control* (pp. 680–685).
- Yang, X. & Scott, J. K. (2018). A comparison of zonotope order reduction techniques. *Automatica*, 95, 378–384.
- Yang, X. & Scott, J. K. (2020). Accurate uncertainty propagation for discrete-time nonlinear systems using differential inequalities with model redundancy. *IEEE Transactions on Automatic Control*, 65(12), 5043–5057.
- Yu, W., Zamora, E., & Soria, A. (2016). Ellipsoid slam: a novel set membership method for simultaneous localization and mapping. *Autonomous Robots*, 40(1), 125–137.
- Zhang, D., Couto, L. D., Gill, P., Benjamin, S., Zeng, W., & Moura, S. J. (2020a). Interval observer for SOC estimation in parallel-connected Lithium-ion batteries. In *2020 American Control Conference* (pp. 1149–1154).
- Zhang, D., Couto, L. D., Park, S., Gill, P., & Moura, S. J. (2020b). Nonlinear state and parameter estimation for li-ion batteries with thermal coupling. In *Proc. of the 21st IFAC World Congress* (pp. 12658–12663). To appear.
- Zhang, Y. & Jiang, J. (2008). Bibliographical review on reconfigurable fault-tolerant control systems. *Annual Reviews in Control*, 32, 229–252.
- Zhou, Y. & Baras, J. S. (2015). Reachable set approach to collision avoidance for uavs. In *Proc. of the 54th IEEE Conference on Decision and Control* (pp. 5947–5952).



Derivation of computational complexities

This appendix details the derivation of the computational complexities of basic operations on zonotopes and constrained zonotopes, and the set-based state estimators presented in this doctoral thesis.

A.1 Zonotopes

A.1.1 Basic operations

A.1.1.1 Linear image

Let $Z = \{\mathbf{G}_z, \mathbf{c}_z\} \subset \mathbb{R}^n$, $\mathbf{R} \in \mathbb{R}^{n_r \times n}$, with $\mathbf{c}_z \in \mathbb{R}^n$, $\mathbf{G}_z \in \mathbb{R}^{n \times n_g}$, and consider the linear image $\mathbf{R}Z = \{\mathbf{R}\mathbf{G}_z, \mathbf{R}\mathbf{c}_z\}$. Then,

$$O(\mathbf{R}Z) = O(\mathbf{R}\mathbf{c}_z) + O(\mathbf{R}\mathbf{G}_z) = O(nn_r) + O(nn_g n_r) = O(nn_g n_r).$$

A.1.1.2 Minkowski sum

Let $Z = \{\mathbf{G}_z, \mathbf{c}_z\} \subset \mathbb{R}^n$, $W = \{\mathbf{G}_w, \mathbf{c}_w\} \subset \mathbb{R}^n$, with $\mathbf{c}_z \in \mathbb{R}^n$, $\mathbf{G}_z \in \mathbb{R}^{n \times n_g}$, $\mathbf{c}_w \in \mathbb{R}^n$, $\mathbf{G}_w \in \mathbb{R}^{n \times n_{g_w}}$, and consider the Minkowski sum

$$Z \oplus W = \left\{ \begin{bmatrix} \mathbf{G}_z & \mathbf{G}_w \end{bmatrix}, \mathbf{c}_z + \mathbf{c}_w \right\}.$$

Then,

$$O(Z \oplus W) = O(\mathbf{c}_z + \mathbf{c}_w) = O(n).$$

A.1.1.3 Generator reduction

Let $Z = \{\mathbf{G}, \mathbf{c}\} \subset \mathbb{R}^n$, with $\mathbf{c} \in \mathbb{R}^n$, $\mathbf{G} \in \mathbb{R}^{n \times n_g}$. Let $k_g \leq (n_g - n)$ denote the number of generators to be eliminated. The computational complexity of generator reduction of zonotopes is the same presented in [Scott et al. \(2016\)](#) and [Yang & Scott \(2018\)](#):

$$O(\text{eliminate } k_g \text{ generators from } Z) = O(n^2 n_g + k_g n_g n).$$

A.1.1.4 Interval hull

Let $Z = \{\mathbf{G}, \mathbf{c}\} \subset \mathbb{R}^n$, with $\mathbf{c} \in \mathbb{R}^n$, $\mathbf{G} \in \mathbb{R}^{n \times n_g}$. Consider the interval hull of Z given by $[\mathbf{c} - \boldsymbol{\zeta}, \mathbf{c} + \boldsymbol{\zeta}]$, where ([Kühn, 1998](#))

$$\zeta_i = \sum_{j=1}^{n_g} |\mathbf{G}_{ij}|.$$

Then,

$$\begin{aligned} O(\text{interval hull of } Z) &= O(\mathbf{c} - \boldsymbol{\zeta}) + O(\mathbf{c} + \boldsymbol{\zeta}) + nO\left(\sum_{j=1}^{n_g} |\mathbf{G}_{ij}|\right) = O(\mathbf{c} + \boldsymbol{\zeta}) + nO\left(\sum_{j=1}^{n_g} |\mathbf{G}_{ij}|\right) \\ &= O(n) + nO(n_g) = O(nn_g). \end{aligned}$$

A.1.1.5 Zonotope inclusion

Let $Z = \mathbf{p} \oplus \mathbf{M}B_\infty^{n_g}$, with $\mathbf{p} \in \mathbb{R}^n$, $\mathbf{M} \in \mathbb{I}\mathbb{R}^{n \times n_g}$. Consider the zonotope inclusion $\diamond(Z) = \mathbf{p} \oplus [\text{mid}(\mathbf{M}) \ \mathbf{P}]B_\infty^{n_g+n}$, where $P_{ii} = \sum_{j=1}^{n_g} \frac{1}{2} \text{diam}(M_{ij})$, $P_{ij} = 0$ for $i \neq j$ ([Alamo et al., 2005](#)). Then,

$$\begin{aligned} O(\diamond(Z)) &= O(\text{mid}(\mathbf{M})) + O(\mathbf{P}) = O(\text{mid}(\mathbf{M})) + nO(P_{ii}) \\ &= O(nn_g) + nO(n_g) = O(nn_g). \end{aligned}$$

A.1.2 ZMV

A.1.2.1 Prediction step

Let $\boldsymbol{\mu} : \mathbb{R}^n \times \mathbb{R}^{n_w} \rightarrow \mathbb{R}^n$ be continuously differentiable, and let $\nabla_x \boldsymbol{\mu}$ denote the gradient of $\boldsymbol{\mu}$ with respect to its first argument. Let $\hat{X}_{k-1} = \{\mathbf{G}_x, \mathbf{c}_x\} \subset \mathbb{R}^n$, and $W = \{\mathbf{G}_w, \mathbf{c}_w\} \subset \mathbb{R}^{n_w}$ with $\mathbf{c}_x \in \mathbb{R}^n$, $\mathbf{G}_x \in \mathbb{R}^{n \times n_g}$, $\mathbf{c}_w \in \mathbb{R}^{n_w}$, and $\mathbf{G}_w \in \mathbb{R}^{n_w \times n_{gw}}$. Let Z be a zonotope such that $\boldsymbol{\mu}(\mathbf{c}_x, W) \subseteq Z$, and let $\mathbf{M} \in \mathbb{I}\mathbb{R}^{n \times n_g}$ be an interval matrix satisfying $\nabla_x^T \boldsymbol{\mu}(\hat{X}_{k-1}, W) \mathbf{G}_x \subseteq \mathbf{M}$. According to Theorem 4 in [Alamo et al. \(2005\)](#),

$$\boldsymbol{\mu}(\hat{X}_{k-1}, W) \subseteq \bar{X}_k = Z \oplus \diamond(\mathbf{M}B_\infty^{n_g}),$$

in which \bar{X}_k has $n_g + n_{g_w} + 2n$ generators if Z is also computed by the mean value extension. Therefore,

$$\begin{aligned} O(\text{ZMV}_{\text{prediction}}) &= O(Z \oplus \diamond(\mathbf{MB}_{\infty}^{n_g})) + O(\diamond(\mathbf{MB}_{\infty}^{n_g})) + O(Z) + O(\text{interval hull of } \hat{X}_{k-1}, W) \\ &= O(n) + O(nn_g + n^2n_g) + O(n_w n_{g_w} + nn_w n_{g_w}) + O(nn_g + n_w n_{g_w}) \\ &= O(n^2n_g + nn_w n_{g_w}). \end{aligned}$$

A.1.2.2 Update step

Let $\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{D}_u\mathbf{u}_k + \mathbf{D}_v\mathbf{v}_k$, with $\mathbf{y}_k \in \mathbb{R}^{n_y}$, $\mathbf{x}_k \in \mathbb{R}^n$, $\mathbf{u}_k \in \mathbb{R}^{n_u}$, $\mathbf{v}_k \in \mathbb{R}^{n_v}$. Consider the zonotopes $\bar{X}_k = \{\mathbf{G}_x, \mathbf{c}_x\} \subset \mathbb{R}^n$, $V = \{\mathbf{G}_v, \mathbf{c}_v\} \subset \mathbb{R}^{n_v}$ with $\mathbf{c}_x \in \mathbb{R}^n$, $\mathbf{G}_x \in \mathbb{R}^{n \times \bar{n}_g}$, $\mathbf{c}_v \in \mathbb{R}^{n_v}$, and $\mathbf{G}_v \in \mathbb{R}^{n_v \times n_{g_v}}$. For simplicity, define $m_g = n_g + n_{g_w}$. Following the intersection method proposed in [Bravo et al. \(2006\)](#), the update step is computed iteratively using one row of \mathbf{y}_k at a time, as described below.

Algorithm: Update step

- (1) Assign $\tilde{X}_k = \{\tilde{\mathbf{G}}_x, \tilde{\mathbf{c}}_x\} \leftarrow \bar{X}_k$, $i \leftarrow 1$.
- (2) Assign $\tilde{y} \leftarrow i$ -th row of \mathbf{y}_k , $\mathbf{p}^T \leftarrow i$ -th row of \mathbf{C} , $\mathbf{d}_u^T \leftarrow i$ -th row of \mathbf{D}_u , $\mathbf{d}_v^T \leftarrow i$ -th row of \mathbf{D}_v .
- (3) Compute the strip $S \leftarrow \{\mathbf{x} : |\mathbf{p}^T\mathbf{x} - d| \leq \sigma\}$, where $d = \tilde{y} - \mathbf{d}_u^T\mathbf{u}_k - \mathbf{d}_v^T\mathbf{c}_v$, $\sigma = \sum_{\ell=1}^{n_{g_v}} \sum_{j=1}^{n_v} |(\mathbf{D}_v)_{i,j}(\mathbf{G}_v)_{j\ell}|$.
- (4) Compute the tight strip $\tilde{S} \leftarrow \{\mathbf{x} : |\mathbf{p}^T\mathbf{x} - \tilde{d}| \leq \tilde{\sigma}\}$, where

$$\tilde{\sigma} = \frac{1}{2}(\bar{\sigma}^U - \bar{\sigma}^L), \quad \tilde{d} = d + \frac{1}{2}(\bar{\sigma}^U + \bar{\sigma}^L),$$

$$\text{with } \bar{\sigma}^L = \max\{-\sigma, \mathbf{p}^T\tilde{\mathbf{c}}_x - \|\tilde{\mathbf{G}}_x^T\mathbf{p}\|_1 - d\}, \quad \bar{\sigma}^U = \min\{\sigma, \mathbf{p}^T\tilde{\mathbf{c}}_x + \|\tilde{\mathbf{G}}_x^T\mathbf{p}\|_1 - d\}.$$

- (5) Assign $\tilde{X}_k \leftarrow \tilde{X}_k \cap \tilde{S} = \{\mathbf{G}(j^*), \mathbf{c}(j^*)\}$, where

$$j^* = \arg \min_j \det(\mathbf{G}(j)\mathbf{G}(j)^T),$$

with $j \in \{0, 1, \dots, \bar{n}_g\}$,

$$\mathbf{c}(j) = \begin{cases} \tilde{\mathbf{c}}_x + \left(\frac{\tilde{d} - \mathbf{p}^T\tilde{\mathbf{c}}_x}{\mathbf{p}^T(\tilde{\mathbf{G}}_x)_{:,j}} \right) (\tilde{\mathbf{G}}_x)_{:,j} & \text{if } 1 \leq j \leq \bar{n}_g \text{ and } \mathbf{p}^T(\tilde{\mathbf{G}}_x)_{:,j} \neq 0 \\ \tilde{\mathbf{c}}_x & \text{otherwise} \end{cases}$$

$$\mathbf{G}(j) = \begin{cases} [\mathbf{g}_1^j \ \mathbf{g}_2^j \ \dots \ \mathbf{g}_{\bar{n}_g}^j] & \text{if } 1 \leq j \leq \bar{n}_g \text{ and } \mathbf{p}^T(\tilde{\mathbf{G}}_x)_{:,j} \neq 0 \\ \tilde{\mathbf{G}}_x & \text{otherwise} \end{cases}$$

$$\mathbf{g}_\ell^j = \begin{cases} (\tilde{\mathbf{G}}_x)_{:, \ell} - \left(\frac{\mathbf{p}^T (\tilde{\mathbf{G}}_x)_{:, \ell}}{\mathbf{p}^T (\tilde{\mathbf{G}}_x)_{:, j}} \right) (\tilde{\mathbf{G}}_x)_{:, j} & \text{if } \ell \neq j \\ \left(\frac{\tilde{\sigma}}{\mathbf{p}^T (\tilde{\mathbf{G}}_x)_{:, j}} \right) (\tilde{\mathbf{G}}_x)_{:, j} & \text{if } \ell = j \end{cases}$$

(6) If $i < n_y$, go to Step 2 and assign $i \leftarrow i + 1$. Otherwise, assign $\hat{X}_k \leftarrow \tilde{X}_k$.

The computational complexity is

$$\begin{aligned} O(\text{ZMV}_{\text{update}}) &= n_y O(\text{update with the } i\text{-th row of } \mathbf{y}_k) \\ &= n_y (O(\text{Step (3)}) + O(\text{Step (4)}) + O(\text{Step (5)})). \end{aligned}$$

But,

$$O(\text{Step (3)}) = O(1) + O(n_u) + O(n_v) + O(n_v n_{g_v}) = O(n_u + n_v n_{g_v}),$$

$$O(\text{Step (4)}) = O(1) + O(n) + O(n \bar{n}_g) = O(n \bar{n}_g),$$

$$O(\text{Step (5)}) = \bar{n}_g O(n^3) + \bar{n}_g O(n^2 \bar{n}_g) + \bar{n}_g O(n) + \bar{n}_g O(n \bar{n}_g) = O(n^3 \bar{n}_g + n^2 \bar{n}_g^2).$$

Since $\bar{n}_g = m_g + 2n$, hence

$$\begin{aligned} O(\text{ZMV}_{\text{update}}) &= n_y (O(n_u + n_v n_{g_v}) + O(n \bar{n}_g) + O(n^3 \bar{n}_g + n^2 \bar{n}_g^2 + n \bar{n}_g)) \\ &= n_y (O(n^3 \bar{n}_g + n^2 \bar{n}_g^2 + n_u + n_v n_{g_v})) \\ &= O(n_y (n^3 (m_g + n) + n^2 (m_g + n)^2 + n_u + n_v n_{g_v})). \end{aligned}$$

A.1.2.3 Order reduction

Let n_g denote the desired number of generators of \hat{X}_k after order reduction, i.e., \hat{X}_k must have the same complexity as \hat{X}_{k-1} . Order reduction is performed by eliminating k_g generators from \hat{X}_k . For simplicity, define $m_g = n_g + n_{g_w}$. The prediction and update steps of the ZMV lead to \hat{X}_k with $m_g + 2n$ generators. Therefore, $k_g = n_{g_w} + 2n$. Consequently,

$$\begin{aligned} O(\text{ZMV}_{\text{reduction}}) &= O(\text{eliminate } k_g \text{ generators from } \hat{X}_k) \\ &= O(n^2 (m_g + n) + k_g (m_g + n) n) \\ &= O(n^2 (m_g + n) + (n_{g_w} + n) (m_g + n) n) = O(n^2 (m_g + n) + n (n_{g_w} + n) (m_g + n)). \end{aligned}$$

A.1.3 ZFO

A.1.3.1 Prediction step

Let $\boldsymbol{\eta} : \mathbb{R}^m \rightarrow \mathbb{R}^n$ be of class \mathcal{C}^2 , and let $\mathbf{z} = (\mathbf{x}_{k-1}, \mathbf{w}_{k-1}) \in \mathbb{R}^m$ denote its argument, where $m = n + n_w$. Let $Z = \hat{X}_{k-1} \times W = \{\mathbf{G}, \mathbf{c}\} \subset \mathbb{R}^m$ be a zonotope with $m_g = n_g + n_{g_w}$ generators. For each $q = 1, 2, \dots, n$, let $\mathbf{Q}^{[q]} \in \mathbb{I}\mathbb{R}^{m \times m}$ and $\tilde{\mathbf{Q}}^{[q]} \in \mathbb{I}\mathbb{R}^{m_g \times m_g}$ be interval matrices satisfying $\mathbf{Q}^{[q]} \supseteq \mathbf{H}\boldsymbol{\eta}_q(Z)$ and $\tilde{\mathbf{Q}}^{[q]} \supseteq \mathbf{G}^T \mathbf{Q}^{[q]} \mathbf{G}$. Moreover, define $\tilde{\mathbf{c}}, \tilde{\mathbf{G}}, \tilde{\mathbf{G}}_d$ as in Theorem 3. Then, the method in [Combastel \(2005\)](#) ensures that

$$\boldsymbol{\eta}(Z) \subseteq \bar{X}_k = \boldsymbol{\eta}(\mathbf{c}) \oplus \nabla^T \boldsymbol{\eta}(\mathbf{c})(\mathbf{G}B_\infty^{m_g}) \oplus \tilde{\mathbf{c}} \oplus [\tilde{\mathbf{G}} \ \tilde{\mathbf{G}}_d] B_\infty^{\tilde{m}_g},$$

where $\tilde{m}_g = (1/2)m_g^2 + (3/2)m_g + n$, and therefore \bar{X}_k has $(1/2)m_g^2 + (5/2)m_g + n$ generators. Thus,

$$\begin{aligned} O(\text{CZFO}_{\text{prediction}}) &= O(\boldsymbol{\eta}(\mathbf{c}) \oplus \nabla^T \boldsymbol{\eta}(\mathbf{c})(\mathbf{G}B_\infty^{m_g}) \oplus \tilde{\mathbf{c}} \oplus [\tilde{\mathbf{G}} \ \tilde{\mathbf{G}}_d] B_\infty^{\tilde{m}_g}) \\ &= O(n) + O(\tilde{\mathbf{c}}) + O(\tilde{\mathbf{G}}) + O(\tilde{\mathbf{G}}_d) + nO(\tilde{\mathbf{Q}}^{[q]}) + O(\text{interval hull of } Z) \\ &= O(n) + O(nm_g) + O(nm_g^2) + O(nm_g^2) + nO(m^2m_g + mm_g^2) + O(mm_g) \\ &= O(n(m^2m_g + mm_g^2)). \end{aligned}$$

A.1.3.2 Update step

Let $\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{D}_u \mathbf{u}_k + \mathbf{D}_v \mathbf{v}_k$, with $\mathbf{y}_k \in \mathbb{R}^{n_y}$, $\mathbf{x}_k \in \mathbb{R}^n$, $\mathbf{u}_k \in \mathbb{R}^{n_u}$, and $\mathbf{v}_k \in \mathbb{R}^{n_v}$. Consider the zonotopes $\bar{X}_k = \{\mathbf{G}_x, \mathbf{c}_x\} \subset \mathbb{R}^n$ and $V = \{\mathbf{G}_v, \mathbf{c}_v\} \subset \mathbb{R}^{n_v}$, with $\mathbf{c}_x \in \mathbb{R}^n$, $\mathbf{G}_x \in \mathbb{R}^{n \times \bar{n}_g}$, $\mathbf{c}_v \in \mathbb{R}^{n_v}$, and $\mathbf{G}_v \in \mathbb{R}^{n_v \times n_{g_v}}$. For simplicity, define $m_g = n_g + n_{g_w}$. The ZFO also follows the intersection method proposed in [Bravo et al. \(2006\)](#). Therefore, the update step is equivalent to the ZMV update, but with $\bar{n}_g = (1/2)m_g^2 + (5/2)m_g + n$. Hence

$$\begin{aligned} O(\text{ZFO}_{\text{update}}) &= n_y (O(n^3 \bar{n}_g + n^2 \bar{n}_g^2 + n_u + n_v n_{g_v})) \\ &= O(n_y (n^3 (m_g^2 + n) + n^2 (m_g^2 + n)^2 + n_u + n_v n_{g_v})). \end{aligned}$$

A.1.3.3 Order reduction

Let n_g denote the desired number of generators of \hat{X}_k after order reduction, i.e., \hat{X}_k must have the same complexity as \hat{X}_{k-1} . Order reduction is performed by eliminating k_g generators from \hat{X}_k . For simplicity, define $m_g = n_g + n_{g_w}$. The prediction and update steps of ZFO lead to \hat{X}_k with $(1/2)m_g^2 + (5/2)m_g + n$ generators. Therefore, $k_g = (1/2)m_g^2 + (5/2)m_g + n - n_g$. Consequently,

$$\begin{aligned} O(\text{ZFO}_{\text{reduction}}) &= O(\text{eliminate } k_g \text{ generators from } \hat{X}_k) \\ &= O(n^2 (m_g^2 + m_g + n) + k_g (m_g^2 + m_g + n)n) \end{aligned}$$

$$\begin{aligned}
&= O(n^2(m_g^2 + n) + (m_g^2 + n)(m_g^2 + n)n) \\
&= O(n^2(m_g^2 + n) + n(m_g^2 + n)^2).
\end{aligned}$$

A.2 Constrained zonotopes

A.2.1 Basic operations

A.2.1.1 Linear image

Let $Z = \{\mathbf{G}_z, \mathbf{c}_z, \mathbf{A}_z, \mathbf{b}_z\} \subset \mathbb{R}^n$, $\mathbf{R} \in \mathbb{R}^{n_r \times n}$, with $\mathbf{c}_z \in \mathbb{R}^n$, $\mathbf{G}_z \in \mathbb{R}^{n \times n_g}$, $\mathbf{A}_z \in \mathbb{R}^{n_c \times n_g}$, $\mathbf{b}_z \in \mathbb{R}^{n_c}$, and consider the linear image $\mathbf{R}Z = \{\mathbf{R}\mathbf{G}_z, \mathbf{R}\mathbf{c}_z, \mathbf{A}_z, \mathbf{b}_z\}$. Then,

$$O(\mathbf{R}Z) = O(\mathbf{R}\mathbf{c}_z) + O(\mathbf{R}\mathbf{G}_z) = O(nn_r) + O(nn_g n_r) = O(nn_g n_r).$$

A.2.1.2 Minkowski sum

Let $Z = \{\mathbf{G}_z, \mathbf{c}_z, \mathbf{A}_z, \mathbf{b}_z\} \subset \mathbb{R}^n$, $W = \{\mathbf{G}_w, \mathbf{c}_w, \mathbf{A}_w, \mathbf{b}_w\} \subset \mathbb{R}^n$, with $\mathbf{c}_z \in \mathbb{R}^n$, $\mathbf{G}_z \in \mathbb{R}^{n \times n_g}$, $\mathbf{A}_z \in \mathbb{R}^{n_c \times n_g}$, $\mathbf{b}_z \in \mathbb{R}^{n_c}$, $\mathbf{c}_w \in \mathbb{R}^n$, $\mathbf{G}_w \in \mathbb{R}^{n \times n_{g_w}}$, $\mathbf{A}_w \in \mathbb{R}^{n_{c_w} \times n_{g_w}}$, $\mathbf{b}_w \in \mathbb{R}^{n_{c_w}}$, and consider the Minkowski sum

$$Z \oplus W = \left\{ \left[\mathbf{G}_z \quad \mathbf{G}_w \right], \mathbf{c}_z + \mathbf{c}_w, \begin{bmatrix} \mathbf{A}_z & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_w \end{bmatrix}, \begin{bmatrix} \mathbf{b}_z \\ \mathbf{b}_w \end{bmatrix} \right\}.$$

Then,

$$O(Z \oplus W) = O(\mathbf{c}_z + \mathbf{c}_w) = O(n).$$

A.2.1.3 Generalized intersection

Let $Z = \{\mathbf{G}_z, \mathbf{c}_z, \mathbf{A}_z, \mathbf{b}_z\} \subset \mathbb{R}^n$, $Y = \{\mathbf{G}_y, \mathbf{c}_y, \mathbf{A}_y, \mathbf{b}_y\}$, $\mathbf{R} \in \mathbb{R}^{n_r \times n}$, with $\mathbf{G}_z \in \mathbb{R}^{n \times n_g}$, $\mathbf{c}_z \in \mathbb{R}^n$, $\mathbf{A}_z \in \mathbb{R}^{n_c \times n_g}$, $\mathbf{b}_z \in \mathbb{R}^{n_c}$, $\mathbf{G}_y \in \mathbb{R}^{n_r \times n_{g_r}}$, $\mathbf{c}_y \in \mathbb{R}^{n_r}$, $\mathbf{A}_y \in \mathbb{R}^{n_{c_r} \times n_{g_r}}$, $\mathbf{b}_y \in \mathbb{R}^{n_{c_r}}$. Consider the generalized intersection

$$Z \cap_{\mathbf{R}} Y = \left\{ \left[\mathbf{G}_z \quad \mathbf{0} \right], \mathbf{c}_z, \begin{bmatrix} \mathbf{A}_z & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_y \\ \mathbf{R}\mathbf{G}_z & -\mathbf{G}_y \end{bmatrix}, \begin{bmatrix} \mathbf{b}_z \\ \mathbf{b}_y \\ \mathbf{c}_y - \mathbf{R}\mathbf{c}_z \end{bmatrix} \right\}.$$

Then,

$$\begin{aligned}
O(Z \cap_{\mathbf{R}} Y) &= O(\mathbf{R}\mathbf{G}_z) + O(-\mathbf{G}_y) + O(\mathbf{c}_y - \mathbf{R}\mathbf{c}_z) = O(nn_g n_r) + O(n_r n_{g_r}) + O(n_r + nn_r) \\
&= O(nn_g n_r + n_r n_{g_r}).
\end{aligned}$$

A.2.1.4 Generator reduction

Let $Z = \{\mathbf{G}, \mathbf{c}, \mathbf{A}, \mathbf{b}\} \subset \mathbb{R}^n$, with $\mathbf{c} \in \mathbb{R}^n$, $\mathbf{G} \in \mathbb{R}^{n \times n_g}$, $\mathbf{A} \in \mathbb{R}^{n_c \times n_g}$, $\mathbf{b} \in \mathbb{R}^{n_c}$. Let $k_g \leq n_g - (n + n_c)$ denote the number of generators to be eliminated. The computational complexity of generator reduction of constrained zonotopes is the same presented in [Scott et al. \(2016\)](#) and [Yang & Scott \(2018\)](#), with reduction operated over the lifted zonotope

$$Z^+ = \left\{ \begin{bmatrix} \mathbf{G} \\ \mathbf{A} \end{bmatrix}, \begin{bmatrix} \mathbf{c} \\ -\mathbf{b} \end{bmatrix} \right\}.$$

Therefore,

$$O(\text{eliminate } k_g \text{ generators from } Z) = O((n + n_c)^2 n_g + k_g n_g (n + n_c)).$$

A.2.1.5 Constraint elimination

Let $Z = \{\mathbf{G}, \mathbf{c}, \mathbf{A}, \mathbf{b}\} \subset \mathbb{R}^n$, with $\mathbf{c} \in \mathbb{R}^n$, $\mathbf{G} \in \mathbb{R}^{n \times n_g}$, $\mathbf{A} \in \mathbb{R}^{n_c \times n_g}$, $\mathbf{b} \in \mathbb{R}^{n_c}$. Let $k_c \leq n_c$ denote the number of constraints to be eliminated from Z . Following the constraint elimination algorithm in [Scott et al. \(2016\)](#), for each eliminated constraint the remaining constraints are first preconditioned through Gauss-Jordan elimination with full pivoting and then subjected to a rescaling procedure before the next constraint is eliminated. Therefore,

$$\begin{aligned} O(\text{eliminate 1 constraint from } Z) &= O(\text{pre-conditioning}) + O(\text{rescaling}) \\ &\quad + O(\text{eliminate the constraint}). \end{aligned}$$

From the Appendix in [Scott et al. \(2016\)](#), we have that

$$O(\text{pre-conditioning}) + O(\text{rescaling}) = O(n_c^2 n_g + n_c n_g^2).$$

To eliminate the constraint, one of the generators of Z is first chosen by minimizing an approximation of the Hausdorff distance between Z and the set obtained by constraint elimination. This requires solving (3.15) for each $j \in \{1, 2, \dots, n_g\}$, which has the reported complexity $O((n_g + n_c)^3)$, and computing the Hausdorff distance $H_j^* = \|\mathbf{G}\mathbf{d}_j^*\|_2 + \|\mathbf{d}_j^*\|_2$, with \mathbf{d}_j^* the solution of (3.15) for each j . This is done with the following complexity:

$$\begin{aligned} O(\text{eliminate the constraint}) &= O(\text{Hausdorff distance minimization}) + n_g O(H_j^*) \\ &= O((n_g + n_c)^3) + n_g O(nn_g) = O((n_g + n_c)^3 + nn_g^2). \end{aligned}$$

Then,

$$O(\text{eliminate 1 constraint from } Z) = O(n_c^2 n_g + n_c n_g^2) + O((n_g + n_c)^3 + nn_g^2) = O((n_g + n_c)^3 + nn_g^2).$$

Finally, the complexity of eliminating k_c constraints from Z can be bounded by

$$\begin{aligned} O(\text{eliminate } k_c \text{ constraints from } Z) &= k_c O(\text{eliminate 1 constraint from } Z) \\ &= O(k_c(n_g + n_c)^3 + k_c n n_g^2). \end{aligned}$$

A.2.1.6 Interval hull (Property 1)

Let $Z = \{\mathbf{G}, \mathbf{c}, \mathbf{A}, \mathbf{b}\} \subset \mathbb{R}^n$, with $\mathbf{c} \in \mathbb{R}^n$, $\mathbf{G} \in \mathbb{R}^{n \times n_g}$, $\mathbf{A} \in \mathbb{R}^{n_c \times n_c}$, $\mathbf{b} \in \mathbb{R}^{n_c}$. Consider the interval hull of Z computed as in Property 1. In the standard form, each LP in Property 1 has $N_d = n_g + 1$ decision variables and $N_c = 2n_g + n_c + 1$ constraints. Then,

$$O(\text{interval hull of } Z) = 2nO(N_d N_c^3) = O(nN_d N_c^3) = O(n(1+n_g)(1+2n_g+n_c)^3) = O(nn_g(n_g + n_c)^3).$$

A.2.1.7 CZ-inclusion (Theorem 1)

Let $X = \mathbf{p} \oplus \mathbf{M}B_\infty(\mathbf{A}, \mathbf{b}) \subset \mathbb{R}^m$, $\mathbf{J} \in \mathbb{I}\mathbb{R}^{n \times m}$, $\bar{X} = \bar{\mathbf{p}} \oplus \bar{\mathbf{M}}B_\infty^{n_g} \supseteq X$, $\mathbf{m} \supseteq (\mathbf{J} - \text{mid}(\mathbf{J}))\bar{\mathbf{p}}$, $\triangleleft(\mathbf{J}, X) = \text{mid}(\mathbf{J})X \oplus \mathbf{P}B_\infty^n$, with $\mathbf{p} \in \mathbb{R}^m$, $\mathbf{M} \in \mathbb{R}^{m \times n_g}$, $\mathbf{A} \in \mathbb{R}^{n_c \times n_g}$, $\mathbf{b} \in \mathbb{R}^{n_c}$, $P_{ii} = \frac{1}{2}\text{diam}(m_i) + \frac{1}{2}\sum_{j=1}^{n_g} \sum_{k=1}^m \text{diam}(J_{ik})|\bar{M}_{kj}|$, $P_{ij} = 0$ for $i \neq j$. Assume that \bar{X} is obtained by eliminating n_c constraints from X , and \mathbf{m} is computed using interval arithmetic. Then,

$$\begin{aligned} O(\triangleleft(\mathbf{J}, X)) &= O(\text{mid}(\mathbf{J})X \oplus \mathbf{P}B_\infty^n) + O((\mathbf{J} - \text{mid}(\mathbf{J}))\bar{\mathbf{p}}) + nO\left(\frac{1}{2}\text{diam}(m_i)\right. \\ &\quad \left. + \frac{1}{2}\sum_{j=1}^{n_g} \sum_{k=1}^m \text{diam}(J_{ik})|\bar{M}_{kj}|\right) + O(\text{eliminate } n_c \text{ constraints from } X) \\ &= O(nm + nmn_g + n) + O(nm + nm) + nO(\bar{n}_g m) + O(n_c(n_g + n_c)^3 + n_c mn_g^2) \\ &= O(nmn_g) + O(nm) + O(nmn_g) + O(n_c(n_g + n_c)^3 + mn_c n_g^2) \\ &= O(nmn_g + n_c(n_g + n_c)^3 + mn_g^2 n_c). \end{aligned}$$

A.2.1.8 Closest point (Proposition 1)

Let $Z = \{\mathbf{G}, \mathbf{c}, \mathbf{A}, \mathbf{b}\} \subset \mathbb{R}^n$, $\mathbf{h} \in \mathbb{R}^n$, with $\mathbf{c} \in \mathbb{R}^n$, $\mathbf{G} \in \mathbb{R}^{n \times n_g}$, $\mathbf{A} \in \mathbb{R}^{n_c \times n_g}$, $\mathbf{b} \in \mathbb{R}^{n_c}$. In the standard form, the LP in Proposition 1 has $N_d = n + n_g + 1$ decision variables and $N_c = 2n + 2n_g + n_c + 1$ constraints. Then,

$$\begin{aligned} O(\text{Proposition 1}) &= O(\mathbf{c} + \mathbf{G}\boldsymbol{\xi}^*) + O(N_d N_c^3) = O(n + nn_g) + O((n + n_g + 1)(2n + 2n_g + n_c + 1)^3) \\ &= O((n + n_g)(n + n_g + n_c)^3). \end{aligned}$$

A.2.1.9 Change center (Proposition 2)

Let

$$Z = \{\mathbf{G}, \mathbf{c}, \mathbf{A}, \mathbf{b}\} = \left\{ \left[\mathbf{G}\mathbf{E}_r \quad \mathbf{0} \right], \mathbf{h}, \begin{bmatrix} \mathbf{A}\mathbf{E}_r & \mathbf{0} \\ \mathbf{0} & \mathbf{A} \\ \mathbf{G}\mathbf{E}_r & -\mathbf{G} \end{bmatrix}, \begin{bmatrix} \mathbf{b} - \mathbf{A}\boldsymbol{\xi}_m \\ \mathbf{b} \\ \mathbf{c} - \mathbf{h} \end{bmatrix} \right\} \subset \mathbb{R}^n,$$

with $\mathbf{c} \in \mathbb{R}^n$, $\mathbf{G} \in \mathbb{R}^{n \times n_g}$, $\mathbf{A} \in \mathbb{R}^{n_c \times n_g}$, $\mathbf{b} \in \mathbb{R}^{n_c}$, $\mathbf{h} \in \mathbb{R}^n$, $\mathbf{E}_r = \frac{1}{2} \text{diag}(\boldsymbol{\xi}^U - \boldsymbol{\xi}^L)$, $\boldsymbol{\xi}_m = \frac{1}{2}(\boldsymbol{\xi}^U + \boldsymbol{\xi}^L)$. Consider $(\boldsymbol{\xi}^L, \boldsymbol{\xi}^U)$ computed by solving the LP in Proposition 2, in which $(\tilde{\boldsymbol{\xi}}^L, \tilde{\boldsymbol{\xi}}^U)$ is obtained by Algorithm 1 in the Appendix in Scott et al. (2016). In the standard form, the LP in Proposition 2 has $N_d = 3n_g$ decision variables and $N_c = n + 4n_g$ constraints. Then,

$$\begin{aligned}
O(\text{Proposition 2}) &= O(\mathbf{G}\mathbf{E}_r) + O(\mathbf{A}\mathbf{E}_r) + O(-\mathbf{G}) + O(\mathbf{b} - \mathbf{A}\boldsymbol{\xi}_m) + O(\mathbf{c} - \mathbf{h}) \\
&\quad + O((1/2)\text{diag}(\boldsymbol{\xi}^U - \boldsymbol{\xi}^L)) + O((1/2)(\boldsymbol{\xi}^U + \boldsymbol{\xi}^L)) + O(N_d N_c^3) \\
&\quad + O(\text{Algorithm 1}) \\
&= O(nn_g^2) + O(n_c n_g^2) + O(nn_g) + O(n_c) + O(n_c n_g) + O(n) + O(n_g) + O(n_g) \\
&\quad + O(N_d N_c^3) + O(\text{Algorithm 1}) \\
&= O(nn_g^2) + O(n_c n_g^2) + O(N_d N_c^3) + O(\text{Algorithm 1}) \\
&= O(nn_g^2) + O(n_c n_g^2) + O((3n_g)(n + 4n_g)^3) + O(\text{Algorithm 1}) \\
&= O(nn_g^2) + O(n_c n_g^2) + O(n_g(n + n_g)^3) + O(\text{Algorithm 1}) \\
&= O(n_g(n + n_g)^3 + n_g^2 n_c) + O(n_g^2 n_c) \\
&= O(n_g(n + n_g)^3 + n_g^2 n_c).
\end{aligned}$$

A.2.2 CZMV

A.2.2.1 Prediction step (Theorem 2)

Let $\boldsymbol{\mu} : \mathbb{R}^n \times \mathbb{R}^{n_w} \rightarrow \mathbb{R}^n$ be continuously differentiable, and let $\nabla_x \boldsymbol{\mu}$ denote the gradient of $\boldsymbol{\mu}$ with respect to its first argument. Let $\hat{X}_{k-1} = \{\mathbf{G}_x, \mathbf{c}_x, \mathbf{A}_x, \mathbf{b}_x\} \subset \mathbb{R}^n$ and $W = \{\mathbf{G}_w, \mathbf{c}_w, \mathbf{A}_w, \mathbf{b}_w\} \subset \mathbb{R}^{n_w}$, with $\mathbf{c}_x \in \mathbb{R}^n$, $\mathbf{G}_x \in \mathbb{R}^{n \times n_g}$, $\mathbf{A}_x \in \mathbb{R}^{n_c \times n_g}$, $\mathbf{b}_x \in \mathbb{R}^{n_c}$, $\mathbf{c}_w \in \mathbb{R}^{n_w}$, $\mathbf{G}_w \in \mathbb{R}^{n_w \times n_{g_w}}$, $\mathbf{A}_w \in \mathbb{R}^{n_{c_w} \times n_{g_w}}$, and $\mathbf{b}_w \in \mathbb{R}^{n_{c_w}}$, and choose any $\mathbf{h} \in \hat{X}_{k-1}$. Let Z be a constrained zonotope such that $\boldsymbol{\mu}(\mathbf{h}, W) \subseteq Z$, and let $\mathbf{J} \in \mathbb{I}\mathbb{R}^{n \times n}$ be an interval matrix satisfying $\nabla_x^T \boldsymbol{\mu}(\hat{X}_{k-1}, W) \subseteq \mathbf{J}$. Theorem 2 ensures that

$$\boldsymbol{\mu}(\hat{X}_{k-1}, W) \subseteq \bar{X}_k = Z \oplus \triangleleft(\mathbf{J}, \hat{X}_{k-1} - \mathbf{h}),$$

where \bar{X}_k has $n_g + n_{g_w} + 2n$ generators and $n_c + n_{c_w}$ constraints if Z is also computed by the mean value extension. Therefore,

$$\begin{aligned}
O(\text{CZMV}_{\text{prediction}}) &= O(Z \oplus \triangleleft(\mathbf{J}, \hat{X}_{k-1} - \mathbf{h})) + O(\triangleleft(\mathbf{J}, \hat{X}_{k-1} - \mathbf{h})) + O(Z) \\
&\quad + O(\text{interval hull of } \hat{X}_{k-1}, W) \\
&= O(n) + O(n^2 n_g + n_c(n_g + n_c)^3 + nn_g^2 n_c) + O(nn_w n_{g_w} + n_{c_w}(n_{g_w} + n_{c_w})^3 \\
&\quad + n_w n_{g_w}^2 n_{c_w}) + O(nn_g(n_g + n_c)^3 + n_w n_{g_w}(n_{g_w} + n_{c_w})^3) \\
&= O(n^2 n_g + nn_w n_{g_w} + (nn_g + n_c)(n_g + n_c)^3 + (n_w n_{g_w} + n_{c_w})(n_{g_w} + n_{c_w})^3).
\end{aligned}$$

A.2.2.2 Update step

Let $\bar{X}_k = \{\mathbf{G}_x, \mathbf{c}_x, \mathbf{A}_x, \mathbf{b}_x\} \subset \mathbb{R}^n$, $V = \{\mathbf{G}_v, \mathbf{c}_v, \mathbf{A}_v, \mathbf{b}_v\} \subset \mathbb{R}^{n_v}$, $\mathbf{u}_k \in \mathbb{R}^{n_u}$, and $\mathbf{y}_k \in \mathbb{R}^{n_y}$ with $\mathbf{c}_x \in \mathbb{R}^n$, $\mathbf{G}_x \in \mathbb{R}^{n \times \bar{n}_g}$, $\mathbf{A}_x \in \mathbb{R}^{\bar{n}_c \times \bar{n}_g}$, $\mathbf{b}_x \in \mathbb{R}^{\bar{n}_c}$, $\mathbf{c}_v \in \mathbb{R}^{n_v}$, $\mathbf{G}_v \in \mathbb{R}^{n_v \times n_{g_v}}$, $\mathbf{A}_v \in \mathbb{R}^{n_{c_v} \times n_{g_v}}$, and $\mathbf{b}_v \in \mathbb{R}^{n_{c_v}}$. The update step is performed according to

$$\hat{X}_k = \bar{X}_k \cap_{\mathbf{C}} ((\mathbf{y}_k - \mathbf{D}_u \mathbf{u}_k) \oplus (-\mathbf{D}_v V)),$$

where $\mathbf{C} \in \mathbb{R}^{n_y \times n}$, $\mathbf{D}_u \in \mathbb{R}^{n_y \times n_u}$, and $\mathbf{D}_v \in \mathbb{R}^{n_y \times n_v}$. As a result, from the prediction step of CZMV, \bar{X}_k has $\bar{n}_g = n_g + n_{g_w} + 2n$ generators and $\bar{n}_c = n_c + n_{c_w}$ constraints. Therefore,

$$\begin{aligned} O(\text{CZMV}_{\text{update}}) &= O(\bar{X}_k \cap_{\mathbf{C}} ((\mathbf{y}_k - \mathbf{D}_u \mathbf{u}_k) \oplus (-\mathbf{D}_v V))) \\ &= O(n(n_g + n_{g_w} + 2n)n_y + n_y n_{g_v}) + O(n_y) + O(n_y + n_y n_u) + O(n_y n_v n_{g_v}) \\ &= O(n_y n(m_g + n) + n_y n_u + n_y n_v n_{g_v}), \end{aligned}$$

where $m_g = n_g + n_{g_w}$.

A.2.2.3 Order reduction

Let n_g and n_c denote the desired number of generators and constraints of \hat{X}_k after order reduction, i.e., \hat{X}_k must have the same complexity as \hat{X}_{k-1} . Order reduction is performed by eliminating k_c constraints and then k_g generators from \hat{X}_k . For simplicity, define $m_g = n_g + n_{g_w}$, $m_c = n_c + n_{c_w}$, $\delta_n = n - n_y$, $\delta_w = n_{g_w} - n_{c_w}$, and $\delta_v = n_{g_v} - n_{c_v}$. The prediction and update steps of CZMV lead to \hat{X}_k with $n_g + n_{g_w} + 2n + n_{g_v}$ generators and $n_c + n_{c_w} + n_{c_v} + n_y$ constraints. Therefore, $k_c = n_{c_w} + n_{c_v} + n_y$, and $k_g = n_{g_w} + 2n + n_{g_v} - k_c = n_{g_w} + 2n + n_{g_v} - n_{c_w} - n_{c_v} - n_y$. Consequently,

$$\begin{aligned} O(\text{eliminate } k_c \text{ constraints from } \hat{X}_k) &= O(k_c(m_g + 2n + n_{g_v} + m_c + n_{c_v} + n_y)^3 \\ &\quad + k_c n(m_g + 2n + n_{g_v})^2) \\ &= O((n_{c_w} + n_{c_v} + n_y)(m_g + m_c + n_{g_v} + n_{c_v} + n + n_y)^3 \\ &\quad + (n_{c_w} + n_{c_v} + n_y)n(m_g + n_{g_v} + n)^2) \\ &= O((n_{c_w} + n_{c_v} + n_y)(m_g + m_c + n_{g_v} + n_{c_v} + n + n_y)^3). \end{aligned}$$

Note that by eliminating k_c constraints from \hat{X}_k , the same number of generators (i.e., k_c) is also removed (Scott et al., 2016). Hence,

$$\begin{aligned} O(\text{elim. } k_g \text{ gen. from } \hat{X}_k \text{ after const. elim.}) &= O((n + n_c)^2(m_g + 2n + n_{g_v} - k_c) \\ &\quad + k_g(m_g + 2n + n_{g_v} - k_c)(n + n_c)) \\ &= O((n + n_c)^2(\delta_n + \delta_w + \delta_v + n_g) \\ &\quad + (\delta_n + \delta_w + \delta_v)(\delta_n + \delta_w + \delta_v + n_g)(n + n_c)). \end{aligned}$$

Finally,

$$\begin{aligned}
O(\text{CZMV}_{\text{reduction}}) &= O(\text{eliminate } k_c \text{ constraints from } \hat{X}_k) \\
&\quad + O(\text{elim. } k_g \text{ gen. from } \hat{X}_k \text{ after const. elim.}) \\
&= O((n_{cw} + n_{cv} + n_y)(m_g + m_c + n_{gv} + n_{cv} + n + n_y)^3 \\
&\quad + (n + n_c)^2(n_g + \delta_n + \delta_w + \delta_v) + (n + n_c)(\delta_n + \delta_w + \delta_v)(n_g + \delta_n + \delta_w + \delta_v)).
\end{aligned}$$

A.2.3 CZFO

A.2.3.1 Prediction step

Let $\eta : \mathbb{R}^m \rightarrow \mathbb{R}^n$ be of class \mathcal{C}^2 , and let $\mathbf{z} = (\mathbf{x}_{k-1}, \mathbf{w}_{k-1}) \in \mathbb{R}^m$ denote its argument, where $m = n + n_w$. Let $Z = \hat{X}_{k-1} \times W = \{\mathbf{G}, \mathbf{c}, \mathbf{A}, \mathbf{b}\} \subset \mathbb{R}^m$ be a constrained zonotope with $m_g = n_g + n_{gw}$ generators and $m_c = n_c + n_{cw}$ constraints. For each $q = 1, 2, \dots, n$, let $\mathbf{Q}^{[q]} \in \mathbb{R}^{m \times m}$ and $\tilde{\mathbf{Q}}^{[q]} \in \mathbb{R}^{m_g \times m_g}$ be interval matrices satisfying $\mathbf{Q}^{[q]} \supseteq \mathbf{H}\eta_q(Z)$ and $\tilde{\mathbf{Q}}^{[q]} \supseteq \mathbf{G}^T \mathbf{Q}^{[q]} \mathbf{G}$. Moreover, define $\tilde{\mathbf{c}}, \tilde{\mathbf{G}}, \tilde{\mathbf{G}}_d, \tilde{\mathbf{A}}, \tilde{\mathbf{b}}$, and \mathbf{L} , as in Theorem 3. Choosing any $\mathbf{h} \in Z$, Theorem 3 ensures that

$$\eta(Z) \subseteq \bar{X}_k = \eta(\mathbf{h}) \oplus \nabla^T \eta(\mathbf{h})(Z - \mathbf{h}) \oplus R,$$

where $R = \tilde{\mathbf{c}} \oplus [\tilde{\mathbf{G}} \ \tilde{\mathbf{G}}_d] B_\infty(\tilde{\mathbf{A}}, \tilde{\mathbf{b}}) \oplus \langle \mathbf{L}, (\mathbf{c} - \mathbf{h}) \oplus 2\mathbf{G}B_\infty(\mathbf{A}, \mathbf{b}) \rangle$ and \bar{X}_k has $(1/2)m_g^2 + (5/2)m_g + 2n$ generators and $(1/2)m_c^2 + (5/2)m_c$ constraints. Therefore,

$$\begin{aligned}
O(\text{CZFO}_{\text{prediction}}) &= O(\eta(\mathbf{h}) \oplus \nabla^T \eta(\mathbf{h})(Z - \mathbf{h}) \oplus R) + O(R) \\
&= O(n) + O(nmm_g) + O(R) = O(nmm_g) + O(R).
\end{aligned}$$

But

$$\begin{aligned}
O(R) &= O(\tilde{\mathbf{c}} \oplus [\tilde{\mathbf{G}} \ \tilde{\mathbf{G}}_d] B_\infty(\tilde{\mathbf{A}}, \tilde{\mathbf{b}}) \oplus \langle \mathbf{L}, (\mathbf{c} - \mathbf{h}) \oplus 2\mathbf{G}B_\infty(\mathbf{A}, \mathbf{b}) \rangle) \\
&= O(n) + O(\tilde{\mathbf{c}}) + O(\tilde{\mathbf{G}}) + O(\tilde{\mathbf{G}}_d) + O(\tilde{\mathbf{A}}) + O(\tilde{\mathbf{b}}) + O(\mathbf{L}) \\
&\quad + O(\langle \mathbf{L}, (\mathbf{c} - \mathbf{h}) \oplus 2\mathbf{G}B_\infty(\mathbf{A}, \mathbf{b}) \rangle) + nO(\tilde{\mathbf{Q}}^{[q]}) + O(\text{interval hull of } Z) \\
&= O(n) + O(nm_g) + O(nm_g^2) + O(nm_g^2) + O(m_g m_c^2 + m_g^2 m_c^2) + O(m_g m_c^2) + O(nm^2) \\
&\quad + O(n + nmm_g + m_c(m_g + m_c)^3 + mm_g^2 m_c) + nO(m^2 m_g + mm_g^2) + O(mm_g(m_g + m_c)^3) \\
&= O(n(m^2 m_g + mm_g^2) + (mm_g + m_c)(m_g + m_c)^3).
\end{aligned}$$

Therefore,

$$O(\text{CZFO}_{\text{prediction}}) = O(nmm_g) + O(R) = O(n(m^2 m_g + mm_g^2) + (mm_g + m_c)(m_g + m_c)^3).$$

A.2.3.2 Update step

Let $\bar{X}_k = \{\mathbf{G}_x, \mathbf{c}_x, \mathbf{A}_x, \mathbf{b}_x\} \subset \mathbb{R}^n$, $V = \{\mathbf{G}_v, \mathbf{c}_v, \mathbf{A}_v, \mathbf{b}_v\} \subset \mathbb{R}^{n_v}$, $\mathbf{u}_k \in \mathbb{R}^{n_u}$, and $\mathbf{y}_k \in \mathbb{R}^{n_y}$ with $\mathbf{c}_x \in \mathbb{R}^n$, $\mathbf{G}_x \in \mathbb{R}^{n \times \bar{n}_g}$, $\mathbf{A}_x \in \mathbb{R}^{\bar{n}_c \times \bar{n}_g}$, $\mathbf{b}_x \in \mathbb{R}^{\bar{n}_c}$, $\mathbf{c}_v \in \mathbb{R}^{n_v}$, $\mathbf{G}_v \in \mathbb{R}^{n_v \times n_{g_v}}$, $\mathbf{A}_v \in \mathbb{R}^{n_{c_v} \times n_{g_v}}$, and $\mathbf{b}_v \in \mathbb{R}^{n_{c_v}}$. The update step is performed according to

$$\hat{X}_k = \bar{X}_k \cap_{\mathbf{C}} ((\mathbf{y}_k - \mathbf{D}_u \mathbf{u}_k) \oplus (-\mathbf{D}_v V)),$$

where $\mathbf{C} \in \mathbb{R}^{n_y \times n}$, $\mathbf{D}_u \in \mathbb{R}^{n_y \times n_u}$, and $\mathbf{D}_v \in \mathbb{R}^{n_y \times n_v}$. As a result, from the prediction step of CZFO, \bar{X}_k has $\bar{n}_g = (1/2)m_g^2 + (5/2)m_g + 2n$ generators and $\bar{n}_c = (1/2)m_c^2 + (5/2)m_c$ constraints, where $m_g = n_g + n_{g_w}$, and $m_c = n_c + n_{c_w}$. Therefore,

$$\begin{aligned} O(\text{CZFO}_{\text{update}}) &= O(\bar{X}_k \cap_{\mathbf{C}} ((\mathbf{y}_k - \mathbf{D}_u \mathbf{u}_k) \oplus (-\mathbf{D}_v V))) \\ &= O(n((1/2)m_g^2 + (5/2)m_g + 2n)n_y + n_y n_{g_w}) + O(n_y) + O(n_y + n_y n_u) + O(n_y n_v n_{g_v}) \\ &= O(n_y n(m_g^2 + n) + n_y n_u + n_y n_v n_{g_v}). \end{aligned}$$

A.2.3.3 Order reduction

Let n_g and n_c denote the desired number of generators and constraints of \hat{X}_k after order reduction, i.e., \hat{X}_k must have the same complexity as \hat{X}_{k-1} . Order reduction is performed by eliminating k_c constraints and then k_g generators from \hat{X}_k . For simplicity, define $m_g = n_g + n_{g_w}$, $m_c = n_c + n_{c_w}$, $\delta_n = n - n_y$, $\delta_v = n_{g_v} - n_{c_v}$, and $\tilde{\delta} = m_g^2 - m_c^2$. The prediction and update steps of CZFO lead to \hat{X}_k with $(1/2)m_g^2 + (5/2)m_g + 2n + n_{g_w}$ generators and $(1/2)m_c^2 + (5/2)m_c + n_{c_v} + n_y$ constraints. Therefore, $k_c = (1/2)m_c^2 + (5/2)m_c + n_{c_v} + n_y - n_c$ and $k_g = (1/2)m_g^2 + (5/2)m_g + 2n + n_{g_w} - n_g - k_c = (1/2)m_g^2 + (5/2)m_g + 2n + n_{g_w} - n_g - (1/2)m_c^2 - (5/2)m_c - n_{c_v} - n_y + n_c$. Consequently,

$$\begin{aligned} O(\text{eliminate } k_c \text{ constraints from } \hat{X}_k) &= O(k_c(m_g^2 + m_g + n + n_{g_w} + m_c^2 + m_c + n_{c_v} + n_y)^3 \\ &\quad + k_c n(m_g^2 + m_g + n)^2) \\ &= O((m_c^2 + n_{c_v} + n_y)(m_g^2 + n + n_{g_w} + m_c^2 + n_{c_v} + n_y)^3 \\ &\quad + (m_c^2 + n_{c_v} + n_y)n(m_g^2 + n)^2) \\ &= O((m_c^2 + n_{c_v} + n_y)(m_g^2 + m_c^2 + n_{g_v} + n_{c_v} + n + n_y)^3). \end{aligned}$$

Once again, by eliminating k_c constraints from \hat{X}_k , the same number of generators (i.e., k_c) is also removed (Scott et al., 2016). Hence,

$$\begin{aligned} O(\text{elim. } k_g \text{ gen. from } \hat{X}_k \text{ after const. elim.}) &= O((n + n_c)^2(m_g^2 + n + n_{g_w} - m_c^2 - n_{c_v} - n_y) \\ &\quad + k_g(m_g^2 + n + n_{g_w} - m_c^2 - n_{c_v} - n_y)(n + n_c)) \\ &= O((n + n_c)^2(\tilde{\delta} + \delta_n + \delta_v) \\ &\quad + (\tilde{\delta} + \delta_n + \delta_v)(\tilde{\delta} + \delta_n + \delta_v)(n + n_c)) \end{aligned}$$

$$= O((n + n_c)^2(\tilde{\delta} + \delta_n + \delta_v) + (\tilde{\delta} + \delta_n + \delta_v)^2(n + n_c)).$$

Finally,

$$\begin{aligned} O(\text{CZFO}_{\text{reduction}}) &= O(\text{eliminate } k_c \text{ constraints from } \hat{X}_k) \\ &\quad + O(\text{elim. } k_g \text{ gen. from } \hat{X}_k \text{ after const. elim.}) \\ &= O((m_c^2 + n_{c_v} + n_y)(m_g^2 + m_c^2 + n_{g_v} + n_{c_v} + n + n_y)^3 \\ &\quad + (n + n_c)^2(\tilde{\delta} + \delta_n + \delta_v) + (\tilde{\delta} + \delta_n + \delta_v)^2(n + n_c)). \end{aligned}$$