



UNIVERSIDADE FEDERAL DE MINAS GERAIS

Programa de Pós-Graduação em Engenharia Elétrica

Mechatronics, Control and Robotics (MACRO) research group

Laboratory of Modeling, Analysis and Non Linear System Control (MACSIN) research
group

Alesi Augusto de Paula

STATE ESTIMATION OF DYNAMICAL SYSTEMS WITH MIXED REPRESENTATION OF UNCERTAINTIES

PH.D. Thesis

Belo Horizonte
2022

Alesi Augusto de Paula

**STATE ESTIMATION OF DYNAMICAL SYSTEMS
WITH MIXED REPRESENTATION OF
UNCERTAINTIES**

Thesis submitted to the Graduate Program in Electrical Engineering of the Universidade Federal de Minas Gerais in partial fulfilment of the requirements for the degree of Doctor in Electrical Engineering.

Advisors: Prof. Bruno Otávio Soares Teixeira, Dr.
Prof. Guilherme Vianna Raffo, Dr.

Belo Horizonte
2022

P324s

Paula, Alesi Augusto de.

State estimation of dynamical systems with mixed representation of uncertainties [recurso eletrônico] / Alesi Augusto de Paula. - 2022.
1 recurso online (296 f. : il., color.) : pdf.

Orientadores: Bruno Otávio Soares Teixeira, Guilherme Vianna Raffo.

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

Apêndices: f. 203-296.

Bibliografia: f. 192-202.

Exigências do sistema: Adobe Acrobat Reader.

1. Engenharia elétrica - Teses. 2. Incerteza - Teses. 3. Análise estocástica - Teses. 4. Kalman, Filtragem de - Teses. I. Teixeira, Bruno Otávio Soares. II. Raffo, Guilherme Vianna. 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

FOLHA DE APROVAÇÃO

"STATE ESTIMATION OF DYNAMICAL SYSTEMS WITH MIXED REPRESENTATION OF UNCERTAINTIES"

ALESIO AUGUSTO DE PAULA

Tese de Doutorado submetida à Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em Engenharia Elétrica da Escola de Engenharia da Universidade Federal de Minas Gerais, como requisito para obtenção do grau de Doutor em Engenharia Elétrica. Aprovada em 06 de dezembro de 2022. Por:

Prof. Dr. Bruno Otávio Soares Teixeira
Departamento de Engenharia Eletrônica (UFMG) - Orientador

Prof. Dr. Guilherme Vianna Raffo
Departamento de Engenharia Eletrônica (UFMG) - Co-orientador

Prof. Dr. Víctor Costa da Silva Campos
Departamento de Engenharia Eletrônica (UFMG)

Prof. Dr. Tito Luís Maia Santos
Departamento de Engenharia Elétrica (UFBA)

Prof. Dr. Davide Martino Raimondo
Department of Electrical, Computer and Biomedical Engineering (Università degli Studi di Pavia)

Dr. Brenner Santana Rego
Departamento de Engenharia Elétrica (USP)



Documento assinado eletronicamente por **Bruno Otavio Soares Teixeira, Pró-reitor(a)**, em 06/12/2022, às 13:36, conforme horário oficial de Brasília, com fundamento no art. 5º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



Documento assinado eletronicamente por **Guilherme Vianna Raffo, Professor do Magistério Superior**, em 06/12/2022, às 15:27, conforme horário oficial de Brasília, com fundamento no art. 5º do [Decreto nº 10.543, de 13 de novembro de 2020](#).

Documento assinado eletronicamente por **Brenner Santana Rêgo, Usuário Externo**, em 06/12/2022, às 17:05, conforme horário oficial de Brasília, com fundamento no art. 5º do [Decreto nº 10.543, de](#)



[13 de novembro de 2020.](#)



Documento assinado eletronicamente por **Davide Martino Raimondo, Usuário Externo**, em 07/12/2022, às 12:41, conforme horário oficial de Brasília, com fundamento no art. 5º do [Decreto nº 10.543, de 13 de novembro de 2020.](#)



Documento assinado eletronicamente por **Tito Luís Maia Santos, Usuário Externo**, em 13/12/2022, às 10:51, conforme horário oficial de Brasília, com fundamento no art. 5º do [Decreto nº 10.543, de 13 de novembro de 2020.](#)



Documento assinado eletronicamente por **Victor Costa da Silva Campos, Professor do Magistério Superior**, em 19/12/2022, às 17:09, conforme horário oficial de Brasília, com fundamento no art. 5º do [Decreto nº 10.543, de 13 de novembro de 2020.](#)



A autenticidade deste documento pode ser conferida no site https://sei.ufmg.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **1870080** e o código CRC **94757773**.

Acknowledgements

First of all, I thank my true friend, brother, and Lord Jesus Christ for everything. I have complete certainty in your proposal, care, and salvation! My life and conquests are yours; you can use them all to complete your purposes.

Second, I thank my parents and brothers for the encouragement, comprehension, and support. I have no doubt about your devotion.

Third, I thank my advisors Bruno Otávio Soares Teixeira and Guilherme Vianna Raffo for the patience, ideas, advices, companionship, professionalism, help, and much more. You have no idea how grateful I am. Moreover, I thank Profs. Víctor Costa da Silva Campos and Davide Martino Raimondo for the partnership. In addition, I thank the members of the examining board for all pertinent comments.

Fourth, I thank the Brazilian agencies CAPES, FAPEMIG and CNPq for the financial support. Maybe, the lack of this resource would have canceled many opportunities.

Fifth, I thank Brenner Santana Rego, Richard Alfonso Andrade Alfaro, and Cristiano Martins Monteiro for discussions and help. I wish you success in your all projects.

Sixth, I thank both PPGE and PPGCC for providing a great team of professors and employees. Specially, I thank the Profs. Geraldo Robson Mateus and Lucas de Souza Batista for the valuable discussions.

Finally, I thank all people who direct or indirectly influenced my research. For you, my sincere acknowledgments.

Resumo

Este trabalho propõe estimadores de estados para sistemas dinâmicos lineares e não lineares com representação mista de incertezas. Para alcançar isto, as abordagens *estocásticas* e *baseadas em conjuntos* são combinadas. A primeira abordagem visa determinar propriedades estatísticas que descrevam a natureza estocástica do vetor de estados. Por exemplo, os algoritmos de filtragem de Kalman (KF) retornam as estimativas de média e covariância para definir a função densidade de probabilidade *a posteriori*. A segunda abordagem visa determinar conjuntos que contenham a parcela desconhecida mas limitada do vetor de estados. Por exemplo, algoritmos baseados em zonotos estimam o centro e a matriz geradora para eficientemente caracterizar politopos centralmente simétricos. Sendo assim, a abordagem baseada em conjunto é motivada para trabalhar com análise de pior caso e não requer conhecimento estatístico prévio sobre incertezas. Tais algoritmos são comumente chamados de garantidos porque eles computam conjuntos que incluem todas as possíveis trajetórias de estado.

O tópico de incerteza mista tem sido inicialmente motivado pela presença de ruído aleatório e erro sistemático em equações de processo e medição, respectivamente. Desde então, combinações entre KF e filtragem baseada em conjuntos têm sido exploradas. Atualmente, outras fontes de incerteza têm sido incorporadas pela classe de incerteza mista, tais como erros de aproximação e incertezas epistêmicas, e novas classes de conjuntos têm sido empregadas tais como zonotos restritos.

O foco desta tese é propor algoritmos mistos de incertezas. O ponto de partida é estender os algoritmos mistos de incertezas baseados em zonotos existentes como segue: preditores são estendidos para filtros para melhorar precisão e acurácia; zonotos são substituídos por zonotos restritos para melhorar a precisão com respeito a politopos assimétricos; uma abordagem de programação da diferença de funções convexas é empregada para tratar o erro de linearização analítica; uma análise de solução é introduzida sobre a fronteira de Pareto ótima para justamente combinar diferentes custos; e restrições de igualdade e desigualdade são reforçadas sobre o vetor de estados. Depois, o objetivo é combinar algoritmos estocásticos com algoritmos baseados em conjuntos para gerar novos algoritmos mistos de incertezas. Contribuições adicionais são discutidas ao longo do trabalho por meio de vários exemplos numéricos.

Palavras-chave: Estimação de estados. Incerteza mista. Filtragem estocástica. Filtragem baseada em conjuntos. Restrições de igualdade e desigualdade.

Abstract

This work proposes state estimators for linear and nonlinear dynamical systems with mixed-uncertainty representation. To achieve that, the *stochastic* and *set-based* approaches are combined. The former aims at determining statistical properties that describe the stochastic nature of the state vector. For instance, the Kalman filtering (KF) algorithms return the mean and covariance estimates to define the *a posteriori* probability density function. The latter aims at determining sets that contain the unknown-but-bounded parcel of the state vector. For instance, zonotope-based algorithms estimate the center and generator matrix to efficiently characterize centrally symmetric polytopes. Therefore, the set-based approach is motivated to work on worst-case analysis and does not require any prior statistical knowledge about uncertainties. Such algorithms are commonly called guaranteed because they compute sets that include all possible state trajectories.

The topic of mixed-uncertainty filtering has been initially motivated by the presence of random noise and systematic error in process and measurement equations, respectively. Since then, combinations among KF and set-based filtering have been exploited. Currently, other sources of uncertainty have been incorporated by the class of mixed uncertainty, such as approximation errors and epistemic uncertainties, and new classes of set have been employed such as constrained zonotopes.

The focus of this thesis is to propose mixed-uncertainty algorithms. The starting point is to extend the existing zonotope-based mixed algorithms as follows: predictors are extended to filters to enhance precision and accuracy; zonotopes are replaced by constrained zonotopes to improve the precision with respect to asymmetric polytopes; an approach of difference of convex functions programming is employed to treat the analytical linearization error; a solution analysis is introduced on the Pareto-optimal front to fairly combine different costs; and equality and inequality constraints are enforced on the state vector. Next, the goal is to combine stochastic algorithms with set-based algorithms to build novel mixed-uncertainty algorithms. Further contributions are discussed throughout the work by using various numerical examples.

Keywords: State estimation. Mixed uncertainty. Stochastic filtering. Set-based filtering. Equality and inequality constraints.

List of Figures

1.1	Illustrative sets in 2D.	31
1.2	Illustration of a Pareto curve estimated in 2D.	39
2.1	Illustration of the GRV $X \sim \mathcal{N}(0_{2 \times 1}, I_2)$ in 2D.	47
2.2	Illustrating convex polytopes in 2D.	51
2.3	Illustrative example of the wrapping effect for functions whose input is multivariate and output is monovariate.	55
2.4	Illustrative example of the wrapping and dependency effects in 2D.	56
2.5	Illustrations of the 2D sets described in Example 2.2.6.	60
2.6	Exact intersections between the sets presented in Example 2.2.7.	61
2.7	2D illustration of candidate and optimal parallelotopes.	63
2.8	Diagram to illustrate the mixed-uncertainty vector in 2D.	67
2.9	Illustration of the application of Definitions 2.4.3 and 2.4.4 to a mixed-uncertainty vector, whose results are a confidence CZ $\mathcal{X}^a = c^x \oplus \mathcal{X} \oplus \mathcal{P}$ and a confidence box $\mathcal{I}^a = c^x \oplus \square \mathcal{X} \oplus \square \mathcal{E}$	70
3.1	Results of state estimation for the example of Subsection 3.1.5.1.	91
3.2	Additional results for the example of Subsection 3.1.5.1.	92
3.3	Results of state estimation for the example of Subsection 3.1.5.2.	93
3.4	Results of state estimation for the example of Subsection 3.1.7.	100
3.5	Illustrative diagram of the interval zonotopic filter.	103
3.6	Example of the intersection between a constraint box and a zonotope using strips.	104
3.7	Illustrating differences among volume-minimization methods.	105
3.8	Illustrative diagram of a quadrotor.	108
3.9	State estimation for the quadrotor UAV using the UKF and TUKF algorithms.	112
3.10	State estimation for the quadrotor UAV using the zonotopic filter and interval zonotopic filter algorithms with the segment minimization.	113

3.11	State estimation with ZDC, CZDC, CZFO, and CZMV for a two-state example whose process equations are nonlinear.	119
3.12	State estimation with CZDC, CZFO, and CZMV for the attitude example.	122
4.1	Illustrative diagram of an iteration of the zonotopic and Gaussian Kalman filter.	129
4.2	State estimation for a two-state linear system using the ZGKP and ZGKF.	129
4.3	Three procedures to outer-approximate CZs by zonotopes.	133
4.4	Illustrative diagram of LTVMF.	141
4.5	Illustrative diagram of explicit LTVMF.	144
4.6	State estimation for the three-state LTV system with LTVMF.	146
4.7	Illustrative diagram of the LTIMF algorithm.	153
4.8	Time results of state estimation for the compartmental system regarding one separate simulation.	157
5.1	State estimation for the Van der Pol oscillator using EZGKP and EZGKF.	166
5.2	State estimation for an LPV system using EZGKP, EZGKF, and EZGKF-M.	167
5.3	Illustration of an iteration of the mixed-uncertainty filter (MF) with the quadratic or explicit version.	179
5.4	State estimation for the Van der Pol oscillator with FOMF and DCMF.	182
5.5	State estimation for the batch reactor with FOMF and DCMF.	183
5.6	State estimation for the pendulum with DCMF and MAUKF.	185
A.1	One-step nonlinear transformation of GRVs.	211
A.2	An application of the PDF truncation in 2D.	214
A.3	An application of the ICUT in 2D.	216
B.1	Illustrative example of the order reduction over zonotopes.	220
B.2	One-step nonlinear transformation of zonotopes.	236
D.1	Diagram to illustrate the linearization-discretization and discretization-approximation.	247
D.2	State estimation of the system (3.128)-(3.129) using the ZMV.	252
D.3	State estimation of the two-state batch reactor using the ZMV.	253
D.4	State estimation for the Lorenz system using the ZMV.	255
D.5	State estimation for the system (D.31)-(D.32) using the ZMV.	259

List of Tables

3.1	Overview of the stochastic and set-based algorithms proposed in this chapter, compared to analogous existing methods.	74
3.2	State estimation for a five-state system using linear stochastic algorithms.	85
3.3	Complexity $O(\cdot)$ of DRZF.	90
3.4	Results of $A^\square$ and $r^\square$ for the system of Subsection 3.1.5.1.	91
3.5	Results of $A^\square$ and $r^\square$ for the system of Subsection 3.1.5.2.	92
3.6	State estimation for a five-state system using linear zonotopic algorithms.	100
3.7	Complexity order of the forecast, data assimilation, admissibility, consistency, and reduction steps from CZDC using Proposition 2.3.2.	117
3.8	Results of T^{CPU} and $A^\square$ for the first example (Subsection 3.2.4.1).	119
3.9	Results of T^{CPU} and $A^\square$ for the second example (Subsection 3.2.4.2). The percentage reduction of $A^\square$ for CZDC and CZFO in comparison to CZMV is shown between parenthesis.	122
4.1	Overview of the linear mixed-uncertainty algorithms here proposed.	125
4.2	State estimation for a two-state system using ZGKP and ZGKF.	130
4.3	Results of T^{CPU} , RMSE, and $r^\square$ for the compartmental system.	156
5.1	Overview of the nonlinear mixed-uncertainty algorithms here proposed.	159
5.2	Complexity order $O(\cdot)$ of each step from EZGKF.	164
5.3	Results of T^{CPU} , RMSE, and $r^\square$ for the Van der Pol oscillator using EZGKP and EZGKF.	165
5.4	Results of T^{CPU} , RMSE, and $r^\square$ for an LPV system using EZGKP, EZGKF, and EZGKF-M.	167
5.5	Complexity order $O(\cdot)$, and number of constraints $\tilde{n}_h$ and generators $\tilde{n}_g$ for CZs. The results of the last column are related to Proposition 2.3.2.	168
5.6	Results of T^{CPU} , RMSE, and $r^\square$ for the Van der Pol oscillator using FOMF and DCMF.	181
5.7	Results of T^{CPU} , RMSE, and $r^\square$ for the batch reactor.	184
5.8	Results of T^{CPU} , RMSE, and $r^\square$ for the pendulum using DCMF and MAUKF.	185

6.1	Summary of all state estimators proposed in this thesis.	188
-----	--	-----

List of Algorithms

2.1	LBE (Hertz, 1992)	66
3.1	DGF	79
3.2	DH ₂ GF	83
3.3	DRZF	89
3.4	ZKF	96
3.5	NH _∞ ZF	98
3.6	IZMV	102
3.7	VM3	107
3.8	CZDC	116
4.1	ZGKF	128
4.2	LTVMF	140
4.3	LTVMF-E	144
4.4	LTIMF	152
5.1	EZGKF	163
5.2	MF	177
5.3	MF-E	179
A.1	KP (Kalman, 1960)	205
A.2	KF (Kalman, 1960)	206
A.3	UT (Menegaz et al., 2015)	210
A.4	UKF (Julier & Uhlmann, 1997)	211
A.5	PT (Shimada et al., 1998)	213
A.6	ICUT (Vachhani et al., 2006)	215
A.7	TUKF (Teixeira et al., 2008b)	216
B.1	ZR (Combastel, 2003)	219
B.2	ZKP (Combastel, 2015)	221
B.3	DTZF (Tang et al., 2020)	224
B.4	H _∞ ZF (Tang et al., 2019a)	228

B.5	CZLF (Scott et al., 2016)	229
B.6	ZEKP (Wang & Puig, 2016)	230
B.7	SM (Alamo et al., 2005)	231
B.8	VM1 (Alamo et al., 2005)	231
B.9	VM2 (Bravo et al., 2006)	232
B.10	ZT1	233
B.11	ZT2 (Alamo et al., 2005)	234
B.12	ZT3 (Alamo et al., 2005)	235
B.13	ZMV (Alamo et al., 2005)	237
C.1	ZGKP (Combastel, 2015)	240
C.2	EZGKP (Combastel, 2016)	243
D.1	LD (Adapted from Rego & Raffo (2019))	248
D.2	DA	249
D.3	FL (Adapted from Rego & Raffo (2019))	257

Notation

a	Italic lowercase letters denote scalars or vectors. We define a vector as an element column
A	Italic uppercase letters denote matrices
D^i	denotes the matrix of linear inequality constraint
d^i	denotes the vector of linear inequality constraint
D^e	denotes the matrix of linear equality constraint
d^e	denotes the vector of linear equality constraint
$\mathcal{X}$	Calligraphic uppercase variables denote sets
X	According to the context, italic uppercase letters denote random vectors
$\{X\}$	denotes a sequence of random vectors
$\bar{x}$	denotes an evaluating point, which plays the roll of mean, center, or linearization point according to the context
ϱ^a, ϱ^b	denote the DC components (or convex functions) of a function ϱ such that $\varrho = \varrho^a - \varrho^b$
$\bar{\varrho}^a(x), \bar{\varrho}^b(x)$	denote linear minorant of functions $\varrho^a(x)$ and $\varrho^b(x)$, that is, $\bar{\varrho}^s(x) = \varrho^s(\bar{x}) + F(x - \bar{x})$, where $F = \nabla_x \varrho^s(\bar{x})$ and $s = \{a, b\}$
$\tilde{(\cdot)}$	denotes an augmented vector/function
$\hat{(\cdot)}$	denotes a suboptimal solution
$\check{(\cdot)}$	denotes an auxiliary variable/result
$\overset{*}{(\cdot)}$	denotes the optimal solution of an optimization problem
$(\cdot)^\top$	denotes the transpose operator

$(\cdot)^*$	denotes the conjugate transpose operator
$(\cdot)^\dagger$	denotes the pseudo-inverse operator
$(\cdot)^{-1}$	denotes the inverse of a square matrix
$(\cdot)_{i,:}$	denotes the i th row of a matrix
$(\cdot)_{:,j}$	denotes the j th column of a matrix
P^{xx}	denotes the covariance matrix of a random vector X
$X \sim \mathcal{N}(\bar{x}, P^{\text{xx}})$	denotes a Gaussian random vector X with mean $\bar{x}$ and covariance P^{xx}
$X^\chi \sim \chi^2(n)$	denotes a chi-square random variable with n degrees of freedom, mean n and covariance $2n$
$X^\dagger$	denotes a truncated random vector
$p(x)$	denotes the probability density function of X with respect to the realization x
$c(x)$	denotes the cumulative distribution function of a given function $p(x)$
c^x	denotes the center of the mixed-uncertainty vector x
z^x	denotes an element of a zonotopic (or constrained zonotopic) set $\mathcal{X}$ related to the mixed-uncertainty vector x
g^x	denotes a realization of a Gaussian random vector X related to the mixed-uncertainty vector x
k	denotes a sample of the discrete-time instant
T_s	denotes the sampling time
$N(m, n)$	is the mathematical combination that returns all possible ways to choose n elements from the set $\{1, \dots, m\}$ such that $m \geq n$
$[x]$	denotes an interval vector (or box)
$[M]$	denotes an interval matrix
ζ^L	denotes the lower bound of an interval set
ζ^U	denotes the upper bound of an interval set
$\mathbb{N}^n$	denotes an n -dimensional space of natural numbers

$\mathbb{Z}^n$	denotes an n -dimensional space of integer numbers
$\mathbb{Z}_+^n$	denotes an n -dimensional space of positive integer numbers
$\mathbb{R}^n$	denotes an n -dimensional space of real numbers
$\mathbb{R}_+^n$	denotes an n -dimensional space of positive real numbers
$\mathbb{R}^{m \times n}$	denotes an $(m \times n)$ -dimensional space of real numbers
$0_{m \times n}$	denotes an $(m \times n)$ -dimensional matrix with zero elements
$1_{m \times n}$	denotes an $(m \times n)$ -dimensional matrix with one elements
I_n	denotes an $(n \times n)$ -dimensional identity matrix
$P^{1/2}$	denotes the Cholesky decomposition of P such that $P = P^{1/2}P^{\top/2}$
$P = P^{\top}$	denotes a symmetric matrix
$P > 0_{n \times n}$	denotes an $(n \times n)$ -dimensional positive-definite matrix
$P \geq 0_{n \times n}$	denotes an $(n \times n)$ -dimensional positive semidefinite matrix
$P < 0_{n \times n}$	denotes an $(n \times n)$ -dimensional negative-definite matrix
$P \leq 0_{n \times n}$	denotes an $(n \times n)$ -dimensional negative semidefinite matrix
$\ \cdot\ _2$	denotes the 2-norm of a vector or matrix
$\ \cdot\ _F$	denotes the Frobenius norm of a matrix
$ \cdot $	denotes the elementwise absolute value operator
$\ \cdot\ _{H_2}$	denotes the H_2 norm of a system
$\ \cdot\ _{H_\infty}$	denotes the H_∞ norm of a system
$\text{rank}(\cdot)$	denotes the rank of a matrix
$\text{erf}(\bar{x})$	denotes the error function erf evaluated at $\bar{x}$
$\nabla_x f(\bar{x})$	denotes the Jacobian matrix of $f(x)$ evaluated at $\bar{x}$
$\text{rs}(\cdot)$	denotes <i>row-sum</i> . This operator takes a given matrix $M \in \mathbb{R}^{m \times n}$ to yield a diagonal matrix such that $\text{rs}(M)_{i,i} \triangleq \sum_{j=1}^n M_{i,j} $, for $i = 1, \dots, m$
$\subset$	denotes <i>is contained into</i>
$\subseteq$	denotes <i>is contained into or equal to</i>

$\supset$	denotes <i>contains</i>
$\supseteq$	denotes <i>contains or is equal to</i>
$=$	denotes <i>is equal to</i>
$>$	denotes <i>is greater than</i>
$\geq$	denotes <i>is greater than or equal to</i>
$\leq$	denotes <i>is lesser than or equal to</i>
$<$	denotes <i>is lesser than</i>
$\approx$	denotes <i>is approximately</i>
$\in$	denotes <i>belongs to</i>
$\ni$	denotes the inverse direction of <i>belongs to</i>
$\rightarrow$	denotes <i>to</i>
$\forall$	denotes <i>for all</i>
$\times$	denotes Cartesian product
$\cap$	denotes the direct intersection among sets.
$\emptyset$	denotes an empty set
$\triangleq$	denotes <i>is equal to, by definition</i>
■	denotes the end of proofs
$\text{vert}(\cdot)$	denotes the vertices of a set
$\text{vec}(\cdot)$	denotes the vectorization of a matrix, where the columns are stacked up to yield a vector
$E[\cdot]$	denotes the expected value operator
$\text{tr}(\cdot)$	denotes the trace of a square matrix, i.e., the sum of all diagonal elements
$\det(\cdot)$	denotes the determinant of a square matrix
$\text{cov}(X, Y)$	denotes the cross correlation between the random vectors X and Y
$\text{mid}(\cdot)$	denotes the middle point of an interval vector or interval matrix
$\inf(\cdot)$	denotes infimum

$\sup(\cdot)$	denotes supremum
$\text{diam}(\cdot)$	denotes the diameter of an interval vector or interval matrix
$\text{rad}(\cdot)$	denotes the radius of an interval vector or interval matrix
$\min(x)$	denotes the smallest value of x
$\max(x)$	denotes the maximum value of x
$\text{diag}(\cdot)$	returns a diagonal matrix (from a vector) or a vector (from a square matrix)
$\text{blkdiag}(\cdot)$	denotes a block-diagonal matrix
$A^\square$	denotes the average area ratio of box
$r^\square$	denotes the average largest radius ratio of box
X^{sp}	denotes the sigma-point matrix. That is, each column of X^{sp} represents a deterministically chosen vector called <i>sigma point</i>
φ_c	denotes the desired order to fix the maximum number of constraints of a constrained zonotope
φ_g	denotes the desired order to fix the maximum number of generators of a zonotope or constrained zonotope
α	defines the significance level
η	denotes a scalar that belongs to the interval $[0, 1]$. This scalar is used to scale a biobjective function as follows: $J = (1 - \eta)J_1 + \eta J_2$
$\mathcal{B}^n$	denotes an n -dimensional unitary box composed by n unitary intervals $[-1, 1]$
$\mathcal{B}(A^x, b^x)$	denotes a set of elements $\xi \in \mathbb{R}^n$ that belongs to both the unitary box $\mathcal{B}^n$ and the linear equality constraints $A^x \xi = b^x$
$\mathcal{X} = \{G^x, \bar{x}\}$	denotes a zonotope with generator matrix G^x and center $\bar{x}$. Each column of G^x is called <i>generator</i>
$\mathcal{X} = \{G^x, \bar{x}, A^x, b^x\}$	denotes a constrained zonotope with generator matrix G^x , center $\bar{x}$, and linear equality constraints composed by matrix A^x and vector b^x
$\cap_M$	denotes the generalized intersection among CZs
$\square \mathcal{X}$	denotes an interval enclosure containing the set $\mathcal{X}$

$\mathcal{X}^F$	denotes a compact feasible set
$\diamond \mathcal{X}$	denotes the zonotope inclusion
$\square f$	denotes an interval extension of a function f , where all standard operations are replaced by interval operations and all sets are replaced by interval enclosures
$\mathcal{X}^\downarrow$	denotes a set (zonotope or constrained zonotope) whose order (number of constraints and generators) was reduced
$\mathcal{E} = \{\bar{x}, P^{\text{xx}}, \alpha\}$	denotes a confidence ellipsoid with center $\bar{x}$ and shape matrix P^{xx} such that $(x - \bar{x})^\top (P^{\text{xx}})^{-1} (x - \bar{x}) \leq \varsigma$, where $\varsigma \in \mathbb{R}$ is the greatest realization of a chi-square variable that satisfies the $(1 - \alpha)$ confidence level
$\mathcal{X}^\alpha$	denotes the confidence constrained zonotope for the mixed-uncertainty case whose minimum confidence level is $(1 - \alpha)$
$\mathcal{X}^c$	denotes the constrained confidence CZ. This set is obtained by enforcing $\mathcal{X}^F$ on the CZ $\mathcal{X}^\alpha$ via intersection
$\mathcal{I}^\alpha$	denotes the confidence box for the mixed case whose minimum confidence level is $(1 - \alpha)$
$\mathcal{I}^c$	denotes the constrained confidence box
$\oplus$	denotes Minkowski sum
$\bigoplus_{i=1}^n \mathcal{X}_i$	denotes a series of Minkowski sums, i.e., $\bigoplus_{i=1}^n \triangleq \mathcal{X}_1 \oplus \cdots \oplus \mathcal{X}_n$
$\otimes$	denotes the Kronecker product
$\odot$	denotes the elementwise product between matrices of same dimension
J^{mv}	denotes the minimum-variance objective function for random vectors
J^{eqmv}	denotes the equality-constrained minimum-variance objective function for random vectors
J^{md}	denotes the minimum-distance objective function
J^{map}	denotes the maximum-a-posteriori objective function for random vectors
J^{frob}	denotes the minimum Frobenius norm for zonotopes. This objective function is also known as <i>segment minimization</i>
J^{vol}	denotes the minimum-volume objective function for zonotopes

J^{frad}	denotes the <i>F-radius</i> of a set
J^{eqfrad}	denotes the equality-constrained <i>F-radius</i> of a set
J^{prad}	denotes the <i>P-radius</i> of a set
J^{zg}	denotes the minimum-variance objective function for the mixed-uncertainty case. In this case, the convex linear combination between J^{mv} and J^{frad} is carried out
J^{eqzg}	denotes the equality-constrained minimum-variance objective function for the mixed case. In this case, the convex linear combination between J^{eqmv} and J^{eqfrad} is carried out
$\tilde{J}^s$	denotes a normalized cost function in $[0, 1]$, where $s = \{\text{mv}, \text{eqmv}, \text{md}, \text{frad}, \text{eqfrad}, \text{zg}, \text{eqzg}\}$

Acronyms

CPU	Central processing unit
CZ	Constrained zonotope
CZDC	Set-membership nonlinear filter based on constrained zonotopes and difference of convex functions programming
CZLF	Set-membership linear filter based on constrained zonotopes for linear time-varying systems
CZR	Constrained zonotope reduction
DA	Discretization-approximation
DC	Difference of convex functions
DCMF	Mixed-uncertainty filter based on DC programming
DGF	Gaussian filter for descriptor linear time-varying systems
DH ₂ GF	Minimum H ₂ norm Gaussian filter for descriptor linear time-invariant systems
DH _∞ ZF	Minimum H _∞ norm zonotopic filter for descriptor linear time-invariant systems
DRZF	Minimum-radius zonotopic filter for descriptor linear time-varying systems
DTZF	Minimum-trace zonotopic filter for descriptor linear time-varying systems
EKF	Extended Kalman filter
EZGKP	Extended zonotopic and Gaussian Kalman predictor
FL	Function linearization
FOMF	Mixed-uncertainty filter based on the first-order Taylor extension
GRV	Gaussian random vector

H_2 GF	Minimum H_2 norm Gaussian filter for linear time-invariant systems
H_∞ ZF	Minimum H_∞ norm zonotopic filter for linear time-invariant systems
ICUT	Interval-constrained unscented transformation
INTLAB	INTerval LABoratory
IZMV	Interval zonotopic filter based on the mean value extension
KF	Kalman filter for linear time-varying systems
KP	Kalman predictor for linear time-varying systems
LBE	Lower bound of eigenvalue
LD	Linearization-discretization
LMI	Linear matrix inequalities
LP	Linear program
LPV	Linear parameter-varying
LTI	Linear time-invariant
LTIMF	Mixed-uncertainty filter for descriptor linear time-invariant systems
LTV	Linear time-varying
LTVMF	Mixed-uncertainty filter for descriptor linear time-varying systems
LTVMF-E	Mixed-uncertainty filter for descriptor linear time-varying systems with explicit calculus
MATLAB	MATrix LABoratory
MPT	Multi-parametric toolbox
MT	Mean trace
NH_∞ ZF	New H_∞ zonotopic filter for linear time-invariant systems
PDF	Probability density function
PT	Probability density function truncation
RMSE	Root-mean-square error
SKF	Stationary-gain Kalman filter

SKP	Stationary-gain Kalman predictor
SM	Segment minimization
T^{CPU}	Mean processing time
TUKF	Truncated unscented Kalman filter
UAV	Unmanned aerial vehicle
UKF	Unscented Kalman filter
UT	Unscented transformation
VM	Volume minimization
ZDC	Set-membership zonotopic filter based on difference of convex functions programming
ZEKP	Zonotopic extended Kalman predictor
ZGKF	Zonotopic and Gaussian Kalman filter for linear time-varying systems
ZGKP	Zonotopic and Gaussian Kalman predictor for linear time-varying systems
ZKF	Zonotopic Kalman filter for linear time-varying systems
ZKP	Zonotopic Kalman predictor for linear time-varying systems
ZMV	Zonotopic filter based on the mean value extension
ZOH	Zero order holder
ZR	Zonotope reduction
ZT	Zonotope transformation

Contents

1	Introduction	29
1.1	Motivation and Literature Review	29
1.1.1	Representations of Uncertainty	29
1.1.2	Set-based State Estimators	32
1.1.3	Mixed-Uncertainty State Estimators	36
1.2	Problem Formulation	38
1.2.1	Stochastic Approach	39
1.2.2	Set-based Approach	40
1.2.3	Mixed Approach	41
1.3	Objectives	42
1.4	List of Publications	43
1.5	Outline	44
2	Prolegomena	46
2.1	Random Vectors	46
2.1.1	Definitions	46
2.1.2	Operations	48
2.2	Sets	49
2.2.1	Definitions	49
2.2.2	The Dependency and Wrapping Effects	51
2.2.3	Interval-based Operations	52
2.2.4	Operations with Zonotopes	56
2.2.5	Operations with Constrained Zonotopes	58
2.3	DC Programming	61
2.4	Mixed-Uncertainty Vector	65
2.5	Optimization Problems	69
2.6	Final Remarks	72
3	Extensions of Stochastic and Set-based Approaches	73
3.1	Linear Systems	74

3.1.1	Gaussian Filter for Descriptor Systems	74
3.1.2	Minimum H_2 Norm Gaussian Filter for Descriptor Systems	80
3.1.3	Numerical Results	83
3.1.4	Minimum-Radius Zonotopic Filter for Descriptor Systems	86
3.1.5	Numerical Examples	89
3.1.5.1	LTI System	90
3.1.5.2	LTV System	92
3.1.6	Zonotopic Kalman Filter	93
3.1.6.1	Time-Varying Case	93
3.1.6.2	Time-Invariant Case	96
3.1.7	Numerical Example	99
3.2	Nonlinear Systems	100
3.2.1	Improvements for the Zonotopic Filter	100
3.2.1.1	A New Interval Zonotopic Filter	101
3.2.1.2	Novel Computational Improvement for the Minimum-Volume Approach	102
3.2.2	Quadrotor Unmanned Aerial Vehicle	108
3.2.2.1	Setup	110
3.2.2.2	Numerical Results	111
3.2.3	Set-Membership Filter based on CZs and DC Programming	114
3.2.3.1	Complexity Analysis	116
3.2.4	Numerical Examples	117
3.2.4.1	Two-State Nonlinear Process	118
3.2.4.2	Attitude Estimation	119
3.3	Final Remarks	122
4	Novel Mixed Approaches for Linear Systems	124
4.1	Zonotopic and Gaussian Kalman Filter	125
4.1.1	Numerical Example	128
4.2	Mixed-Uncertainty Filter for Descriptor LTV Systems	130
4.2.1	General Filter Design	130
4.2.1.1	Original Problem	130
4.2.1.2	Normalized Problem	137
4.2.1.3	Algorithm	140
4.2.2	Particular Case with Polytopes	141
4.2.3	Numerical Example	145
4.2.3.1	Setup	145
4.2.3.2	Discussion	145
4.3	Mixed-Uncertainty Filter for Descriptor LTI Systems	146

4.3.1	Estimator Design	146
4.3.2	Algorithm Description	151
4.3.3	Numerical Example	152
4.3.3.1	Setup	153
4.3.3.2	Discussion	155
4.4	Final Remarks	156
5	Novel Mixed Approaches for Nonlinear Systems	158
5.1	Extended Zonotopic and Gaussian Kalman Filter	159
5.1.1	Numerical Results	163
5.1.1.1	Van der Pol Oscillator	164
5.1.1.2	LPV Application	166
5.2	Nonlinear Mixed-Uncertainty State Estimators based on CZs	168
5.2.1	Algorithm Description	176
5.2.2	Special Case	176
5.2.3	Numerical Results	178
5.2.3.1	Van der Pol Oscillator	180
5.2.4	Batch Reactor	181
5.2.4.1	Simple Pendulum	184
5.3	Final Remarks	185
6	Conclusions	187
6.1	Concluding Remarks	187
6.2	Future Directions	190
A	Stochastic State Estimators: a Review	203
A.1	Linear Algorithms	203
A.1.1	Kalman Predictor	204
A.1.2	Kalman Filter	205
A.1.3	Stationary-Gain Kalman Estimators	206
A.2	Nonlinear Algorithms	208
A.2.1	Unscented Kalman Filter	209
A.2.1.1	Unscented Transformation	209
A.2.1.2	Algorithm	210
A.2.2	Truncated Unscented Kalman Filter	212
A.2.2.1	Probability Density Function Truncation	212
A.2.2.2	Interval-Constrained Unscented Transformation	212
A.2.2.3	Algorithm	214

B	Set-based State Estimators: a Review	217
B.1	Linear Algorithms	217
B.1.1	Zonotopic Kalman Predictor	218
B.1.1.1	Zonotope Reduction	218
B.1.1.2	Algorithm	219
B.1.2	Minimum-Trace Zonotopic Filter for Descriptor Systems	221
B.1.3	Minimum H_∞ Norm Zonotopic Filter	224
B.1.4	Constrained Zonotopic Linear Filter	227
B.2	Nonlinear Algorithms	229
B.2.1	Zonotopic Extended Kalman Predictor	229
B.2.2	Zonotopic Filter based on Mean Value Extension	230
B.2.2.1	Intersections among Strips and Zonotopes	230
B.2.2.2	Zonotope Nonlinear Transformation	232
B.2.2.3	Algorithm	236
C	Mixed-Uncertainty State Estimators: a Review	238
C.1	Zonotopic and Gaussian Kalman Predictor	238
C.2	Extended Zonotopic and Gaussian Kalman Predictor	240
D	Practical Aspects to Implement Zonotopic Filters	244
D.1	Continuous-Time Nonlinear Process Models	245
D.2	Approximation and Discretization Procedures	246
D.2.1	Numerical Examples	250
D.3	Nonlinear Measurement Models	254
D.3.1	Numerical Example	257
E	Worst-Case Complexity	260
E.1	Basic Operations	260
E.2	Stochastic Approaches	270
E.3	Set-based Approaches	275
E.4	Mixed-Uncertainty Approaches	290

1

Introduction

1.1 Motivation and Literature Review

1.1.1 Representations of Uncertainty

A state estimator can receive different denominations as *predictor*, *filter*, and *smoother* (Teixeira, 2008; Särkkä, 2013). These denominations separate the estimators in three classes according to the time instant for which the measurements are used to update the current state vector. The predictors assimilate only the past measurements, whereas the filters assimilate both past and current measurements. In turn, the smoothers assimilate past, current and future measurements. From predictors to smoothers, the most accurate and precise estimates are obtained, since more information is assimilated (Teixeira, 2008).

When statistical properties of uncertain variables are available, *random vectors* can be employed to characterize uncertainties based on probability density functions (PDFs) such as uniform, Gaussian, and chi square (Särkkä, 2013). Gaussian PDFs are commonly employed to approximate *a posteriori* PDFs (Kalman, 1960; Julier & Uhlmann, 2004; Teixeira et al., 2010) since a large confidence region is provided around the most likely estimate. For linear systems, the Kalman filter (KF) (Kalman, 1960) provides optimal state-estimation solutions based on the minimum-variance criterion. Since Gaussian random vectors (GRVs) are not closed under nonlinear transformations, some variations of the KF have been proposed to estimate states of nonlinear systems such as the extended Kalman filter (EKF) and the unscented Kalman filter (UKF) (Jazwinski, 2007; Särkkä,

2013). The EKF employs analytical linearization, that is, first- or second-order truncated Taylor series, whereas the UKF employs unscented transformation. Another well-known stochastic nonlinear algorithm is the particle filter (Beadle & Djuric, 1997; Ristic et al., 2004; Särkkä, 2013), which yields random samples to approximate PDFs. These nonlinear algorithms generally provide suboptimal solutions.

When the bounds of uncertain variables are known, they can be represented by *sets* such as polyhedra, polytopes, zonotopes, constrained zonotopes (CZs), parallelotopes, intervals, strips, and ellipsoids (Jaulin & Kieffer, 2001; Moore et al., 2009; Le et al., 2013b; Scott et al., 2016); these sets are illustrated in Figure 1.1 for the two-dimensional case. Polyhedron is the most possible wide set based on hyperplanes. A halfspace is a polyhedron containing a unique hyperplane to bound a given direction. A strip is a nonempty intersection between two halfspaces with parallel hyperplanes. When the polyhedron is compact (that is, bounded and closed) it is called polytope. Different representations of convex polytopes are presented. A CZ is an extension of zonotope, which additionally enforces linear equality constraints on the unitary hypercube. Thereby, CZs allow for incorporation of asymmetry of polytopes. A zonotope extends both box and parallelotope by involving any affine transformation of the unitary hypercube. Therefore, zonotopes can represent centrally symmetric polytopes using a transformation matrix containing more columns (called generators) than rows (dimension of the Euclidean space). A parallelotope is an extension of box, whose unitary hypercube is affinely transformed via a square and invertible matrix. A box is the simplest possible polytope composed of intervals. Finally, ellipsoid is a convex, compact, and centrally symmetric set defined in terms of a square and invertible matrix. Unlike zonotopes, ellipsoids are affine transformations of unitary hyperspheres.

The prior approach is favorable to treat problems involving physical limits and unknown inputs (Schweppe, 1968), for instance, where no statistical knowledge about uncertainty is available. In contrast to the stochastic setting, set-based observers provide guaranteed state-estimation solutions. Subsection 1.1.2 presents many recent contributions on this topic.

There exist cases in which a part of the uncertain variables can be statistically characterized while the other part deterministically defined. For instance, in real-world applications, dynamical processes are influenced by aleatoric errors, whose exact bounds may be unknown or approximately known. With the auxiliary of statistical tools such as high-fidelity model (Peherstorfer et al., 2017; Ghosh et al., 2022) and phenomenological model (Bastogne et al., 2010; Merrih Bayat et al., 2015), as well as strategies of stochastic model predictive control (Mesbah, 2016), random uncertainties could be properly characterized. This is not a trivial task, mainly in state estimation where not all states are measured, but it could justify the existence of random noise in several practical cases such as those in Hanebeck & Horn (2001) and Henningsson (2008). Analogously, the measurements

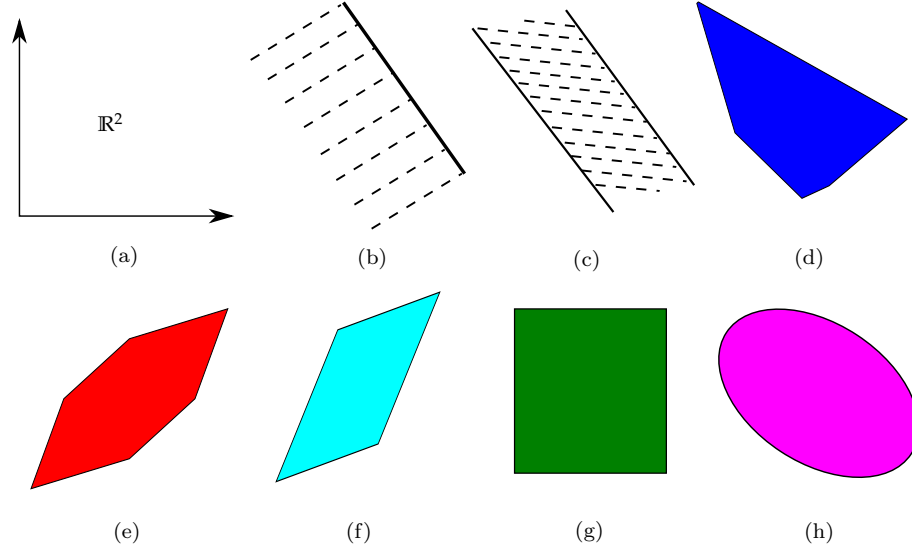


Figure 1.1: Illustrative sets in 2D: (a) Polyhedron, (b) halfspace, (c) strip, (d) CZ, (e) zonotope, (f) parallelotope, (g) box, and (h) ellipsoid.

may be susceptible to systematic errors (related to inner characteristics or the handling of instruments) (Hanebeck & Horn, 2001; Henningsson, 2008). In this case, no statistical property is provided, but the range of measurements could be employed to characterize the existing uncertainties. Likewise, a statistical information can be used to define random vectors in high-probability confidence sets (Scott et al., 2013; Marseglia et al., 2014; Li et al., 2021), and thereby, the different uncertainties could be matched in the set-based framework.

In order to treat this type of problem exploiting each information through own tools, the uncertainties can be mixed in a unified framework, yielding the so-called mixed-uncertainty algorithms. Subsection 1.1.3 presents a range of existing mixed-uncertainty algorithms and shows that the advances obtained in set-based estimation have motivated the proposal of new mixed-uncertainty algorithms.

This thesis aims at providing further contributions to the stochastic, set-based, and mixed-uncertainty frameworks. Our focus is on mixed uncertainties, but the separate approaches should be firstly investigated since they compose the mixed-uncertainty algorithms. This investigation is important to raise possible challenges that each approach will imply in the mixed-uncertainty framework. For instance, Kalman estimators cannot properly accord with state inequality constraints since GRVs have unbounded support by definition. Although this issue is properly treated with set-based estimators, few works take state constraints into account since some classes of sets are not closed under certain operations and/or the observer structure is not suitable. Then, by understanding preliminary strengths and challenges of each approach, we can extend current results to improve performance requirements as precision, accuracy, and computational burden.

Since there exist many works about stochastic filtering (Jazwinski, 2007; Teixeira et al.,

2010; Simon, 2010; Särkkä, 2013), by convenience, we here present more set-based than stochastic approaches. Although not all set-based content has been employed to compose mixed-uncertainty state estimators, it addresses many new possibilities to design mixed-uncertainty filters, whose ideas can be exploited in the future.

1.1.2 Set-based State Estimators

With respect to the stochastic framework, the set-based observers started another branch of state estimation. The basis idea of the set-based estimation is to correct forecast sets using consistency sets. This procedure can be executed through either intersection or gain matrix, separating the set-based observers in, respectively, two different settings (Pourasghar et al., 2019): *set-membership observers* and *interval observers*. Each setting has advantages and drawbacks, with a similarity being the computation of guaranteed sets, that is, inclusion of true states. True states refer to actual trajectories of a system. By employing intersection, the first framework, in principle, disregards optimization problems and can return precise solutions as in Rego et al. (2021). However, intersection does not ensure stability of the estimator and can deteriorate the accuracy of punctual estimates due to asymmetry, which has been early published in our paper de Paula et al. (2022c). When employing minimum-trace criterion or system norms, the interval observers framework can provide particular benefits such as stability, reachability analysis with an explicit link between initial and current states, low-cost computation, and direct treatment for linear parameter-varying (LPV) systems (Wang et al., 2018b,a; Tang et al., 2019a). Conversely, there exists the need of solving optimization problems, whose formulation will influence both precision and accuracy of estimator.

Especially over the last two decades, many papers presented relevant contributions on the set-based state estimation. Next, we present a historical perspective on this topic. Among the set-based estimators pioneers, the work of Schweppe (1968) has proposed a linear algorithm based on ellipsoids. In this case, Minkowski sum and intersection of ellipsoids have been addressed. Afterwards, Durieu et al. (1996) have proved that the optimization problems involving these operations are convex under the trace and volume criteria. In Grocholsky et al. (2006), a technique based on the ellipsoidal filter from Schweppe (1968) has been applied to robotic systems, making use of the bounded nonlinear measurements to define a given ellipsoid. Recently, Li et al. (2020) have proposed a nonlinear filter based on ellipsoids, analytical linearization, and nonconvex and inhomogeneous quadratic inequality constraints on the states. This algorithm treats the time-varying linearization remainder through a semidefinite programming problem and has been numerically tested in a moving target tracking application.

For set-based state estimation, both representations of halfspace and vertex have been a little motivated, because they offer high computational burden to propagate *polytopes*.

On the other hand, they are useful in certain applications where the complexity of polytopes is fixed or computationally feasible, such as robust filtering (Wang et al., 2018b; Rego & Raffo, 2019), difference of convex functions (DC) programming problems (Horst & Thoai, 1999; Alamo et al., 2008), change of polytope configurations to combine prediction (using interval bundles) with update (using CZs) (Rego et al., 2018b), Pontryagin difference of zonotopes (Althoff, 2015), and projection of polytopes (Spletzer & Taylor, 2003). In this sense, Spletzer & Taylor (2003) have proposed an approach to cooperative localization for multirobot teams, whose idea is similar to the work of Grocholsky et al. (2006). That is, the bearing and range measurements induce linear inequality constraints on the configuration space of the robot team, which are consistent to sensor measurements. Due to the augmented nature of the configuration space, the set of constraints that defines a polytope in halfspace representation is projected to estimate certain parameters of interest of that configuration. As the exact projection may be too expensive, linear programming problems have been proposed to overestimate it using the original polytope as the constraints.

Intervals (or boxes) are sets that are as efficient as ellipsoids regarding numerical implementation. They are closed under nonlinear transformations and offer low-cost computations. However, interval operations may yield conservative enclosures. Aiming at computational efficiency, Di Marco et al. (2004) have promoted appealing directions to build low-cost algorithms and solve problems involving autonomous robot navigation. To perform the operations required by the algorithms, the authors have indicated the use of efficient representations as ellipsoids, boxes, and parallelotopes. However, only the two latter sets have been discussed. In Jaulin (2009), a nonlinear observer based on intervals has been proposed, which is robust with respect to outliers. This algorithm has been executed in a case study involving an underwater robot. The work of Efimov et al. (2013) has proposed two linear algorithms based on intervals, which have exploited similarity transformation and cooperative (Metzler) form. In Meslem et al. (2018), a linear observer based on intervals has been proposed. This algorithm has been developed under both minimum robustly positive invariant set and H_∞ norm. Unlike the conventional observers based only on the gain matrix, the work of Wang et al. (2018c) has proposed a linear observer based on both intervals and additional matrices, which play the role of degrees of freedom to better reduce the H_∞ norm of the estimation error. These additional matrices are related by means of linear equality constraints.

In parallel to interval arithmetic, many contributions have been addressed to convex polytopes, since more general sets than boxes can be defined. In this context, the parallelotopes appear as specific affine mapping of unitary boxes. Parallelotopes are also efficient representations that allow for mitigation of part of the wrapping effect generated by interval computations. In Chisci et al. (1996), a linear state estimator based on parallelotopes has been proposed. This algorithm recursively computes minimum-volume parallelotopes

in polynomial time. To build parallelotopic and practical observers for nonlinear systems with unknown inputs, [Rapaport & Gouzé \(2003\)](#) have provided assumptions for observable and unobservable systems.

The most general affine transformation of unitary boxes defines the so-called *zonotopes*. This class has been studied since the 1960s ([Bolker, 1969](#)). However, it became recognized for state estimation only in 1998 with the work of [Kühn \(1998\)](#). This paper has made explicit many properties related to zonotopes. In addition, the zonotope class has been presented as a low-cost representation (unlike polytopes) that has been able to control the wrapping effect (unlike intervals) and closed under the Minkowski sum (unlike ellipsoids and parallelotopes). Taking advantage of these properties, [Combastel \(2003\)](#) has proposed a linear algorithm based on zonotopes. This paper has mitigated two challenges for zonotopes: intersection and order growth (number of generators). The intersection between zonotopes has been solved by overestimating it by an interval intersection with singular value decomposition in the data-assimilation step. Regarding the second issue, an order reduction algorithm has been proposed to control the complexity of zonotopes. Afterwards, this observer was extended to nonlinear systems using the truncated Taylor series ([Combastel, 2005](#)).

The zonotopic filter based on the mean value extension, here called ZMV, has been proposed for the nonlinear case in [Alamo et al. \(2005\)](#). In the data-assimilation step, the ZMV performs intersection between zonotopes and strips based on segment minimization (SM) or volume minimization (VM). Due to the need to employ interval arithmetic, the wrapping and dependency effects may affect both precision and convergence of the ZMV. For this reason, [Alamo et al. \(2008\)](#) have proposed an alternative methodology based on DC programming to replace interval arithmetic. Conversely, outer approximation of zonotopes by boxes or parallelotopes is required to fix the number of vertices. Motivated by the speed and precision of the SM and VM, respectively, [Le et al. \(2013a\)](#) have sought a tradeoff between those performance criteria to accelerate the VM without losing much precision, as in the SM; the new criterion has been called *P*-radius. In [Le et al. \(2013b\)](#), the ZMV has been individually performed with the SM, VM, and *P*-radius minimization to feed a model predictive controller with purpose of stabilizing a magnetic levitation system. The work of [Rego & Raffo \(2019\)](#) has proposed an approximate version of the ZMV to feed a mixed H_2/H_∞ controller, which has been tested in a case study involving a tilt-rotor UAV with load transportation. Next, [García et al. \(2020\)](#) have proposed an algorithm based on the ZMV for distributed estimation and control of vehicle formations. This algorithm has been illustrated with numerical examples involving a fleet of vehicles.

The minimum-volume criterion of [Alamo et al. \(2005\)](#) has been computationally improved in [Bravo et al. \(2006\)](#) since nonlinear programs have been replaced by explicit calculus with candidate zonotopes. However, to exactly employ such improvement, the volume of every candidate zonotope must be computed. This drawback could be miti-

gated by exploiting the convex nature of the original volume minimization, from which a convenient format for the optimization problem can be extracted. Not restricted to the volume minimization, the intersection between strip and zonotope could be used to extend the ZMV to enforce state inequality constraints with the format of zonotopes.

Novel paradigms between set-based and stochastic approaches have been introduced in [Combastel \(2015\)](#) through the criterion F_w -radius. A linear algorithm based on zonotopes has been proposed, whose performance is evaluated in terms of a square matrix related to the Frobenius norm. This algorithm has a predictor structure and is here called ZKP. Due to the similarities with the Kalman predictor (KP) ([Kalman, 1960](#)), [Wang & Puig \(2016\)](#) have extended the ZKP following the EKF idea, which employs analytical linearization; the resulting observer is here called zonotopic extended Kalman predictor (ZEKP). This algorithm has been applied to a quadrotor UAV. The ZEKP algorithm is not rigorously robust due to the lack of the guaranteed bounds on the linearization error. However, approaches as first-order extension ([Combastel, 2005](#)) and DC programming ([Alamo et al., 2008](#)) could be used to reach the robustness. The zonotopic observer from [Combastel \(2015\)](#) has also been extended in [Orihuela et al. \(2018\)](#) to a distributed setting for large-scale, state-coupled plants. The estimator from [Combastel \(2015\)](#) has also inspired the work of [Wang et al. \(2018a\)](#) to propose a zonotopic Kalman observer for descriptor linear systems. Additionally, the paper has proposed an observer based on the W -radius minimization (similar to P -radius). According to this same observer structure, [Wang et al. \(2018b\)](#) have proposed a state estimator based on H_∞ norm to address descriptor LPV systems. The resulting algorithm has been illustrated in a case study with a truck-trailer system. Afterwards, the work of [Wang et al. \(2018a\)](#) has been extended to switched systems in [Ifqir et al. \(2020\)](#) using the W -radius minimization, with the proposed algorithm being validated in a case study with vehicle lateral dynamics.

According to the idea of degrees of freedom exploited in [Wang et al. \(2018c\)](#), an observer based on zonotopes has been proposed in [Tang et al. \(2019a\)](#). This paper has developed an efficient strategy to recursively compute the smallest box containing the state zonotope. In [Tang et al. \(2019b\)](#), a special case of [Tang et al. \(2019a\)](#) has been derived that has also considered the H_∞ norm and reachability analysis. The observer structure from [Tang et al. \(2019a\)](#) has also inspired [Tang et al. \(2020\)](#) to propose a state estimator for descriptor linear systems, where the additional design matrices (degrees of freedom) are optimally computed by solving a quadratic problem with linear equality constraints. In this case, the Lagrange multipliers are employed to formulate an unconstrained quadratic optimization problem and, thereby, to make explicit the solutions.

Some developments performed by [Wang et al. \(2018a\)](#), [Tang et al. \(2019a\)](#) and [Tang et al. \(2020\)](#) could be exploited in stochastic filtering to yield novel state estimators with good perspectives of precision and/or processing time. The main motivation is the presence of degrees of freedom (additional design matrices), which play the role of slack

variables and are able to treat descriptor systems. The degrees of freedom could be also used in minimum-volume formulations to yield tighter solutions than those obtained by means of minimum-trace formulations.

Aiming at extending zonotopes to complete the intersection operation, the class of CZs has been proposed in [Scott et al. \(2016\)](#). Basically, CZs have linear equality constraints in addition to the unitary hypercube from zonotopes. CZs are equivalent to convex polytopes. However, they maintain the computational benefits of zonotopes. The use of CZs brings up relevant benefits to both estimate states and detect faults of linear systems, as illustrated in [Scott et al. \(2016\)](#). Motivated by this novelty, [Rego et al. \(2018b\)](#) have proposed a state estimator based on interval extension, interval bundles, and CZs for nonlinear systems with linear measurement model. For nonlinear systems with linear measurement model, [Rego et al. \(2020a\)](#) have extended both works of [Alamo et al. \(2005\)](#) and [Combastel \(2005\)](#) using CZs. The novel algorithms have been executed in a case study with a quadrotor UAV. Recently, [Rego et al. \(2021\)](#) have extended the work of [Rego et al. \(2020a\)](#) to treat both nonlinear measurement models and nonlinear equality constraints on the states, called *invariants*. These algorithms have been implemented in a case study to estimate the attitude of a flying robot. Although more precise than the original zonotopic observers, the algorithms of [Rego et al. \(2021\)](#) still carry the sensitivity to the wrapping and dependency effects due to (mainly) the use of interval arithmetic over nonlinear operations. This drawback could be mitigated by employing DC programming (as done in [Alamo et al. \(2008\)](#)), resulting in a third algorithm whose motivation is to replace the use of the mean value and first-order extensions ([Rego et al., 2021](#)) when the interval effects imply conservative results or divergence of estimator. However, the number of vertices must be fixed through the outer approximation of CZs by boxes or parallelotopes to control the computational burden.

1.1.3 Mixed-Uncertainty State Estimators

State estimators are often developed assuming a unique type of uncertainty. A great challenge is to combine approaches to better solve some state-estimation problems. For instance, [Morrell & Stirling \(1991\)](#) and [Chen et al. \(1997\)](#) have extended the KF ([Kalman, 1960](#)) to incorporate interval parameters under the Bayesian and minimum-variance perspectives. In [Balestrino et al. \(2006\)](#) and [Rego et al. \(2018a\)](#), the stochastic and set-based approaches have been combined to improve the accuracy of the algorithms. Likewise, [Combastel \(2016\)](#) and [Pan et al. \(2022\)](#) have merged stochastic and set-bounded uncertainties to design unified algorithms.

When it comes to combining uncertainties in state estimators, three research fields are highlighted in this thesis. The first one consists of switching between (or combining) different algorithms given set-bounded uncertainties ([Balestrino et al., 2006](#); [Chabane](#)

et al., 2014; Rego et al., 2018a; Meslem & Martinez, 2019). When the switching mode is chosen, an algorithm is run at a time; when the combination mode is chosen, all algorithms are run at once. These methodologies are appealing to mitigate the weaknesses of the individual algorithms, related to computational issues, precision or accuracy. The second field consists of defining hybrid uncertainties from compact sets (Scott et al., 2013; Marseglia et al., 2014; Mesbah, 2016; Villanueva & Houska, 2020; D’Jorge et al., 2020). In other words, the bounds of compact sets are also used to define random vectors, whose result is a hybrid method containing both confidence and robustness senses. This technique is useful to accelerate fault diagnosis and to formulate control strategies with probabilistic constraints that imply large attraction domains. In this case, the robustness of each method relies on the preset confidence level. Finally, the third field consists of merging different representations for uncertainties into a same estimator (Hanebeck & Horn, 2001; Henningsson, 2008; Noack et al., 2012; Combastel, 2015; Sun et al., 2019; Pan et al., 2022; Li et al., 2021). Since both stochastic and set-bounded uncertainties are present in the third case, the state-estimation problem becomes even more challenging. In this work, we refer to this scenario as *mixed uncertainty*.

Advantageously, mixed-uncertainty algorithms allow to combine different information about dynamics and measurements without the need of setting a unique type of uncertainty. For instance, KF would require that all uncertainties (including those non-Gaussian) were approximated by GRVs, while set-based algorithms would demand that all uncertainties (including those whose bounds are not exactly known) were represented by sets. Moreover, stochastic and set-theoretic tools can be employed to individually manipulate the corresponding uncertainties. In Henningsson (2008), a double integrator process (trajectory estimation) is affected by white noise, while the measurements are coarsely quantized through rounding to nearest integers. In Teixeira et al. (2008a), the dynamics of a Van der Pol oscillator are affected by an unknown input but with known bounds, and the measurements are influenced by white noise. In Li et al. (2021), a real application has been addressed in which a robot assists the dressing of a long-sleeved jacket onto a human arm; in this case, the designed algorithm is subject to Gaussian processes and approximation errors.

Due to closeness between shape and covariance matrices (Schweppe, 1968), the first mixing of uncertainties has involved ellipsoids and GRVs (Hanebeck & Horn, 2001; Henningsson, 2008; Noack et al., 2012). Afterwards, new developments have been addressed to zonotopes to mitigate conservatism caused by Minkowski sum of ellipsoids (Combastel, 2015, 2016). Recently, CZs (Scott et al., 2016) have been applied to mixed-uncertainty filtering in Pan et al. (2022) to approximate the parameter mapping by a convex polytope, whose asymmetry is efficiently propagated over Minkowski sum and affine transformation.

In the nonlinear context, we highlight the following works with mixed uncertainties. Combastel (2016) has extended the linear zonotopic and Gaussian Kalman estimator of

Combastel (2015) (here called ZGKP) using quasi-LPV approach; the resulting algorithm is here called EZGKP. Aiming at improvement of precision and accuracy, both the ZGKP and EZGKP algorithms could be extended to filters. Furthermore, quadratic programs could be derived to the novel state estimators with the goal of enforcing equality and inequality constraints on the states. Sun et al. (2019) have proposed a minimum-variance state estimator based on GRVs and ellipsoids where analytical linearizations have been applied to nonlinear models like the extended Kalman filter, that is, disregarding the linearization remainder. Li et al. (2021) have designed a zonotopic state estimator to approach affine Gaussian noises and finite time horizon. In this work, the process and measurement models have been learned with a limited amount of data, yielding uncertainties called *epistemic*. In order to analytically obtain linearized functions with compact input sets, and thereby, to treat the original problem in the set-membership observers framework, Li et al. (2021) have proposed procedures to bound the GRVs and the linearization errors by boxes with high probability. The performance criterion used is the segment minimization of Alamo et al. (2005), since zonotopes and strips are intersected.

Here, mixed uncertainties are treated in the *interval observers framework*, where state forecasts and measurement errors are combined by means of a Kalman-like gain to yield *a posteriori* state estimates (Pourasghar et al., 2019). The justification is that such uncertainties are defined as the summation of real values. That is, the mixing is defined in the Euclidean space and, then, elements from sets and realizations from random vectors are taken into account. Therefore, mixed-uncertainty filters should comply with the superposition principle.

If two types of uncertainty are assumed for dynamical systems, then two different objectives must be pursued. As these objectives are conflicting, they characterize biobjective problems, which are commonly solved in terms of monobjective problems. Rigorously, optimization problems with more than one objective should be studied in light of the Pareto curve, which relates solutions from the variable space to the objective space. Our main interest is to obtain global optima, since they yield optimal costs on the Pareto-optimal front, as illustrated in Figure 1.2. So far, works on mixed uncertainty (Henningsson, 2008; Noack et al., 2012; Combastel, 2015, 2016; Sun et al., 2019) have solved convex monobjective problems using the weighted-sum approach, with the choice of the scalarization parameter being let free to the user.

1.2 Problem Formulation

Consider the discrete-time nonlinear dynamical system

$$x_k = f(x_{k-1}, u_{k-1}, w_{k-1}, k-1), \quad (1.1)$$

$$y_k = h(x_k, v_k, k), \quad (1.2)$$

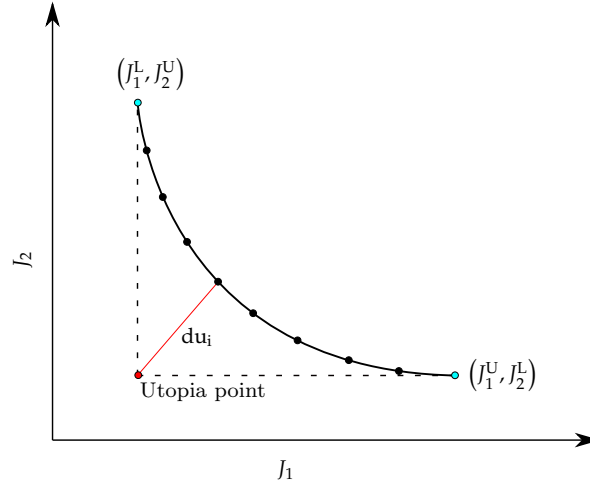


Figure 1.2: As the costs J_1 and J_2 are conflicting, they characterize a biobjective optimization problem. The Pareto-optimal front (—) is estimated by a finite set of points ($\cdot$), whose cyan color is used to point out the limiting values. The smallest distance du_i between the estimated curve and the Utopia point (red point) characterizes the best tradeoff between J_1 and J_2 in the sense that both costs are close to the knee of the curve.

where $f : \mathbb{R}^n \times \mathbb{R}^p \times \mathbb{R}^q \times \mathbb{N} \rightarrow \mathbb{R}^n$ and $h : \mathbb{R}^n \times \mathbb{R}^r \times \mathbb{Z}_+ \rightarrow \mathbb{R}^m$ are the known process and measurement models, respectively, and $x_k \in \mathbb{R}^n$ is the state vector to be estimated, which is assumed to satisfy the nonlinear equality constraints

$$g(x_k, k) = 0_{m_c \times 1} \quad (1.3)$$

and the linear inequality constraints

$$D_k^i x_k \leq d_k^i \quad (1.4)$$

such that $g : \mathbb{R}^n \times \mathbb{Z}_+ \rightarrow \mathbb{R}^{m_c}$, $D_k^i \in \mathbb{R}^{n_c \times n}$, and $d_k^i \in \mathbb{R}^{n_c}$. Equations (1.3) and (1.4) depict, respectively, general invariants and convex polyhedra, which will be incorporated by state estimators via either state-space representation or descriptor representation. The vectors $w_{k-1} \in \mathbb{R}^q$ and $v_k \in \mathbb{R}^r$ represent process and measurement noise terms, respectively. For $k \in \mathbb{Z}_+$, the measured output vector $y_k \in \mathbb{R}^m$ and the deterministic input vector $u_{k-1} \in \mathbb{R}^p$ are assumed to be known. Three different assumptions are made on the variables w_{k-1} , v_k , and x_0 .

1.2.1 Stochastic Approach

For the stochastic approach, we assume that the noise terms belong to independent stochastic processes, with zero mean, and covariance matrices $Q_k = E[w_k w_k^\top]$ and $R_k = E[v_k v_k^\top]$, such that $E[w_k w_j^\top] = 0_{q \times q}$ and $E[v_k v_j^\top] = 0_{r \times r}$ for all $k \neq j$, with $E[\cdot]$ being the expected value operator, $(\cdot)^\top$ the transpose operator, and $0_{q \times q}$ the zero matrix. In turn,

the initial state term x_0 is assumed to be a random vector, which is approximated by a GRV with mean $\hat{x}_0$ and covariance $\hat{P}_0^{\text{xx}}$. The mean $\hat{x}_0$ and the covariance matrices Q_{k-1} , R_k , and $\hat{P}_0^{\text{xx}}$ are assumed to be known. Therefore, the stochastic filtering specific problem is defined in Problem 1.2.1.

Problem 1.2.1 (Stochastic Filtering). Given the input vector u_{k-1} , the measurements $\{y_1, \dots, y_k\}$, the GRVs W_{k-1} , V_k , and $\hat{X}_0$, with $E[W_{k-1}] = 0_{q \times 1}$ and $E[V_k] = 0_{r \times 1}$, and assuming that the prior GRVs are uncorrelated, the goal is to estimate the *mean* $\hat{x}_k$ and the *covariance* $\hat{P}_k^{\text{xx}}$ of the *a posteriori* PDF $p(x_k | (y_1, \dots, y_k))$, $\forall k \geq 1$, for the dynamical system given by (1.1)-(1.4). For Kalman-based estimators, the minimum-variance objective function is defined as

$$J_k^{\text{mv}}(K_k) \triangleq \text{tr}(\hat{P}_k^{\text{xx}}), \quad (1.5)$$

where $\text{tr}(\cdot)$ represents the trace of a square matrix, “mv” denotes *minimum variance*, and K_k is the gain matrix that minimizes J_k^{mv} .

Remark 1.2.1. Stochastic estimators can also define the mode of the *a posteriori* PDF as the objective function, that is, $J_k^{\text{map}} \triangleq -p(x_k | (y_1, \dots, y_k))$ (Prakash et al., 2010; Rengaswamy et al., 2013), where “map” denotes *maximum a posteriori*.

1.2.2 Set-based Approach

For the set-based approach, the noise terms w_{k-1} and v_k are unknown but bounded by the corresponding sets $\mathcal{W}_{k-1}$ and $\mathcal{V}_k$. Although the uncertainties can be statistically known (a truncated Gaussian PDF for instance), the focus is to represent the bounds of uncertainty. Moreover, the initial state x_0 is assumed to belong to the initial set $\hat{X}_0$. The sets $\mathcal{W}_{k-1}$, $\mathcal{V}_k$, and $\hat{X}_0$ are assumed to be known. Thus, the basis filtering problem involving sets can be defined as: set-membership filtering (Problem 1.2.2) and interval filtering (Problem 1.2.3).

Problem 1.2.2 (Set-membership Filtering). Given the input vector u_{k-1} , the measurements $\{y_1, \dots, y_k\}$, and the sets $\mathcal{W}_{k-1}$, $\mathcal{V}_k$, and $\hat{X}_0$, the goal is to estimate the set $\hat{X}_k$, $\forall k \geq 1$, for the dynamical system given by (1.1)-(1.4). Specifically, $\hat{X}_k$ is obtained by means of intersection between a forecast set and a correction set. For zonotopes, $\hat{X}_k$ is completely characterized by the estimates of *generator matrix* $\hat{G}_k^x$ and *center* $\hat{x}_k$. To achieve that, two objective functions can be minimized individually, namely the *Frobenius norm* of $\hat{G}_k^x$ and the *volume* of the zonotope $\hat{X}_k$, which are defined (respectively) as

$$J_k^{\text{frob}}(\kappa_k) \triangleq \|\hat{G}_k^x\|_F = \sqrt{\text{tr}(\hat{G}_k^x (\hat{G}_k^x)^T)}, \quad (1.6)$$

$$J_k^{\text{vol}}(\kappa_k) \triangleq \text{Vol}(\hat{X}_k) = 2^n \sum_{i=1}^{N(n_g, n)} |\det(T_i)|, \quad (1.7)$$

where κ_k is the design vector that minimizes J_k^{frob} and J_k^{vol} , $\sqrt{\cdot}$ is the square-root operator, $\det(\cdot)$ is the matrix determinant, $|\cdot|$ is the absolute value operator, $N(n_g, n)$ is the mathematical combination that returns all possible ways to choose n elements from a set with size $n_g \geq n$, and $T_i \in \mathbb{R}^{n \times n}$ denotes all matrices that can be obtained by taking n columns from the matrix $\hat{G}_k^x$. For CZs, linear equality constraints given by $\hat{A}_k^x \in \mathbb{R}^{n_h \times n_g}$ and $\hat{b}_k^x \in \mathbb{R}^{n_h}$ are additionally necessary to characterize $\hat{X}_k$. Since CZs explicitly compute intersections, there is no optimization problem to be solved.

Remark 1.2.2. The product $\hat{G}_k^x (\hat{G}_k^x)^\top$ in (1.6) yields a matrix called *covariation* (Combastel, 2015), whose trace defines the *F-radius* of the zonotope $\hat{X}_k$. Both *F-radius* and Frobenius norm then quantify the covariation matrix. Alternatively, the zonotopic estimators can be defined through the *P-radius* of the zonotope $\hat{X}_k$ as the performance criterion, which is given by $J_k^{\text{prad}} \triangleq \max_{x_k \in \hat{X}_k} \|x_k - \hat{x}_k\|_{2,P}^2$, where P is a positive-definite design matrix (Le et al., 2012, 2013c,b).

Problem 1.2.3 (Interval Filtering). Given the input vector u_{k-1} , the measurements $\{y_1, \dots, y_k\}$, and the sets $\mathcal{W}_{k-1}$, $\mathcal{V}_k$, and $\hat{X}_0$, the goal is to estimate the set $\hat{X}_k$, $\forall k \geq 1$, for the dynamical system given by (1.1)-(1.4). Specifically, $\hat{X}_k$ is obtained by means of an observer structure from which a Kalman-like gain combines a forecast set with a correction set. For zonotopes, the *F-radius* of $\hat{G}_k^x$ can be minimized, which is defined as

$$J_k^{\text{frob}}(K_k) \triangleq \|\hat{G}_k^x\|_F^2, \quad (1.8)$$

where K_k is the gain matrix that minimizes J_k^{frob} . For CZs, the set $\hat{X}_k$ is firstly outer-approximated by a zonotope $\tilde{X}_k$ as done in Pan et al. (2022), after J_k^{frob} is employed.

Remark 1.2.3. The centers of $\mathcal{W}_{k-1}$ and $\mathcal{V}_k$ could be nonzero, but they are here canceled in the interval filtering to keep the standard treatment done in the literature (Combastel, 2015; Wang & Puig, 2016; Wang et al., 2018a,b; Tang et al., 2019a,b, 2020). This issue has not been clarified yet, but it will be useful to prove the unbiased property of our interval observers.

Remark 1.2.4. Problem 1.2.3 can also be defined with H_∞ norm, as done in Tang et al. (2019a) and Tang et al. (2019b) for unconstrained linear time-invariant (LTI) systems. Accordingly, for zonotopes, we could minimize the smallest box containing the set $\hat{X}_k$ instead. These possibilities will be exploited afterwards.

1.2.3 Mixed Approach

In the scope of state estimation with mixed uncertainties, the vectors w_{k-1} , v_k , and x_0 are composed of three components, namely: mixed center $(c_{k-1}^w, c_k^v, c_0^x)$, constrained zonotopic $(z_{k-1}^w, z_k^v, z_0^x)$, and Gaussian $(g_{k-1}^w, g_k^v, g_0^x)$, whose letters c , z , and g denote center, CZ, and Gaussian, respectively.

and GRV, respectively. Here, we consider $c_{k-1}^w = 0_{q \times 1}$ and $c_k^v = 0_{r \times 1}$. Because of set-based and stochastic parcels, there exist two objectives to be reached simultaneously. In the literature, the biobjective optimization problem is often transformed to a monobjective optimization problem via weighted-sum approach. Such a transformation is called *scalarization* (Boyd et al., 2004). By employing this transformation, the basis filtering problem involving mixed uncertainties is formulated in Problem 1.2.4.

Problem 1.2.4 (Mixed-Uncertainty Filtering). Consider the dynamical system in (1.1)-(1.4), for which the input vector u_{k-1} and the measurements $\{y_1, \dots, y_k\}$ are known. Given center estimate $\hat{c}_0^x$, GRVs W_{k-1} , V_k , and $\hat{X}_0$, and CZs $\mathcal{W}_{k-1}$, $\mathcal{V}_k$, and $\hat{X}_0$, the goal is to determine the mixed state estimates given by $\hat{c}_k^x$, $\hat{X}_k$, and $\hat{X}_k$, for $k \geq 1$, based on the minimum-variance criterion. To achieve that, we define the monobjective cost function

$$J_k^{zg}(K_k) \triangleq (1 - \eta)J_k^{\text{frad}} + \eta J_k^{\text{mv}}, \quad (1.9)$$

where $\eta \in [0, 1]$ is a known weight parameter, J_k^{mv} and J_k^{frad} are defined in (1.5) and (1.8), respectively, and K_k is the gain matrix that minimizes J_k^{zg} .

1.3 Objectives

The main goal of this thesis is to propose linear and nonlinear state estimators with mixed-uncertainty representation. Here, stochastic uncertainties are characterized by GRVs, whereas set-bounded uncertainties are characterized by zonotopes or CZs. Next, the main objectives are studied:

- To propose novel mixed-uncertainty algorithms by combining stochastic with set-based techniques.
- To extend mixed-uncertainty predictors to mixed-uncertainty filters.
- To discuss some practical aspects to implement state estimators given continuous-time and nonlinear dynamical systems with sampled measurement data.

Specifically, the following issues are investigated:

1. To relate the H_2 norm of LTI uncertain dynamical systems to the steady-state variance to extend fixed-gain estimators according to the observer structure revisited in Tang et al. (2019a), where linear equality constraints (degrees of freedom) are enforced.
2. To extend the KP to linear time-varying (LTV) descriptor systems according to the observer structure revisited in Tang et al. (2020), where linear equality constraints (degrees of freedom) are enforced.

3. To rewrite the volume minimization problem of the ZMV as a linear program (LP).
4. To introduce interval constraints on the state vector for the ZMV using an intersection among strips and zonotopes.
5. To propose the interval minimum-radius criterion to the observer structure from [Tang et al. \(2020\)](#), and rewrite the resulting optimization problem as an LP.
6. To treat the linearization error of the ZEKP using guaranteed methods (DC programming and first-order Taylor extension) and to extend it to filter.
7. To propose the filter version of the ZGKP and EZGKP algorithms.
8. To propose a Pareto analysis to yield optimal solutions in mixed-uncertainty algorithms.
9. To introduce inequality and equality constraints on the state vector for the mixed-uncertainty algorithms.
10. To formulate mixed-uncertainty algorithms such that the minimum-variance optimization problems can be solved explicitly or through quadratic programs.
11. To introduce analysis with CZs in mixed-uncertainty algorithms based on zonotopes.

1.4 List of Publications

Next, we point out all scientific publications developed during the doctoral course:

- ([de Paula et al., 2019](#)). de Paula, A. A., Raffo, G. V. and Teixeira, B. O. S. (2019). Zonotopic and Gaussian state estimator: the predictor and filter algorithms, *Anais do 14o Simpósio Brasileiro de Automação Inteligente*, pp. 1–6.
- ([de Paula et al., 2020](#)). de Paula, A. A., Campos, V. C. d. S., Raffo, G. V. and Teixeira, B. O. S. (2020). Novel Gaussian state estimator based on H2 norm and steady-state variance, *Anais do XXIII Congresso Brasileiro de Automática*, pp. 1–7.
- ([de Paula et al., 2022c](#)). de Paula, A. A., Raffo, G. V. and Teixeira, B. O. S. (2022). Zonotopic Filtering for Uncertain Nonlinear Systems: Fundamentals, Implementation Aspects, and Extensions [Applications of Control], *IEEE Control Systems Magazine* 42(1): 19–51.
- de Paula et al. Set-based state estimation for discrete-time constrained nonlinear systems: an approach based on constrained zonotopes and DC programming. **Submitted.**

- Tottoli et al. Set-based state estimation of a Li-ion cell using DC programming and constrained zonotopes. **Submitted.**
- de Paula et al. Extended Mixed Filtering based on Zonotopic and Gaussian Uncertainties for Discrete-Time Nonlinear Systems. **Submitted.**
- de Paula et al. Minimum-Radius Criterion for a Zonotopic State Estimator based on Degrees of Freedom. **To be submitted.**
- de Paula et al. State Estimators for Discrete-Time Descriptor Linear Systems with Mixed Uncertainties and State Constraints. **To be submitted.**
- de Paula et al. Mixed-Uncertainty Nonlinear State Estimators with Linear Approximations. **To be submitted.**

1.5 Outline

The chapters of this thesis are structured as follows.

- Chapter 2 provides theoretical background about stochastic, set-based, and mixed uncertainties such as definitions, operations, and initial results. In order to support the formulation of constrained optimization problems involving those uncertainties, as well as the solution of unconstrained optimization problems, Section 2.5 is presented.
- In Chapter 3, new stochastic and set-based state estimators are proposed, which are extensions of those revisited in Appendices A and B. Specifically, Sections 3.1.1 and 3.1.2 present stochastic filters that extend existing predictors by adding design matrices and introducing analysis for descriptor systems. Subsection 3.1.4 presents a new cost criterion to improve the precision of an existing algorithm. Subsection 3.1.6 presents the extension from a prediction to filter, as well as a low-cost algorithm to LTI systems. After, Sections 3.2.1 and 3.2.3 present nonlinear set-membership observers based on either zonotopes or CZs to incorporate state constraints.
- Chapter 4 introduces novel mixed-uncertainty state estimators for the linear case. Section 4.1 presents a filter based on zonotopes and minimum-variance performance criterion to extend a linear predictor revisited in Appendix C. In Section 4.2, a filter based on CZs is introduced to treat LTV descriptor systems with state linear inequality constraints, which exploits additional design matrices (degrees of freedom) and introduces a Pareto analysis to solve biobjective optimization problems using cost normalization. In Section 4.3, an LTI filter based on CZs and specific Pareto analysis is proposed to estimate states from LTI descriptor systems, with the main motivation being the low-cost computation.

- Chapter 5 introduces novel mixed-uncertainty state estimators for the nonlinear case. Specifically, Section 5.1 presents a filter based on zonotopes, quasi-LPV approximation, and minimum-variance performance criterion to extend a nonlinear predictor revisited in Appendix C. After, Section 5.2 introduces a filter based on CZs, Pareto analysis with cost normalization, and minimum-variance criterion to estimate states from nonlinear state-space systems with state nonlinear invariants and linear inequality constraints.
- Chapter 6 concludes the thesis and addresses several continuity proposals to the mixed-uncertainty case. These proposals are related to the theoretical advances reached in Chapters 3-5.
- In Appendices A-C, a review of stochastic, set-based, and mixed-uncertainty methods is presented. Appendix D discusses some practical issues about implementing state estimators under continuous-time nonlinear dynamical systems with sampled measurement data. This topic is long, being then separate from the main text. Ultimately, all proofs of worst-case complexity informed over the thesis are proved in Appendix E.

2

Prolegomena

This chapter aims at introducing preliminary concepts and results. Here, we focus on continuous random vectors, and on convex and compact sets to characterize uncertainties. In order to clarify the key results, we discuss illustrative examples.

2.1 Random Vectors

2.1.1 Definitions

Definition 2.1.1 ([Nair \(2013\)](#)). Let Ω be a sample space and $\mathcal{S}$ a measurable set. An *uncertain variable* x is a measurable function that maps Ω into $\mathcal{S}$ based on its range $[x] \subset \mathcal{S}$, that is, $[x] = \{x(\omega) : \omega \in \Omega\}$.

Definition 2.1.2 ([Kay \(1993\)](#)). Let Ω be a sample space and $\mathcal{S}$ a measurable set. A *random variable* x is a mapping from the sample space Ω into the measurable set $\mathcal{S}$ based on a probability distribution. If $\mathcal{S} \subset \mathbb{N}$, then the random variable is said to be discrete. If $\mathcal{S} \subset \mathbb{R}$, then the random variable is said to be continuous.

A continuous n -dimensional random vector X is described by a probability density function (PDF) $p(x)$, where x is a realization of X . The mean $\bar{x} \in \mathbb{R}^n$ and the covariance matrix $P^{xx} \in \mathbb{R}^{n \times n}$ of X , associated to the first two moments of the random vector X , are defined as $\bar{x} = E[X]$ and $P^{xx} = \text{cov}(X, X) \triangleq E[(X - \bar{x})(X - \bar{x})^\top]$.

From the mean $\bar{x}$ and the covariance matrix P^{xx} , the PDF of the GRV X is fully defined and denoted as $X \sim \mathcal{N}(\bar{x}, P^{xx})$. Next, an example is proposed to illustrate a GRV.

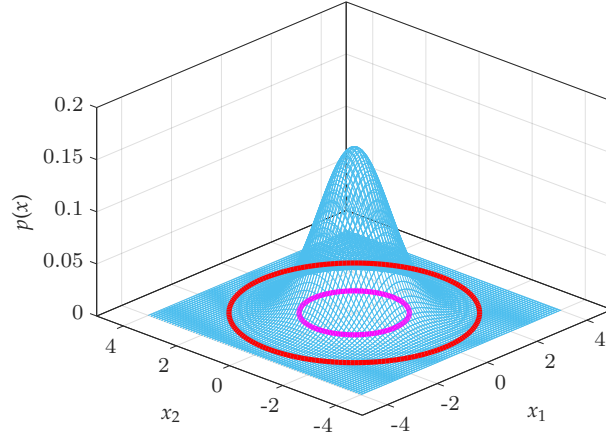


Figure 2.1: Illustration of the GRV $X \sim \mathcal{N}(0_{2 \times 1}, I_2)$ in 2D for $x_1, x_2 \in [-4, 4]$. When the probability related to each realization is also sketched, the bell shape is verified (cyan graph). The magenta and red ellipses correspond to the projections of the GRV X on the plane $x_1 O x_2$ for the confidence levels equal to 68.27% and 99.73%, respectively.

Example 2.1.1. Consider the GRV $X \sim \mathcal{N}(0_{2 \times 1}, I_2)$ illustrated in Figure 2.1. Note that the largest probabilities $p(x)$ concentrate around the mean $\bar{x} = 0_{2 \times 1}$. In order to verify a significant difference between the 68.27% confidence level and the usual 99.73% confidence level, we also sketch the magenta and red ellipses, respectively.

Given a GRV $X \sim \mathcal{N}(\bar{x}, P^{xx})$ with covariance $P^{xx} > 0_{n \times n}$, a chi-square random variable X^χ is defined as

$$X^\chi \triangleq (X - \bar{x})^\top (P^{xx})^{-1} (X - \bar{x}), \quad (2.1)$$

where $(\cdot)^{-1}$ is the matrix inversion operator, and $x^\chi \geq 0$ is a realization of X^χ . For brevity, a chi-square random variable can be represented by its n degrees of freedom, that is, $X^\chi \sim \chi^2(n)$, where $E[X^\chi] = n$ and $\text{cov}(X^\chi, X^\chi) = 2n$.

The cumulative distribution function $c(x)$ of a given PDF $p(x)$ is defined as

$$c(x) \triangleq \int_{-\infty}^{x_1} \cdots \int_{-\infty}^{x_n} p(\theta_1, \dots, \theta_n) d\theta_1 \cdots d\theta_n, \quad (2.2)$$

with $0 \leq c(x) \leq 1$, for all $x \in \mathbb{R}^n$.

Definition 2.1.3 (Kay (1993)). Without loss of generality, a *discrete-time stochastic process* is defined as an infinite sequence of random vectors $\{X\} = \{X_0, X_1, \dots, X_k, \dots\}$, with one random vector X_k for each time instant $k \in \mathbb{N}$, and each realization of the stochastic process takes on a value that is represented as an infinite sequence of numbers, that is, $\{x\} = \{x_0, x_1, \dots, x_k, \dots\}$.

A stochastic process is called *stationary* if its probabilistic description does not change with respect to the time origin. A stochastic process is called *wide-sense stationary* if: (i) its mean $E[X_k]$ is constant over time k , and (ii) its covariance $\text{cov}(X_{k_1}, X_{k_2})$ depends only upon the time difference, that is, $k_2 - k_1$.

Here, a *white noise* is defined as a wide-sense stationary stochastic process with: (i) zero mean $E[X_k] = 0$, (ii) constant covariance $\text{cov}(X_k, X_{k+\tau}) = Q$ for $\tau = 0$, and (iii) uncorrelated samples $\text{cov}(X_k, X_{k+\tau}) = 0_{n \times n}$ for $\tau \neq 0$, where $0_{n \times n}$ represents the null matrix.

The power spectral density quantifies the distribution of power of a given stochastic process over the frequency. Here, we are interested in discrete-time signals. Then, the power-spectral-density matrix of a *zero-mean* wide-sense stationary stochastic process $\{W\}$ is given by

$$S^w(\omega) \triangleq \sum_{\tau=-\infty}^{+\infty} Q_\tau e^{-j\omega\tau}, \quad (2.3)$$

where $Q_\tau = \text{cov}(W_\tau, W_0)$ is the autocorrelation matrix, $\omega \in [-\pi, \pi]$ is the angular frequency, and $k = 0$ is the time origin of the SP.

If the stochastic process $\{W\}$ is a white noise with $Q = I_b$, where I_b is the $(b \times b)$ -dimensional identity matrix, then, its power spectral density is $S^w(\omega) = I_b$. That is, the power spectral density of the white noise is constant over its frequency spectrum.

2.1.2 Operations

An appealing property of GRVs is to keep the Gaussianity under affine transformations.

Property 2.1.1 (Affine transformation of GRVs ([Särkkä, 2013](#))). Let $X \sim \mathcal{N}(\bar{x}, P^{xx})$ be a GRV with mean $\bar{x} \in \mathbb{R}^n$ and covariance $P^{xx} \in \mathbb{R}^{n \times n}$. Applying the affine transformation $y = Lx + b$, with $L \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$, yields the GRV $Y \sim \mathcal{N}(\bar{y}, P^{yy})$, where

$$\begin{aligned} \bar{y} &= L\bar{x} + b, \\ P^{yy} &= LP^{xx}L^\top, \\ P^{xy} &= P^{xx}L^\top, \end{aligned}$$

with $P^{xy} = \text{cov}(X, Y)$ being the cross correlation between the GRVs X and Y .

If the GRVs are uncorrelated, then their sum is readily computed.

Property 2.1.2 (Sum of uncorrelated GRVs ([Särkkä, 2013](#))). Let $X_1 \sim \mathcal{N}(\bar{x}_1, P^{x_1x_1})$ and $X_2 \sim \mathcal{N}(\bar{x}_2, P^{x_2x_2})$ be uncorrelated GRVs. Then, adding X_1 to X_2 yields another GRV X_3 given by $X_3 = X_1 + X_2 \sim \mathcal{N}(\bar{x}_1 + \bar{x}_2, P^{x_1x_1} + P^{x_2x_2})$.

GRVs can also be augmented using Cartesian product.

Property 2.1.3 (Cartesian product of GRVs). Let $X_1 \sim \mathcal{N}(\bar{x}_1, P^{x_1x_1})$ and $X_2 \sim \mathcal{N}(\bar{x}_2, P^{x_2x_2})$ be GRVs with $\bar{x}_1 \in \mathbb{R}^n$, $\bar{x}_2 \in \mathbb{R}^m$, $P^{x_1x_1} \in \mathbb{R}^{n \times n}$, and $P^{x_2x_2} \in \mathbb{R}^{m \times m}$. Then, the Cartesian product between X_1 and X_2 yields the augmented GRV $X_3 \sim \mathcal{N}(\bar{x}_3, P^{x_3x_3})$ such that

$$\bar{x}_3 = \begin{bmatrix} \bar{x}_1 \\ \bar{x}_2 \end{bmatrix}, \quad P^{x_3x_3} = \begin{bmatrix} P^{x_1x_1} & 0_{n \times m} \\ 0_{m \times n} & P^{x_2x_2} \end{bmatrix}.$$

The following theorem allows to overestimate the product between a covariance matrix M and a parameter $\zeta \in [-1, 1]$ by an upper bounded covariance matrix $\bar{M}$.

Theorem 2.1.1 (Combastel (2016)). Let $M \in \mathbb{R}^{n \times n}$ be a symmetric matrix and $M = V\Lambda V^{-1}$ be its spectral decomposition with $V^T V = I_n$ and real diagonal Λ . Let $\bar{M} = V|\Lambda|V^T$. Then, $\bar{M} \geq 0_{n \times n}$ and, $\forall \zeta \in [-1, 1]$, $\zeta M \leq \bar{M}$.

As GRVs are not closed under nonlinear transformations, some procedures can be applied to approximate the transformed random vector by a GRV. In Gustafsson & Hendebly (2011) and de Paula (2018), some well-known strategies are revisited: truncated Taylor expansion and UT. Due to the fact that UT mitigates some drawbacks of the other approach, it is revisited in Subsection A.2.1.1.

2.2 Sets

Next, we define some types of sets and usual operations.

2.2.1 Definitions

Definition 2.2.1 (Interval (Jaulin & Kieffer, 2001)). An interval $[x] \triangleq [x^L, x^U]$ is a one-dimensional set given by $\{x \in \mathbb{R} : x^L \leq x \leq x^U\}$.

Definition 2.2.2 (Unitary interval (Jaulin & Kieffer, 2001)). A unitary interval is denoted as $\mathcal{B} \triangleq [-1, 1]$.

Definition 2.2.3 (Box (Jaulin & Kieffer, 2001)). A box is an n -dimensional interval vector defined as $[x] \triangleq \{x \in \mathbb{R}^n : x_i^L \leq x_i \leq x_i^U, i = 1, \dots, n\}$, with x_i^L and x_i^U being the known lower and upper bounds.

Definition 2.2.4 (Unitary box (Jaulin & Kieffer, 2001)). A unitary box composed of n unitary intervals is denoted as $\mathcal{B}^n$.

Definition 2.2.5 (Constrained unitary box (Scott et al., 2016)). A constrained unitary box is defined as

$$\mathcal{B}(A^x, b^x) \triangleq \{\xi : \xi \in \mathcal{B}^{n_g}, A^x \xi = b^x\}, \quad (2.4)$$

where $A^x \in \mathbb{R}^{n_h \times n_g}$ and $b^x \in \mathbb{R}^{n_h}$ represent equality linear constraints on the unitary box $\mathcal{B}^{n_g}$.

Definition 2.2.6 (Interval matrix (Jaulin & Kieffer, 2001)). An interval matrix $[A]$ is the set defined as

$$[A] \triangleq \{A \in \mathbb{R}^{m \times n} : A_{i,j}^L \leq A_{i,j} \leq A_{i,j}^U, i = 1, \dots, m, j = 1, \dots, n\}. \quad (2.5)$$

Definition 2.2.7 (Strip (Vicino & Zappa, 1996)). A strip $\mathcal{S} \subset \mathbb{R}^n$ is defined as

$$\mathcal{S} \triangleq \{x \in \mathbb{R}^n : |c^\top x - d| \leq \rho\}, \quad (2.6)$$

where $\rho > 0$, $d \in \mathbb{R}$, and $c \in \mathbb{R}^n$ represent the width, shift, and direction of the strip $\mathcal{S}$, respectively. Moreover, $v_+ = \{x \in \mathbb{R}^n : c^\top x = d + \rho\}$ and $v_- = \{x \in \mathbb{R}^n : c^\top x = d - \rho\}$ are the parallel hyperplanes that bound $\mathcal{S}$.

Definition 2.2.8 (Minkowski sum (Kühn, 1998)). The Minkowski sum of two sets $\mathcal{X}$ and $\mathcal{Y}$ (represented by $\oplus$) is defined as

$$\mathcal{X} \oplus \mathcal{Y} \triangleq \{(x + y) : x \in \mathcal{X}, y \in \mathcal{Y}\}.$$

Next, the set *zonotope* is defined as an affine transformation of the unitary box $\mathcal{B}^{n_g}$.

Definition 2.2.9 (Zonotope (Kühn, 1998)). Let $G^x \in \mathbb{R}^{n \times n_g}$ and $\bar{x} \in \mathbb{R}^n$ be the generator matrix and the center, respectively. The zonotope $\mathcal{X}$ of order n_g is defined as

$$\mathcal{X} \triangleq \{G^x, \bar{x}\} = G^x \mathcal{B}^{n_g} \oplus \bar{x} = \{(G^x \xi + \bar{x}) : \xi \in \mathcal{B}^{n_g}\}, \quad (2.7)$$

where $\xi \in \mathbb{R}^{n_g}$ is an element of the unitary box $\mathcal{B}^{n_g}$.

If the generator matrix G^x is square, then the zonotope $\mathcal{X}$ can represent two special sets: *box* (diagonal generator matrix) and *parallelotope* (invertible generator matrix). The generator matrix of a zonotope $\mathcal{X}$ can also be an interval matrix as given by (2.5). In this case, Definition 2.2.9 is extended as follows.

Definition 2.2.10 (Family of zonotopes (Alamo et al., 2005)). Let $[G^x] \in \mathbb{R}^{n \times n_g}$ and $\bar{x} \in \mathbb{R}^n$ be the interval generator matrix and the center, respectively. The family of zonotopes $[\mathcal{X}]$ of order n_g is defined as

$$[\mathcal{X}] = [G^x] \mathcal{B}^{n_g} \oplus \bar{x} \triangleq \{(G^x \xi + \bar{x}) : G^x \in [G^x], \xi \in \mathcal{B}^{n_g}\},$$

where $\xi \in \mathbb{R}^{n_g}$ is an element of the unitary box $\mathcal{B}^{n_g}$.

Next, the set *constrained zonotope* (CZ) $\mathcal{X}$ is defined as the affine transformation of the constrained unitary box $\mathcal{B}(A^x, b^x)$.

Definition 2.2.11 (Constrained zonotope (Scott et al., 2016)). Consider that the linear equality constraints A^x and b^x are enforced on the unitary box $\mathcal{B}^{n_g}$ in Definition 2.2.9. Thereby, zonotopes are extended to CZs as

$$\mathcal{X} \triangleq \{G^x, \bar{x}, A^x, b^x\} = G^x \mathcal{B}(A^x, b^x) \oplus \bar{x}. \quad (2.8)$$

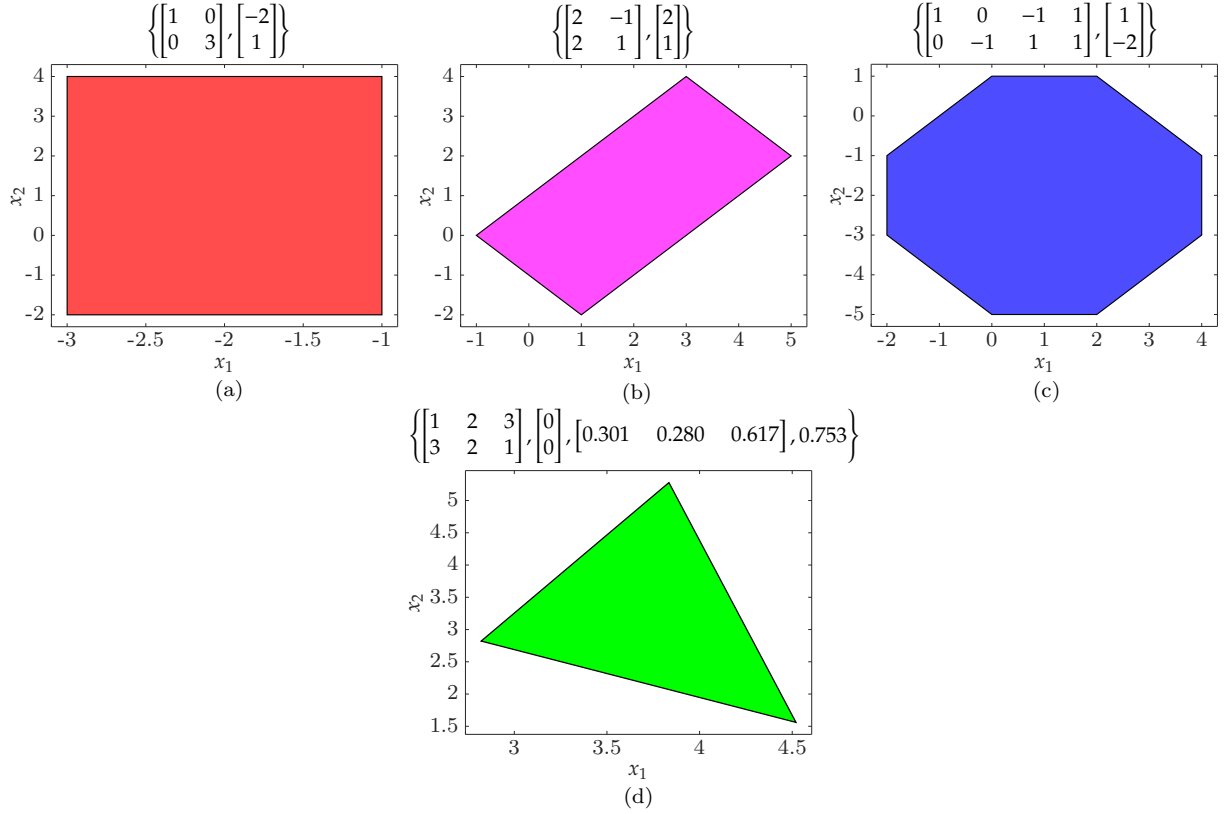


Figure 2.2: Illustrating convex polytopes in 2D. (a) box, (b) parallelogram, (c) centrally symmetric convex polytope, and (d) asymmetric convex polytope.

Thereby, a zonotope is extended to a CZ. Zonotopes are special cases of CZs for which the constraints given by A^* and b^* are empty arguments, that is, no equality constraint is considered on the unitary box $\mathcal{B}^{n_g}$. As $\mathcal{B}(A^*, b^*) \subset \mathcal{B}^{n_g}$, using CZs rather than zonotopes over operations leads to smaller sets. Next, zonotope and CZ are illustrated in Example 2.2.1.

Example 2.2.1. Three types of zonotopes and a CZ are sketched in Figure 2.2. A box is generated in (a) by employing Definition 2.2.9 with generator matrix being diagonal. In comparison with Definition 2.2.3, a box does have dependencies among its axis, justifying its rectangular geometry. A parallelogram is obtained in (b) by employing Definition 2.2.9 with generator matrix being invertible. This set reminds a tilted box because there are dependencies among its axis. A zonotope is illustrated in (c) with the most general form of Definition 2.2.9. Definition 2.2.11 allows for adding equality constraints to the unitary hypercube of zonotopes, yielding a CZ as in (d).

2.2.2 The Dependency and Wrapping Effects

When operations are performed among sets, two effects can arise, namely *dependency effect* and *wrapping effect* (Kühn, 1998; Jaulin & Kieffer, 2001; Moore et al., 2009). Both effects characterize undesired overestimations. Let $y = h(x)$ be a nonlinear transformation,

where x is an element of the convex set $\mathcal{X}$; this set can be an ellipsoid, box, parallelotope, zonotope, or polytope, for instance. We are interested in computing an enclosure of the range of a convex set $\mathcal{X}$ over h . This enclosure must belong to the same class of sets of $\mathcal{X}$. The exact set is built by transforming each element x from $\mathcal{X}$ over h , where the standard operations (or among elements) are performed. Due to the nonlinearity, the actual set $h(\mathcal{X})$ may be nonconvex, even if $\mathcal{X}$ is convex. In order to approximate the exact set, a significantly large set of points (grid) of $\mathcal{X}$ could be elementwise transformed, but this procedure may be costly and/or impracticable. To feasibly reach a convex set $\mathcal{Y}$ with the same class of $\mathcal{X}$, there exists a series of definitions and results that guarantees the inclusion $h(\mathcal{X}) \subseteq \mathcal{Y}$ (Kühn, 1998; Jaulin & Kieffer, 2001; Moore et al., 2009; Le et al., 2013b). The resulting set $\mathcal{Y}$ may be larger than the exact one $h(\mathcal{X})$ because the former is convex while the latter may be nonconvex. Moreover, $\mathcal{Y}$ may not be the smallest possible outer approximation of $h(\mathcal{X})$ due to the existence of the dependency and wrapping effects.

The dependency effect is the phenomenon responsible by the conservatism of set-valued operations caused by the multioccurrence of variables (Jaulin & Kieffer, 2001). Intuitively, the multioccurrence does not affect the exact result for elementwise operations. However, for operations with sets, we should write a transformation with the smallest possible amount of repeated variables, since the results may be different. This is the reason why interval operations are not closed under additive inverse, multiplicative inverse, or subdistributivity, for instance, except for degenerated intervals (Moore et al., 2009).

In turn, the wrapping effect is a phenomenon that yields excess of conservatism through the inclusion of extra points (Moore et al., 2009). Some examples of wrapping effect appear in the outer approximation of: (i) nonconvex sets by convex sets, (ii) intersection among zonotopes, (iii) high-complexity sets by low-complexity sets via order-reduction algorithms, and (iv) bounded sets by unbounded sets (as strips representing bounded measurements). Both dependency and wrapping effects have been previously illustrated in our paper de Paula et al. (2022c, Sidebar: Understanding the Dependency and Wrapping Effects).

2.2.3 Interval-based Operations

Next, some set-valued operations are reviewed.

Definition 2.2.12 (Interval operators (Jaulin & Kieffer, 2001)). Given an interval matrix $[A]$, its (i, j) th infimum, its (i, j) th supremum, its (i, j) th midpoint, its (i, j) th diameter, and its (i, j) th radius are, respectively, defined as

$$\inf_{i,j}([A]) \triangleq A_{i,j}^L, \quad (2.9)$$

$$\sup_{i,j}([A]) \triangleq A_{i,j}^U, \quad (2.10)$$

$$\text{mid}_{i,j}([A]) \triangleq \frac{A_{i,j}^L + A_{i,j}^U}{2}, \quad (2.11)$$

$$\text{diam}_{i,j}([A]) \triangleq A_{i,j}^U - A_{i,j}^L, \quad (2.12)$$

$$\text{rad}_{i,j}([A]) \triangleq \frac{\text{diam}_{i,j}([A])}{2}, \quad (2.13)$$

for $i = 1, \dots, m$, and $j = 1, \dots, n$.

Definition 2.2.13 (Interval sum, difference, and product (Jaulin & Kieffer, 2001)). The sum, difference, and product of the boxes $[x] = [x^L, x^U]$ and $[y] = [y^L, y^U]$ are computed as

$$[x] + [y] \triangleq [x^L + y^L, x^U + y^U], \quad (2.14)$$

$$[x] - [y] \triangleq [x^L - y^U, x^U - y^L], \quad (2.15)$$

$$[x][y] \triangleq [\min\{\mathbf{S}\}, \max\{\mathbf{S}\}], \quad (2.16)$$

where $\mathbf{S} = \{x^L y^L, x^L y^U, x^U y^L, x^U y^U\}$.

Definition 2.2.14 (Interval intersection (Jaulin & Kieffer, 2001)). Given the boxes $[x], [y] \subset \mathbb{R}^n$, their intersection $[z] = [x] \cap [y]$ is *exactly* defined as

$$[z]_i \triangleq [\max\{x_i^L, y_i^L\}, \min\{x_i^U, y_i^U\}], \quad i = 1, \dots, n. \quad (2.17)$$

Consider a nonlinear transformation $y = h(x)$. The function h is called *standard* when its input x is an element (real vector); in this case, y is also an element. If the element x is replaced by a set $\mathcal{X}$, then the transformation $h(\mathcal{X})$ also yields a set. In terms of elements, the exact image $\mathcal{Y}$ is built by applying elementary operations to individually all elements $x \in \mathcal{X}$. Often, nonlinear operations on a convex set yield a nonconvex set.

Let $\mathcal{X} = [x]$ be a box. If interval arithmetic operations are applied to the transformation $\mathcal{Y} = h(\mathcal{X})$ rather than elementary operations, then the function h is replaced by the *natural interval extension* $\square h$, which yields a box $[y] = \square h([x])$ (Alamo et al., 2005; Moore et al., 2009). According to the fundamental theorem of interval arithmetic (Moore, 1966), the following inclusion is satisfied $h([x]) \subseteq \square h([x])$.

Definition 2.2.15 (Natural interval extension (Alamo et al., 2005)). Let $y = h(x)$ be a nonlinear transformation. Given the box $[x]$, the natural interval extension $\square h$ is obtained substituting x by $[x]$, and all standard operations by the corresponding interval operations.

When it comes to evaluating boxes $[x]$ over monotonic functions f , or nonmonotonic functions but with monotonic behavior under $[x]$, the natural interval extension consists of individually evaluating the bounds x^L and x^U over h (Moore et al., 2009). In this case, the endpoints are ordered according to the increasing ($\square h = [h(x^L), h(x^U)]$) or decreasing ($\square h = [h(x^U), h(x^L)]$) characteristic of f . This is the case of functions such as $\exp(\cdot)$, $\log(\cdot)$, $\sin(\cdot)$, $\cos(\cdot)$, $\tan(\cdot)$, $\cot(\cdot)$, and $\sqrt{\cdot}$.

Let $[x] = [x, x] = x$ be a *degenerated box*. Then, the natural interval extension $\square h$ is reduced to the standard function h .

Remark 2.2.1. Note that the natural interval extension $\square h$ works on boxes. Therefore, any set $\mathcal{X}$ must be approximated by a box $[x]$ before applying the natural interval extension. Commonly, the box $[x]$ is defined as an outer approximation of $\mathcal{X}$, in order to guarantee the inclusion property (Alamo et al., 2005). This interval overapproximation is called *interval enclosure*. If the interval enclosure of $\mathcal{X}$ is the smallest box containing $\mathcal{X}$, then it is called *interval hull*, here denoted as $\square\mathcal{X}$.

Given a series of sets $\{\mathcal{X}_1, \dots, \mathcal{X}_n\}$, we have the following property related to its interval hull.

Property 2.2.1 (Tang et al. (2019a)). For a series of sets $\mathcal{Z}_i$, $i = 1, \dots, n$, we have the following equivalence:

$$\square\left(\bigoplus_{i=1}^n \mathcal{Z}_i\right) = \bigoplus_{i=1}^n \square\mathcal{Z}_i,$$

where

$$\bigoplus_{i=1}^n \mathcal{Z}_i \triangleq \mathcal{Z}_1 \oplus \dots \oplus \mathcal{Z}_n.$$

To perform state estimation, we are interested in operating convex sets. A challenge is to obtain the smallest possible overestimation for the actual set over operations. However, due to the dependency and wrapping effects, the operations among sets may return larger sets; see Subsection 2.2.2 for more details. Since these effects commonly appear due to interval arithmetic operations, they are illustrated next.

Example 2.2.2 (Moore et al. (2009)). Consider the transformation $y = x - x$, where $x \in \mathbb{R}$ is an element of the interval $[x] = [x^L, x^U]$, with $x^L \neq x^U$. In terms of elementary operations, $y = 0$ for all $x \in [x]$. However, applying the natural interval extension (Definition 2.2.15), where all standard operations are replaced by interval operations, we obtain the symmetric and nondegenerated interval

$$[y] = [x^L - x^U, x^U - x^L] = (x^U - x^L)[-1, 1].$$

This example illustrates the dependency effect in one dimension.

Example 2.2.3 (Jaulin & Kieffer (2001)). Consider the transformation $y = (\text{sign}(x))^2$, where $\text{sign}(x) = 1$ if $x \geq 0$, or $\text{sign}(x) = -1$ otherwise. In terms of standard operations, $y = 1$ for all $x \in \mathbb{R}$. However, given $\text{sign}(x) \in [-1, 1]$, the natural interval extension (Definition 2.2.15) leads to $[y] = [-1, 1]^2 = [0, 1]$. This example illustrates the wrapping effect in one dimension.

Example 2.2.4. Consider the zonotope $\mathcal{X} = \left\{ \begin{bmatrix} 1 & -2 \\ 1 & 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \end{bmatrix} \right\} \subset \mathbb{R}^2$ and its interval hull $\square\mathcal{X}$. In Figure 2.3, the zonotope $\mathcal{X}$ (red solid line) and the box $\square\mathcal{X}$ (blue dashed line) are sketched. Moreover, a thin grid of $\mathcal{X}$ with 40,000 random points (black points) is illustrated. Let

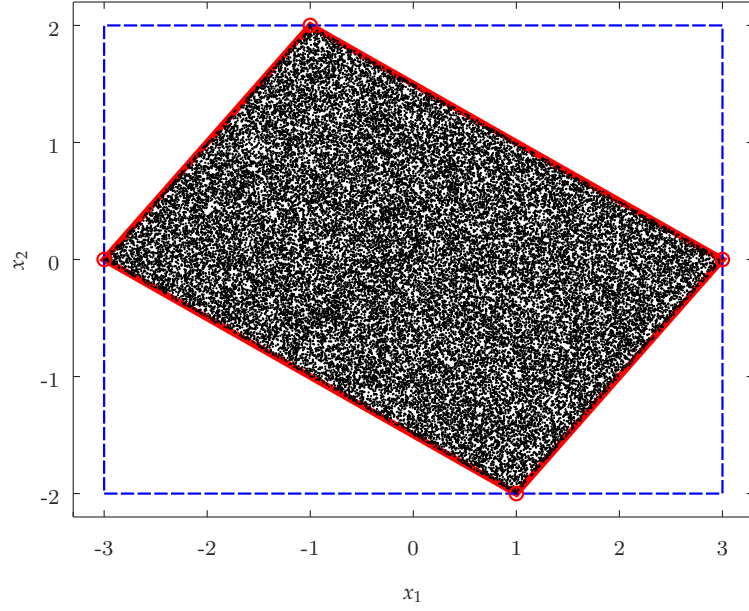


Figure 2.3: Illustrative example of the wrapping effect for functions whose input is multivariate and output is monovariate. The goal is to transform the zonotope $\mathcal{X}$ (—) over the transformations $h_1(x) = x_1 + x_2$ and $h_2(x) = x_1 x_2$. To approximate the actual interval, a grid of $\mathcal{X}$ is provided with 40,000 random points (.), including the extreme points (o). Applying standard operations to these points, the intervals $h_1(\mathcal{X}) = [-3, 3]$ and $h_2(\mathcal{X}) \approx [-2.250, 1.125]$ are yielded. To make this procedure less costly, the natural interval extension can be applied to $h_1(\mathcal{X})$ and $h_2(\mathcal{X})$. In doing so, the intervals $h_1(\square\mathcal{X}) = [-5, 5]$ and $h_2(\square\mathcal{X}) = [-6, 6]$ are obtained, where $\square\mathcal{X}$ (--) is the interval hull of $\mathcal{X}$.

$h_1(x) = x_1 + x_2$ and $h_2(x) = x_1 x_2$ be the desired transformations. Applying standard operations to the grid of points, we reach $h_1(\mathcal{X}) = [-3, 3]$ and $h_2(\mathcal{X}) \approx [-2.250, 1.125]$. On the other hand, applying the sum and product operations between intervals, given by (2.14) and (2.16) respectively, we obtain $h_1(\square\mathcal{X}) = [-5, 5]$ and $h_2(\square\mathcal{X}) = [-6, 6]$. This example illustrates the wrapping effect for functions whose input is multivariate and output is monovariate. Specifically, $\mathcal{X}$ is outer-approximated by its interval hull.

Example 2.2.5. Consider the functions $h_1(x_1, x_2) = \begin{bmatrix} (x_1 - x_2)^2 \\ x_1 x_2 \end{bmatrix}$ and $h_2(x_1, x_2) = \begin{bmatrix} (x_1 - x_2)(x_1 + x_2) \\ x_1 x_2 \end{bmatrix}$, where $x_1 \in [x_1] = [1, 3]$ and $x_2 \in [x_2] = [2, 9]$. Assume a thin grid is provided for the intervals $[x_1]$ and $[x_2]$. Transforming each point of the grid over h_1 or h_2 leads to the actual set $h = h_1 = h_2$. Now, assume the natural interval extension (Definition 2.2.15) is applied to both h_1 and h_2 . Note that the difference between h_1 and h_2 is that h_2 has repeated variables over its first component, therefore being subject to the dependency effect. In Figure 2.4, the red points illustrate that the exact image is not convex. Therefore, both h_1 and h_2 are subject to the wrapping effect, since interval arithmetic returns boxes as outputs. On one hand, h_1 yields the smallest box containing the actual set; it occurs because the function h_1 and the corresponding operators are continuous (Jaulin & Kieffer, 2001). On the other hand, h_2 yields a larger box than h_1 does, due to the dependency effect.

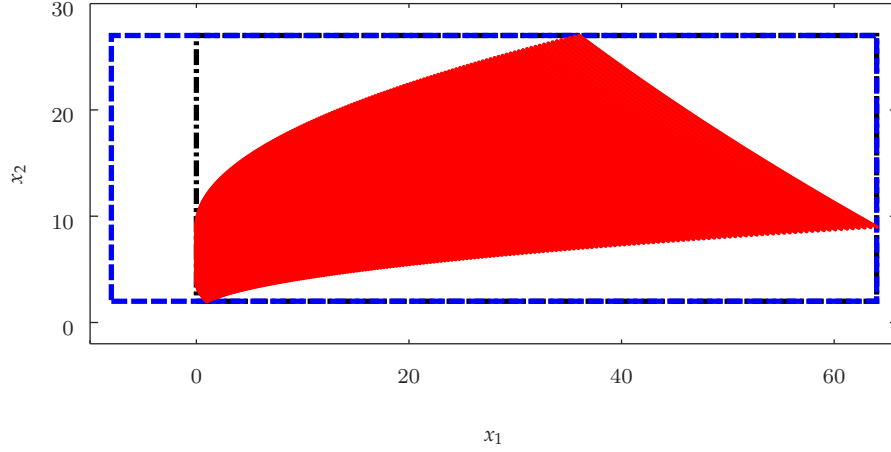


Figure 2.4: Illustrative example of the wrapping and dependency effects in $\mathbb{R}^2$. The equivalent functions $h_1 = [(x_1 - x_2)^2 \ x_1 x_2]^\top$ and $h_2 = [(x_1 - x_2)(x_1 - x_2) \ x_1 x_2]^\top$ are considered to transform the intervals $[x_1] = [1, 3]$ and $[x_2] = [2, 9]$. Completing a grid of 40,000 points, and transforming it over h_1 and h_2 , the exact image $h = h_1 = h_2$ (red dots) is generated. By using the intervals $[x_1]$ and $[x_2]$ instead of the grid and applying interval arithmetic operations, the natural interval extensions $\square h_1$ (black dash-dotted line) and $\square h_2$ (blue dashed line) generate boxes as images.

The direct application of interval arithmetic may be undesirable in some cases due to the resulting conservatism. Instead, we can apply the *mean value extension* over operations, since it may yield better precision than or equal to the natural interval extension (Alamo et al., 2005).

Proposition 2.2.1 (Mean value extension (Alamo et al., 2005)). Let $h : \mathbb{R}^n \rightarrow \mathbb{R}^m$ be a nonlinear function whose first-order derivatives are continuous in $\mathcal{X} \subset \mathcal{I}$, where $\mathcal{I} \subset \mathbb{R}^n$ represents a bounded box containing some set $\mathcal{X}$, and consider some real vector $x \in \mathcal{X}$. Then,

$$h(\mathcal{X}) \subseteq h(x) \oplus \square \{\nabla_x h(\mathcal{X})\} (\mathcal{X} - x), \quad (2.18)$$

where $\square \{\nabla_x h(\mathcal{X})\}$ is an interval enclosure of $\nabla_x h(\mathcal{X})$.

Originally, the term $\square \{\nabla_x h(\mathcal{X})\}$ is treated as an interval enclosure (Alamo et al., 2005). Here, we apply the natural interval extension, given by Definition 2.2.15, to compute such an enclosure. When employing Definition 2.2.15, interval arithmetic is closed under intersection and nonlinear mapping. However, if the input sets are not originally interval and if there exist the wrapping and dependency effects, then the obtained boxes may not be the smallest possible ones.

2.2.4 Operations with Zonotopes

Next, the three standard properties of zonotopes are presented.

Property 2.2.2 (Kühn (1998)). Given the zonotope $\mathcal{X} = \{G^x, \bar{x}\} \subset \mathbb{R}^n$, the affine transformation $\mathcal{Y} = L\mathcal{X} \oplus b$, where $L \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$, yields the transformed zonotope

$$\mathcal{Y} = \{LG^x, (L\bar{x} + b)\}. \quad (2.19)$$

Property 2.2.3 (Kühn (1998)). The Minkowski sum of two zonotopes $\mathcal{X} = \{G^x, \bar{x}\}$ and $\mathcal{W} = \{G^w, \bar{w}\}$ is given by

$$\mathcal{X} \oplus \mathcal{W} = \left\{ \begin{bmatrix} G^x & G^w \end{bmatrix}, (\bar{x} + \bar{w}) \right\}. \quad (2.20)$$

Property 2.2.4 (Combastel (2005)). Given the zonotopes $\mathcal{X} = \{G^x, \bar{x}\} \subset \mathbb{R}^n$ and $\mathcal{W} = \{G^w, \bar{w}\} \subset \mathbb{R}^m$, the Cartesian product between them is given by

$$\mathcal{X} \times \mathcal{W} = \left\{ \begin{bmatrix} G^x & 0_{n \times n_g^w} \\ 0_{n_g^w \times n_g} & G^w \end{bmatrix}, \begin{bmatrix} \bar{x} \\ \bar{w} \end{bmatrix} \right\}. \quad (2.21)$$

Analogously to Definition 2.2.15, the next result allows to compute a *zonotope inclusion* through the generalization of Kühn's method (Kühn, 1998). As the interval enclosure, the zonotope inclusion is a procedure that yields larger output sets than the input sets. However, the zonotope inclusion yields a zonotope, which is more accurate than a box.

Proposition 2.2.2 (Zonotope inclusion (Alamo et al., 2005)). Consider a family of zonotopes represented by $[\mathcal{X}] = [G^x] \mathcal{B}^{n_g} \oplus \bar{x}$, where $[G^x] \subset \mathbb{R}^{n \times n_g}$ is an interval generator matrix and $\bar{x} \in \mathbb{R}^n$ is the center. The zonotope inclusion $\diamond \mathcal{X} \supseteq [\mathcal{X}]$ is given by

$$\diamond \mathcal{X} \triangleq \left[\text{mid}([G^x]) \quad L \right] \begin{bmatrix} \mathcal{B}^{n_g} \\ \mathcal{B}^n \end{bmatrix} \oplus \bar{x}, \quad (2.22)$$

where $L \in \mathbb{R}^{n \times n}$ is a diagonal matrix, whose elements are computed from the intervals of $[G^x]$ as

$$L_{i,i} = \sum_{j=1}^{n_g} \text{rad}_{i,j}([G^x]), \quad i = 1, \dots, n. \quad (2.23)$$

The intersection between a zonotope $\mathcal{X} \subset \mathbb{R}^n$ and a strip $\mathcal{S} \subset \mathbb{R}^n$ is parametrized by a design vector $\kappa \in \mathbb{R}^n$ as follows.

Property 2.2.5 (Intersection (Alamo et al., 2005)). Given the zonotope $\mathcal{X} = \{G^x, \bar{x}\}$, the strip $\mathcal{S} = \{x \in \mathbb{R}^n : |c^\top x - d| \leq \rho\}$, and the design vector $\kappa \in \mathbb{R}^n$, define the zonotope $\check{\mathcal{X}} = \{\check{G}^x, \check{x}\}$, where

$$\check{G}^x = \begin{bmatrix} (I_n - \kappa c^\top) G^x & \rho \kappa \end{bmatrix}, \quad (2.24)$$

$$\check{x} = \bar{x} + \kappa (d - c^\top \bar{x}). \quad (2.25)$$

Then, $\mathcal{X} \cap \mathcal{S} \subseteq \check{\mathcal{X}}$.

Remark 2.2.2. The parameter κ is computed according to some performance criterion as the segment minimization (SM) and the volume minimization (VM). Although the SM looks like reducing the number of segments (generators), it corresponds to minimizing the Frobenius norm or *F-radius*; see Subsection B.2.2.1 for details about implementation, interpretation, advantages, and drawbacks of each algorithm.

Note that operations as Minkowski sum (Property 2.2.3), zonotope inclusion (Proposition 2.2.2) and intersection (Property 2.2.5) yield zonotopes with a larger number of generators. In iterative computations, such growth should be limited to reduce the computational demand; this procedure is known as *order reduction*. When employing zonotopic algorithms, we here use the zonotope order reduction (ZR) proposed by Combastel (2003); see Subsection B.1.1.1 for details. Let $\mathcal{X} \subset \mathbb{R}^n$ be a zonotope with order $n_g \geq n$. If ZR is applied to $\mathcal{X}$ to reduce its n_g generators to $\varphi = n$, then the zonotope $\mathcal{X}$ is outer-approximated by its interval hull $\square\mathcal{X} = [\zeta^L, \zeta^U] = [\bar{x} - r^z, \bar{x} + r^z]$, where

$$r^z \triangleq \text{rs}(G^x) = |G^x| 1_{n_g \times 1}. \quad (2.26)$$

Zonotopes are not closed under nonlinear transformations. Then, some procedure should be used to approximate the image set by a zonotope. In Alamo et al. (2005), zonotopes are propagated using the mean value extension (Proposition 2.2.1). This strategy is revisited in Subsection B.2.2.2. In Combastel (2005), zonotopes are transformed using the first-order Taylor extension.

2.2.5 Operations with Constrained Zonotopes

In this subsection, the standard properties of CZs are presented.

Property 2.2.6 (Scott et al. (2016)). Given the CZ $\mathcal{X} = \{G^x, \bar{x}, A^x, b^x\} \subset \mathbb{R}^n$, the affine transformation $\mathcal{Y} = L\mathcal{X} \oplus b$, where $L \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$, yields the transformed CZ

$$\mathcal{Y} = \{LG^x, (L\bar{x} + b), A^x, b^x\}. \quad (2.27)$$

Property 2.2.7 (Scott et al. (2016)). The Minkowski sum of two CZs $\mathcal{X} = \{G^x, \bar{x}, A^x, b^x\} \subset \mathbb{R}^n$ and $\mathcal{W} = \{G^w, \bar{w}, A^w, b^w\} \subset \mathbb{R}^n$ is given by

$$\mathcal{X} \oplus \mathcal{W} = \left\{ \begin{bmatrix} G^x & G^w \end{bmatrix}, (\bar{x} + \bar{w}), \begin{bmatrix} A^x & 0_{n_h \times n_g^w} \\ 0_{n_h^w \times n_g} & A^w \end{bmatrix}, \begin{bmatrix} b^x \\ b^w \end{bmatrix} \right\}. \quad (2.28)$$

Property 2.2.8 (Scott et al. (2016)). Given CZs $\mathcal{X} = \{G^x, \bar{x}, A^x, b^x\} \subset \mathbb{R}^n$ and $\mathcal{W} = \{G^w, \bar{w}, A^w, b^w\} \subset$

$\mathbb{R}^m$, the generalized intersection between them is given by

$$\mathcal{X} \cap_M \mathcal{W} = \left\{ \left[G^x \quad 0_{n \times n_g^w} \right], \bar{x}, \begin{bmatrix} A^x & 0_{n_h \times n_g^w} \\ 0_{n_h^w \times n_g} & A^w \\ MG^x & -G^w \end{bmatrix}, \begin{bmatrix} b^x \\ b^w \\ \bar{w} - M\bar{x} \end{bmatrix} \right\}, \quad (2.29)$$

where $M \in \mathbb{R}^{m \times n}$ is a matrix that relates the elements of $\mathcal{X}$ to the elements of $\mathcal{W}$.

Regarding Property 2.2.8, matrix M represents the correspondence between the different spaces. If $M = I_n$, then the direct intersection between sets is carried out. For the remaining cases, we are interested in extracting the information of $\mathcal{X}$ which is related to $\mathcal{Y}$ by means of the linear transformation $M\mathcal{X}$. We highlight that the intersection between zonotopes is not exact, since this operation may yield asymmetric polytopes.

Property 2.2.9 (Rego et al. (2020a)). Given the CZs $\mathcal{X} = \{G^x, \bar{x}, A^x, b^x\} \subset \mathbb{R}^n$ and $\mathcal{W} = \{G^w, \bar{w}, A^w, b^w\} \subset \mathbb{R}^m$, the Cartesian product between them is given by

$$\mathcal{X} \times \mathcal{W} = \left\{ \begin{bmatrix} G^x & 0_{n \times n_g^w} \\ 0_{n^w \times n_g} & G^w \end{bmatrix}, \begin{bmatrix} \bar{x} \\ \bar{w} \end{bmatrix}, \begin{bmatrix} A^x & 0_{n_h \times n_g^w} \\ 0_{n_h^w \times n_g} & A^w \end{bmatrix}, \begin{bmatrix} b^x \\ b^w \end{bmatrix} \right\}. \quad (2.30)$$

Additionally to zonotopes, CZs require the reduction of the number of constraints over iterative processes. The reduction algorithm for CZs is presented in Scott et al. (2016), and abbreviated here as

$$\mathcal{X}^\downarrow = \text{CZR}(\mathcal{X}, \varphi_c, \varphi_g), \quad (2.31)$$

where φ_c and φ_g are the desired maximum constraint and generator orders, respectively. We use *order* to denote the number of constraints and generators. CZR can be summarized in four steps: (i) rescaling; (ii) preconditioning; (iii) elimination of constraints and (partial) generators; and (iv) elimination of generators. The first three steps lead to reduction from n_h to φ_c and from n_g to $(n_g - \varphi_c)$, while the latter finishes the reduction from $(n_g - \varphi_c)$ to φ_g . The reduction algorithm for CZs requires another algorithm to reduce the number of generators. Whenever employing CZs, in this thesis we choose to reduce generators using the algorithm proposed in Scott et al. (2016).

In order to obtain the interval hull of a CZ, $\square\mathcal{X} = [\zeta^L, \zeta^U]$, such that $\mathcal{X} \subset \square\mathcal{X}$, we have the following proposition.

Proposition 2.2.3 (Rego et al. (2020a)). Let $\mathcal{X} = \{G^x, \bar{x}, A^x, b^x\} \subset \mathbb{R}^n$ and let $G_{i,:}^x$ denote the i th row of G^x . The interval hull $[\zeta^L, \zeta^U] \supseteq \mathcal{X}$ is obtained by solving LPs for each $i = 1, \dots, n$:

$$\zeta_i^L \triangleq \min_{\xi} \left\{ G_{i,:}^x \xi + \bar{x}_i : \xi \in \mathcal{B}(A^x, b^x) \right\}, \quad i = 1, \dots, n, \quad (2.32)$$

$$\zeta_i^U \triangleq \max_{\xi} \left\{ G_{i,:}^x \xi + \bar{x}_i : \xi \in \mathcal{B}(A^x, b^x) \right\}, \quad i = 1, \dots, n. \quad (2.33)$$

As zonotopes, CZs are not closed under nonlinear mappings. Some recent contributions

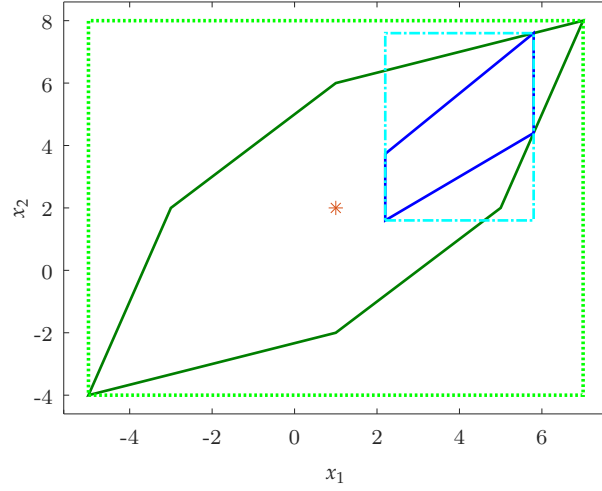


Figure 2.5: Illustrations of the 2D sets described in Example 2.2.6. The sketched sets are CZ (blue solid line) and zonotope (green solid line), besides their interval hulls in cyan dash-dotted line and green dotted line, respectively. All sets have the same center (*).

have been given for this topic in Rego et al. (2018b), Rego et al. (2020a), and Rego et al. (2021). The first work propagates CZs by converting to interval bundles, while the other ones propose both mean value extension (Alamo et al., 2005) and first-order Taylor extension (Combastel, 2005) for CZs.

Next, two examples are proposed to illustrate some properties and propositions related to boxes, zonotopes, and CZs.

Example 2.2.6. Consider the CZ $\mathcal{X} = \left\{ \begin{bmatrix} 1 & 2 & 3 \\ 3 & 2 & 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 2 \end{bmatrix}, [0.5 \ 0.1 \ 1.5], 1.5 \right\}$. This set is illustrated in Figure 2.5 together with its interval hull. A zonotope is also sketched by disregarding the linear equality constraints from the CZ $\mathcal{X}$. Note that the zonotope is centrally symmetric while the CZ is not. In this example, we illustrate the typical case of having a CZ whose center is outside its limits.

Example 2.2.7. This example aims at illustrating that intersection between zonotopes yields an asymmetric set, while the intersection between boxes is exact. Consider the zonotopes $\mathcal{X}_1 = \left\{ \begin{bmatrix} 1 & 2 & 3 \\ 3 & 2 & 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \end{bmatrix} \right\}$ and $\mathcal{X}_2 = \left\{ \begin{bmatrix} 1 & 0 & 3 \\ -3 & 2 & 1 \end{bmatrix}, \begin{bmatrix} 2 \\ 2 \end{bmatrix} \right\}$, which are equivalent to CZs with empty constraints. Using Property 2.2.8, the intersection between $\mathcal{X}_1$ and $\mathcal{X}_2$ yields

$$\mathcal{X}_3 = \left\{ \begin{bmatrix} 1 & 2 & 3 & 0 & 0 & 0 \\ 3 & 2 & 1 & 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 & 2 & 3 & -1 & 0 & -3 \\ 3 & 2 & 1 & 3 & -2 & -1 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \end{bmatrix} \right\}.$$

Note that $\mathcal{X}_3$ has equality constraints, then zonotopes cannot properly address intersections. Now, consider the boxes $[x_1] = \begin{bmatrix} [0, 2] \\ [-2, 2] \end{bmatrix}$ and $[x_2] = \begin{bmatrix} [-2, 2] \\ [0, 2] \end{bmatrix}$. Using Definition 2.2.14, the intersection between $[x_1]$ and $[x_2]$ yields the box $[x_3] = \begin{bmatrix} [0, 2] \\ [0, 2] \end{bmatrix}$. All these sets are

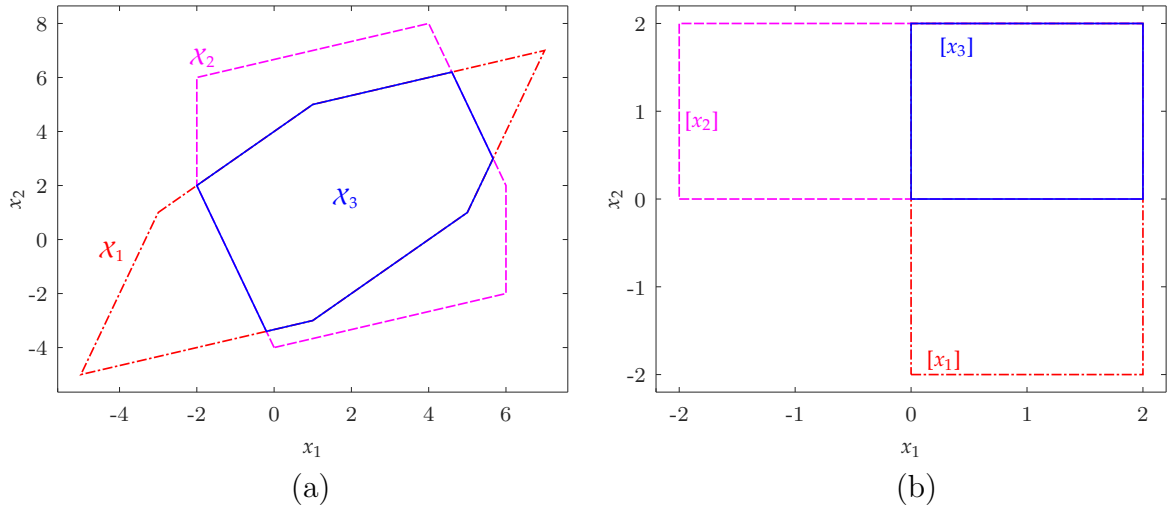


Figure 2.6: Exact intersections between the sets presented in Example 2.2.7. In graph (a), the intersection between the CZs X_1 and X_2 yields the CZ X_3 . In graph (b), the intersection between the boxes $[x_1]$ and $[x_2]$ yields the box $[x_3]$.

illustrated in Figure 2.6.

2.3 DC Programming

In this thesis, a DC programming approach is used to compute linearization enclosures. For an in-depth reading about DC programming, the reader is referred to Alamo et al. (2008), Tao & An (1997), and Horst & Thoai (1999). Next, we define a DC function.

Definition 2.3.1 (Alamo et al. (2008)). Consider a polytope $\mathcal{P} \subset \mathbb{R}^n$ and a function $\varrho : \mathbb{R}^n \rightarrow \mathbb{R}^m$. If ϱ can be rewritten as the difference between two convex functions ϱ^a and ϱ^b in $\mathcal{P}$, then, ϱ is called DC on $\mathcal{P}$.

The determination of the convex functions ϱ^a and ϱ^b may not be a trivial task. Some procedures to guide the choice of such functions are presented in Alamo et al. (2008). Next, we define the general form of a DC programming problem. Then, two results are presented to enclose the global solutions in intervals.

Definition 2.3.2 (Alamo et al. (2008)). Consider that the function $\varrho : \mathbb{R}^n \rightarrow \mathbb{R}^m$ is DC on $\mathcal{P} \subset \mathbb{R}^n$, with ϱ^a and ϱ^b being its DC components such that $\varrho(z) = \varrho^a(z) - \varrho^b(z)$. Then, for each component $i = 1, \dots, m$, the i th DC programming problems are formulated as

$$\min_{z \in \mathcal{P}} \varrho_i(z), \max_{z \in \mathcal{P}} \varrho_i(z). \quad (2.34)$$

Definition 2.3.3 (Alamo et al. (2008)). Let $\varrho(z) = (\varrho^a(z) - \varrho^b(z)) \in \mathbb{R}^m$ be DC on $\mathcal{P} \subset \mathbb{R}^n$. Then, the linear minorant of ϱ^s , with $s = \{a, b\}$, is defined as

$$\bar{\varrho}^s(z) \triangleq \varrho^s(\bar{z}) + F^s(z - \bar{z}), \quad (2.35)$$

where

$$F^s \triangleq \nabla_z \varrho^s(\bar{z}) = \left[\begin{array}{ccc} \frac{\partial \varrho_1^s}{\partial z_1} & \dots & \frac{\partial \varrho_1^s}{\partial z_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial \varrho_m^s}{\partial z_1} & \dots & \frac{\partial \varrho_m^s}{\partial z_n} \end{array} \right] \bigg|_{\bar{z}} \quad (2.36)$$

is the Jacobian matrix evaluated at some $\bar{z} \in \mathcal{P}$. The term *minorant* comes from the convexity of ϱ^s that implies the inequalities $\varrho^s(z) \geq \bar{\varrho}^s(z)$, $\forall z \in \mathcal{P}$.

Proposition 2.3.1 (Alamo et al. (2008)). Let $\varrho(z) = (\varrho^a(z) - \varrho^b(z)) \in \mathbb{R}^m$ be a DC function on the polytope $\mathcal{P} \subset \mathbb{R}^n$. Then, according to Definition 2.3.3, the following inequalities hold:

$$\min_{z \in \mathcal{P}} \varrho_i(z) \geq \min_{z \in \text{vert}(\mathcal{P})} \bar{\varrho}_i^a(z) - \varrho_i^b(z), \quad (2.37)$$

$$\max_{z \in \mathcal{P}} \varrho_i(z) \leq \max_{z \in \text{vert}(\mathcal{P})} \varrho_i^a(z) - \bar{\varrho}_i^b(z), \quad (2.38)$$

for $i = 1, \dots, m$, with $\text{vert}(\mathcal{P})$ being the set of vertices of $\mathcal{P}$, with $\bar{\varrho}_i^a(z) - \varrho_i^b(z)$ being a concave function, and with $\varrho_i^a(z) - \bar{\varrho}_i^b(z)$ being a convex function.

According to Proposition 2.3.1, the optimization problems are influenced by two elements, namely: (i) polytope $\mathcal{P}$; and (ii) convex functions ϱ^a and ϱ^b in $\mathcal{P}$. Regarding (i), the vertices of a polytope are individually evaluated over (2.37)-(2.38). In order to control the number of vertices, and thereby, the computational cost associated to (2.37)-(2.38), we approximate the polytope $\mathcal{P}$ by either its interval hull $\square\mathcal{P} \supseteq \mathcal{P}$ or a parallelotope $\mathcal{T} \supseteq \mathcal{P}$, since these representations involve 2^n vertices. Polytopes are here represented by CZs. Therefore, the interval hull $\square\mathcal{P}$ is obtained with Proposition 2.2.3. In order to obtain a tight parallelotope $\mathcal{T}$ to polytope $\mathcal{P}$, we next propose a new result, in which a candidate parallelotope $\mathcal{C} \supseteq \mathcal{P}$ is tightened via LPs. The set $\mathcal{C}$ is here computed as in Remark 2.3.1. In Example 2.3.1, we illustrate that the candidate parallelotope $\mathcal{C}$ affects the precision of the parallelotope $\mathcal{T}$.

Remark 2.3.1. Given a CZ $\mathcal{P}$, a parallelotope $\mathcal{C} \supseteq \mathcal{P}$ is obtained by reducing all constraints and generators of $\mathcal{P}$ with Method 4 of Yang & Scott (2018).

Proposition 2.3.2. Let $\mathcal{C} = \{G^c, \bar{z}^c\} \subset \mathbb{R}^n$ be a parallelotope containing the CZ $\mathcal{P} = \{G^z, \bar{z}^z, A^z, b^z\} \subset \mathbb{R}^n$. By solving the LPs

$$\begin{aligned} \zeta_i^L &= \min_{\xi^c, \xi^z} \left\{ \xi_i^c : G_{i,:}^c \xi^c + \bar{z}_i^c = G_{i,:}^z \xi^z + \bar{z}_i^z, \xi^c \in \mathcal{B}^n, \xi^z \in \mathcal{B}(A^z, b^z) \right\}, \quad i = 1, \dots, n \\ \zeta_i^U &= \max_{\xi^c, \xi^z} \left\{ \xi_i^c : G_{i,:}^c \xi^c + \bar{z}_i^c = G_{i,:}^z \xi^z + \bar{z}_i^z, \xi^c \in \mathcal{B}^n, \xi^z \in \mathcal{B}(A^z, b^z) \right\}, \quad i = 1, \dots, n, \end{aligned}$$

we obtain the parallelotope $\mathcal{T} = \{G^t, \bar{z}^t\} \subseteq \mathcal{C}$, where

$$\begin{aligned} G^t &= G^c \text{diag}(\text{rad}([\zeta^L, \zeta^U])), \\ \bar{z}^t &= \bar{z}^c + G^c \text{mid}([\zeta^L, \zeta^U]). \end{aligned}$$

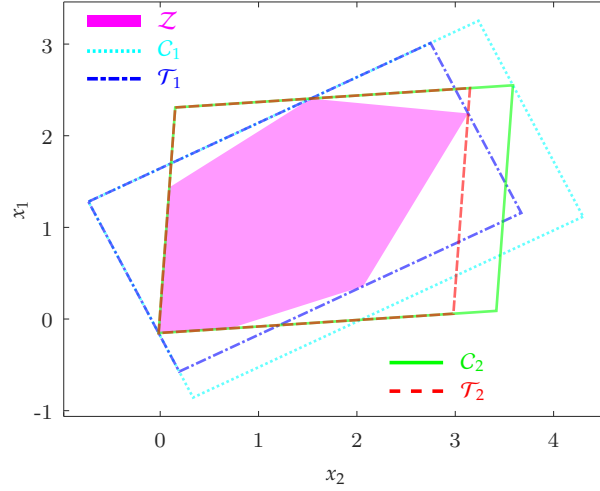


Figure 2.7: 2D illustration of candidate and optimal parallelotopes.

Proof. Given the parallelotope $\mathcal{C} \supseteq \mathcal{P}$, a new parallelotope $\mathcal{T}$ is investigated to contain the CZ $\mathcal{P}$. The set $\mathcal{T}$ must be an outer approximation of $\mathcal{P}$ containing or being equal to $\mathcal{C}$. To achieve that, both the generator matrix G^c and the center $\bar{z}^c$ of $\mathcal{C}$ are firstly fixed. After, the slack variable $\xi^c \in \mathcal{B}^n$ is manipulated to tighten the facets of $\mathcal{C}$ onto the CZ $\mathcal{P}$. This strategy implies $2n$ LPs with $G_{i,:}^c \xi^c + \bar{z}_i^c = G_{i,:}^z \xi^z + \bar{z}_i^z$ being the i th linear equality constraint, where $i = 1, \dots, n$ and $\xi^z \in \mathcal{B}(A^z, b^z)$. Since the minimization and maximization of ξ_i^c yield the smallest possible box $[\zeta^L, \zeta^U] \subset \mathcal{B}^n$, we guarantee that $\mathcal{T} \subseteq \mathcal{C}$. Then, $\mathcal{T} = G^c[\zeta^L, \zeta^U] \oplus \bar{z}^c$ is a parallelotopic outer approximation of $\mathcal{P}$, which can be rewritten as a zonotope using a rescaling (see $\mathcal{T}$ in the statement of Proposition 2.3.2). ■

Example 2.3.1. Let $\mathcal{P}^\downarrow$ be the zonotope obtained by eliminating all equality constraints of the CZ

$$\mathcal{P} = \left\{ \begin{bmatrix} 0.8843 & 0.4264 & 0.4563 & 0.8526 \\ 0.0147 & 0.4841 & 0.9564 & 0.3430 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 0.1876 & 0.9477 & -0.5257 & -1.1156 \end{bmatrix}, -1.5923 \right\}$$

using CZR in (2.31). Applying Lemma 3 of Alamo et al. (2008) to $\mathcal{P}^\downarrow$ yields the parallelotope $\mathcal{C}_1$, whereas reducing all generators of $\mathcal{P}^\downarrow$ using Method 4 of Yang & Scott (2018) yields the parallelotope $\mathcal{C}_2$. By applying Proposition 2.3.2 to $\mathcal{C}_1$ and $\mathcal{C}_2$, we obtain the corresponding parallelotopes $\mathcal{T}_1$ and $\mathcal{T}_2$, with $\mathcal{T}_2$ being more precise than $\mathcal{T}_1$. These sets are sketched in Figure 2.7.

To obtain the vertices of a polytope $\mathcal{P}$, we can employ specific algorithms as those presented in Dyer (1983). Nevertheless, when $\mathcal{P} = \{G^z, \bar{z}\}$ is a box or a parallelotope, the desired vertices can be directly obtained by applying the affine mapping to the vertices of the unitary box $\mathcal{B}^n$, that is,

$$\text{vert}(\mathcal{P}) = G^z \text{vert}(\mathcal{B}^n) + \bar{z} \mathbf{1}_{1 \times 2^n}.$$

Assuming that the set $\text{vert}(\mathcal{B}^n)$ is already known, this procedure has complexity $O(n^2 2^n)$.

In state estimation, we may be interested in employing Proposition 2.3.1 to characterize the linearization remainder $\mathcal{R}$ of a given function ϱ with uncertain input $\mathcal{Z}$ and deterministic input u . To achieve that, we present next Lemma 2.3.1. For practical reasons, the input CZ $\mathcal{Z}$ may be outer-approximated by either a box $\square\mathcal{Z}$ (Proposition 2.2.3) or a parallelotope $\mathcal{T}$ (Proposition 2.3.2), yielding the desired polytope $\mathcal{P}$, before solving the problems (2.40)-(2.41). Then, the control and the manipulation of the vertices are the main challenges of DC programming.

Lemma 2.3.1 (Adapted from Alamo et al. (2008)). Let $\varrho : \mathbb{R}^n \times \mathbb{R}^p \rightarrow \mathbb{R}^m$ be a function in the CZ $\mathcal{Z} \subset \mathbb{R}^n$ and with deterministic input $u \in \mathbb{R}^p$, whose first-order expansion is given by

$$\bar{\varrho}(z, u) = \varrho(\bar{z}, u) + F(z - \bar{z}), \quad (2.39)$$

where $F = \nabla_z \varrho(\bar{z}, u)$, $\bar{z} \in \mathcal{P}$ is any punctual estimate, and $\mathcal{P} \supseteq \mathcal{Z}$ is a convex polytope. Let $e(z, u) \triangleq \varrho(z, u) - \bar{\varrho}(z, u)$ be the linearization error. Let also ϱ^a and ϱ^b be convex functions such that $\varrho = \varrho^a - \varrho^b$ is DC on $\mathcal{P}$. Finally, let $(\varrho^a - \bar{\varrho}^b - \bar{\varrho})$ be a convex majorant of e , and let $(\bar{\varrho}^a - \varrho^b - \bar{\varrho})$ be a concave minorant of e . Then, according to Proposition 2.3.1, a linearization enclosure $\mathcal{R} = [e^-, e^+] \ni e$ is given by

$$e_i^- = \min_{z \in \text{vert}(\mathcal{P})} (\bar{\varrho}_i^a(z, u) - \varrho_i^b(z, u) - \bar{\varrho}_i(z, u)), \quad (2.40)$$

$$e_i^+ = \max_{z \in \text{vert}(\mathcal{P})} (\varrho_i^a(z, u) - \bar{\varrho}_i^b(z, u) - \bar{\varrho}_i(z, u)), \quad (2.41)$$

for $i = 1, \dots, m$. Once (2.40)-(2.41) are solved, the intervals are expressed as the zonotope $\mathcal{R} = \{G^e, c^e\}$ with

$$G^e = \text{diag}(\text{rad}([e^-, e^+])), \quad (2.42)$$

$$c^e = \text{mid}([e^-, e^+]). \quad (2.43)$$

Proof. This proof is similar to Alamo et al. (2008, Proof of Lemma 1), with the difference being that $\mathcal{Z}$ is a CZ (instead of zonotope), and ϱ is any DC function in $\mathcal{P}$ and with deterministic input u . ■

Remark 2.3.2. In Lemma 2.3.1, the variable u is differed from the uncertain variable z because it is deterministic (or a singleton). Then, the linearization of ϱ with respect to u is zero. It means that z is actually responsible by the uncertainty of e .

The choice of the DC components is also important since the optimization problems (2.37)-(2.38) are other sources of conservativeness. Note that different DC functions provide different bounds (Tao & An, 1997). Having this aspect in mind, Proposition 2.3.3 presents a procedure to compute the DC decomposition of a function ϱ . Specifically, we combine quadratic functions of Alamo et al. (2008) with the choice of eigenvalue of

Adjiman & Floudas (1996). This choice is not unique, see for instance (Rohn, 1998), but it should be made carefully since the outer approximation of eigenvalue yields wrapping effect in Proposition 2.3.1.

Proposition 2.3.3 (Adapted from Adjiman & Floudas (1996) and Alamo et al. (2008)). Let $\varrho : \mathbb{R}^n \times \mathbb{R}^p \rightarrow \mathbb{R}^m$ be a function of class $\mathcal{C}^2$ in $\mathcal{P}$ and with deterministic input $u \in \mathbb{R}^p$, and let $\square\mathcal{P} \subset \mathbb{R}^n$ be the interval hull of $\mathcal{P}$. Consider functions $\varrho_i^a(z, u) = \varrho_i(z, u) + \varrho_i^b(z)$ and $\varrho_i^b(z) = \frac{\tilde{\lambda}_i}{2} z^\top z$, for $i = 1, \dots, m$, where

$$\tilde{\lambda}_i = \max\{0, -\check{\lambda}_i\}, \quad (2.44)$$

with $\check{\lambda}_i \in \mathbb{R}$ being computed as in Adjiman & Floudas (1996, Equation (12)), which is a lower bound for the smallest eigenvalue of interval Hessian matrix $[H_i] = \frac{\partial^2 \varrho_i(\square\mathcal{P}, u)}{\partial z^2}$. Then, $\varrho = \varrho^a - \varrho^b$ is a DC function on $\square\mathcal{P}$.

Proof. This proof follows the same discussion presented in Alamo et al. (2008, Section 3), with the difference being the introduction of the deterministic input u . ■

Remark 2.3.3. Analogously to Remark 2.3.2, the quadratic correction $\frac{\tilde{\lambda}_i}{2} z^\top z$ performed by Proposition 2.3.3 includes only uncertain variables since these are responsible by the Hessian matrix is interval.

Remark 2.3.4. Each function ϱ_i^b could be defined with $\frac{1}{2} z^\top Q_i z$, where Q_i is a diagonal matrix whose elements could be obtained via semidefinite programming (Tottoli et al., 2022). Moreover, instead of convexifying ϱ (obtaining ϱ^a), we could convexify $-\varrho$ and place this result in ϱ^b .

Note in Proposition 2.3.3 that the interval Hessian matrix $[H_i] \subset \mathbb{R}^{n \times n}$ is symmetric and has 2^{n^2} vertices; this quantity of vertices could be confirmed via vectorization of matrix. A way to determine $\check{\lambda}_i$ in (2.44) is to evaluate the smallest eigenvalue of all 2^{n^2} vertex matrices. Due to the low efficiency of the prior procedure, Adjiman & Floudas (1996) revisit a result proposed by Hertz (1992) in which, given a symmetric interval matrix, only 2^{n-1} symmetric vertex matrices must be investigated. This amount of vertices comes from the division of the n -dimensional Euclidean space into 2^n orthants, from which only 2^{n-1} orthants are necessary due to their symmetry with respect to the origin. In Algorithm 2.1, we present a manner to obtain the lower bound of eigenvalue (LBE), whose complexity order is given by $O(\text{LBE}) = O(n^3 m 2^{n-1})$.

2.4 Mixed-Uncertainty Vector

In order to combine a GRV and a CZ, the mixed-uncertainty vector is defined next. For completeness, this definition is illustrated in Figure 2.8. The time step k is made explicit

Algorithm 2.1: $\check{\lambda} = \text{LBE}([H_1], \dots, [H_m], \text{vert}(\mathcal{B}^n))$ (Hertz, 1992)

```

1  Considering that the  $2^n$  vertices of  $\mathcal{B}^n$  are symmetrically sorted, define  $S \in \mathbb{R}^{n \times N}$  as
   the first  $N = 2^{n-1}$  non-symmetric vectors from  $\text{vert}(\mathcal{B}^n)$ 
2   $\check{\lambda} = 0_{m \times 1}$ 
3  for  $l = 1, \dots, m$  do
4     $M_l = \inf([H_l])$ 
5  end
6  for  $k = 1, \dots, N$  do
7    for  $i = 1, \dots, n-1$  do
8      for  $j = i+1, \dots, n$  do
9        if  $S_{ik}S_{jk} < 0$  then
10         for  $l = 1, \dots, m$  do
11            $M_{(i,j),l} = \sup_{i,j}([H_l])$ 
12            $M_{(j,i),l} = M_{(i,j),l}$ 
13         end
14         else
15           for  $l = 1, \dots, m$  do
16              $M_{(i,j),l} = \inf_{i,j}([H_l])$ 
17              $M_{(j,i),l} = M_{(i,j),l}$ 
18           end
19         end
20       end
21     end
22     for  $l = 1, \dots, m$  do
23        $\gamma = \min \text{eig}(M_l)$ , where eig returns the  $n$  eigenvalues of  $M_l$ 
24       if  $\gamma < \check{\lambda}_l$  then
25          $\check{\lambda}_l = \gamma$ 
26       end
27     end
28 end

```

because mixed-uncertainty vectors will be employed in dynamical systems over Chapters 4 and 5, and Appendix C. The center c_k^x is destined to concentrate all contribution of centers and means, whereas the CZ $\mathcal{X}_k$ and the GRV X_k are devoted to characterize both the shape and the size of the mixed uncertainty.

Definition 2.4.1. Consider the CZ $\mathcal{X}_k = \{G_k^x, 0_{n \times 1}, A_k^x, b_k^x\}$ and the GRV $X_k \sim \mathcal{N}(0_{n \times 1}, P_k^{\text{xx}})$. Given the elements z_k^x from $\mathcal{X}_k$ and realizations g_k^x from X_k , the mixed-uncertainty vector x_k is defined as

$$x_k \triangleq c_k^x + z_k^x + g_k^x, \quad (2.45)$$

where $c_k^x \in \mathbb{R}^n$ is the mixed center of x_k .

Remark 2.4.1. The terms *accuracy* and *precision* are often used throughout the thesis. The former denotes exactness of punctual estimates (may centers and means) with respect

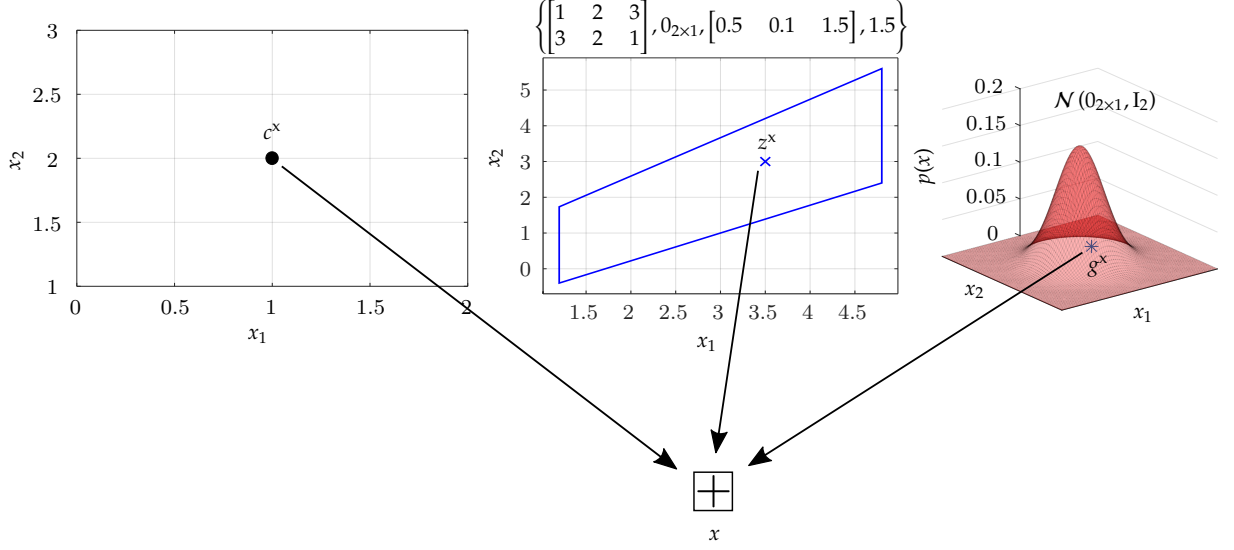


Figure 2.8: Diagram to illustrate the mixed-uncertainty vector in 2D. The vector c^x ($\bullet$) represents the center of the mixed-uncertainty vector x . The realization g^x ($*$) of a GRV (red graph) has a probability $p(g^x)$ related to; for brevity, the realizations of this GRV are illustrated into a box containing a larger 99.9% confidence level. The element z^x ($\times$) corresponds to a point into a CZ (blue graph). The sum of these terms yields the mixed-uncertainty vector x . This example illustrates a case in which the center of CZ does not belong to the own set because $G^x \xi \neq 0_{2 \times 1}$ for $\xi \in \mathcal{B}(A^x, b^x)$ (Rego et al., 2020a). For this reason, applications as linearization and state feedback may demand a punctual estimate more suitable than c^x to improve reliability and accuracy.

to the system state, while the latter refers to the size of uncertainty (may radius and volume).

Remark 2.4.2. In order to obtain the same mixed-uncertainty vector defined in Combastel (2015), we replace CZs by zonotopes in Definition 2.4.1, that is, $\mathcal{X}_k = \{G_k^x, 0_{n \times 1}\}$.

Since GRV is employed in Definition 2.4.1 to characterize the stochastic parcel, the domain/support of the mixed-uncertainty vector is necessarily unbounded. The truncation of this domain may be interesting in several applications to assign sensitivity (fault diagnosis), robustness (guaranteed approximations), and precision (Kalman filtering). By analyzing the definition of a chi-square random variable X^x (2.1), we realize that its realizations $x^x \geq 0$ are obtained (in increasing order of magnitude) from the center $\bar{x}$ (highest probability) to the radial values (decreasing probability) of an associated GRV. If a chi-square PDF is integrated up to a value ς as in (2.2), then we obtain the so-called confidence level $c(\varsigma) = 1 - \alpha$, where $\alpha \in [0, 1]$ is called significance level or type-I error. This integration process includes the realizations of a GRV in a radial form, whose resulting set is the confidence ellipsoid defined as follows.

Definition 2.4.2 (Combastel (2015)). Consider the GRV $X \sim \mathcal{N}(\bar{x}, P^{xx})$, where $P^{xx} \succ 0_{n \times n}$,

and the significance level, or type I error, $\alpha \in [0, 1]$. The confidence ellipsoid is defined as

$$\mathcal{E} \triangleq \{\bar{x}, P^{\text{xx}}, \alpha\} = \{x \in \mathbb{R}^n : (x - \bar{x})^\top (\varsigma P^{\text{xx}})^{-1} (x - \bar{x}) \leq 1\}, \quad (2.46)$$

where $\varsigma \geq 0$ is the greatest value for the chi-square random variable with n degrees of freedom (2.1), taken from the cumulative distribution function (2.2) with confidence level of $(1 - \alpha)$, such that the probability $p(x \in \mathcal{E}) = (1 - \alpha)$ is satisfied.

Next, we define the *confidence constrained zonotope*, in order to merge a CZ with a GRV, given a significance level α , in a CZ. In addition, we define another confidence set, called *confidence box*. Such sets will be resumed in posterior chapters to compute punctual estimates and approximate nonlinear models. Recall that CZs are more precise than boxes whereas boxes have lower complexity. For completeness, these definitions are exemplified in Example 2.4.1. Unlike the confidence set proposed in Li et al. (2021), our defined sets are not limited to applications with finite time horizon. Moreover, our defined sets result in more precise sets than in Combastel (2015) and Li et al. (2021) and have direct applicability to fault diagnosis.

Definition 2.4.3. Let the mixed center c^x , the CZ $\mathcal{X}$, and the GRV X with covariance P^{xx} be characterizations of the mixed-uncertainty vector x . Let also the confidence ellipsoid $\mathcal{E}$ be defined as (2.46). Then, the confidence CZ $\mathcal{X}^\alpha$ is defined as

$$\mathcal{X}^\alpha \triangleq c^x \oplus \mathcal{X} \oplus \mathcal{P}, \quad (2.47)$$

where $\mathcal{P} = \{G^e, 0_{n \times 1}\}$ is the smallest parallelotope containing $\mathcal{E}$, whose generator matrix is computed as (Bo et al., 2013)

$$G^e = (\varsigma P^{\text{xx}})^{1/2}, \quad (2.48)$$

$(\varsigma P^{\text{xx}})^{1/2}$ is the lower triangular matrix obtained by Cholesky decomposition, and $\varsigma \geq 0$ is defined in (2.46).

Definition 2.4.4. Let the mixed center c^x , the CZ $\mathcal{X}$, and the GRV X with covariance P^{xx} be characterizations of the mixed-uncertainty vector x (2.45). Let also the confidence ellipsoid $\mathcal{E}$ be defined as (2.46). Then, the confidence box $\mathcal{I}^\alpha$ is defined as

$$\mathcal{I}^\alpha \triangleq c^x \oplus \square \mathcal{X} \oplus \square \mathcal{E}, \quad (2.49)$$

where $\square \mathcal{X} = [\zeta^{\text{Lz}}, \zeta^{\text{Uz}}]$ is the interval hull of $\mathcal{X}$, whose bounds are computed as in (2.32)-(2.33), $\square \mathcal{E} = [\zeta^{\text{Lg}}, \zeta^{\text{Ug}}]$ is the interval hull of $\mathcal{E}$ (2.46), whose bounds are computed as

$$\zeta^{\text{Lg}} = -r^g, \quad (2.50)$$

$$\zeta^{\text{Ug}} = r^g, \quad (2.51)$$

$$r^{\mathcal{E}} = \sqrt{\varsigma \text{diag}(P^{\text{xx}})}, \quad (2.52)$$

$\varsigma \geq 0$ is defined in (2.46), and $\text{diag}(P^{\text{xx}})$ is a vector containing the diagonal elements of the matrix P^{xx} .

Remark 2.4.3. In Definition 2.4.4, we present the confidence box without starting from $\mathcal{X}^\alpha$ given by Definition 2.4.3. This is motivated by the fact that $\square\mathcal{E}$ can be analytically computed and the outer approximation of ellipsoids by parallelotopes yields larger interval hulls. Then, besides reducing the computational burden by solving smaller LPs and by avoiding Cholesky decomposition, we obtain smaller interval hulls with Definition 2.4.4. Further complexity reduction can be obtained if $\mathcal{X}$ is a zonotope with generator matrix $G^\times$ and center $\bar{x}$. In this case, its interval hull is analytical and readily computed using row sum (2.26). According to this latter methodology, we can also reduce the computational burden of $\square\mathcal{X}$ for CZs at the cost of slightly enlarging the exact solution. First, we eliminate all constraints of a CZ $\mathcal{X}$ by applying CZR, yielding a zonotope $\tilde{\mathcal{X}} \supset \mathcal{X}$. After, we use analytical expressions to obtain the interval hull of $\tilde{\mathcal{X}}$. In doing so, we replace the worst-case complexity of Proposition 2.2.3 $O\left(nn_g(n_h + n_g)^3\right)$ (Rego et al., 2020a) by $O\left(n_h(n_h + n_g)^3 + nn_h n_g^2\right)$.

The use of both Definitions 2.4.3 and 2.4.4 is illustrated in Example 2.4.1.

Example 2.4.1. Consider a mixed-uncertainty vector x characterized by

$$c^\times = \begin{bmatrix} 1 & 3 \end{bmatrix}^\top, \quad \mathcal{X} = \left\{ \begin{bmatrix} 2 & 3 & 4 \\ 4 & 3 & 2 \end{bmatrix}, 0_{2 \times 1}, \begin{bmatrix} 0.5 & 0.1 & 1.5 \end{bmatrix}, 1 \right\}, \quad X \sim \mathcal{N}\left(0_{2 \times 1}, \begin{bmatrix} 1 & -1 \\ -1 & 2 \end{bmatrix}\right).$$

Consider also a significance level $\alpha = 0.0027$. Next, we show how Definitions 2.4.3 and 2.4.4 use such information to yield the sets sketched in Figure 2.9. Given X and α , as in Definition 2.4.3 compute the confidence ellipsoid $\mathcal{E}$ and, thereby, the parallelotope $\mathcal{P}$, whose Minkowski sum with the mixed center $c^\times$ and the CZ $\mathcal{X}$ yields the confidence CZ $\mathcal{X}^\alpha$ in (2.47). Definition 2.4.4 computes the separate interval hull of $\mathcal{E}$ and $\mathcal{X}$, whose Minkowski sum with the center $c^\times$ yields the confidence box $\mathcal{I}^\alpha$ in (2.49). This result is more precise than computing the direct interval hull of $\mathcal{X}^\alpha$ given by Definition 2.4.3 since $\square\mathcal{E} \subseteq \square\mathcal{P}$.

2.5 Optimization Problems

This section revisits some results on optimization that are necessary to develop the methods discussed and proposed on this research project. Next, some relations involving the matrix trace and vectorization are summarized.

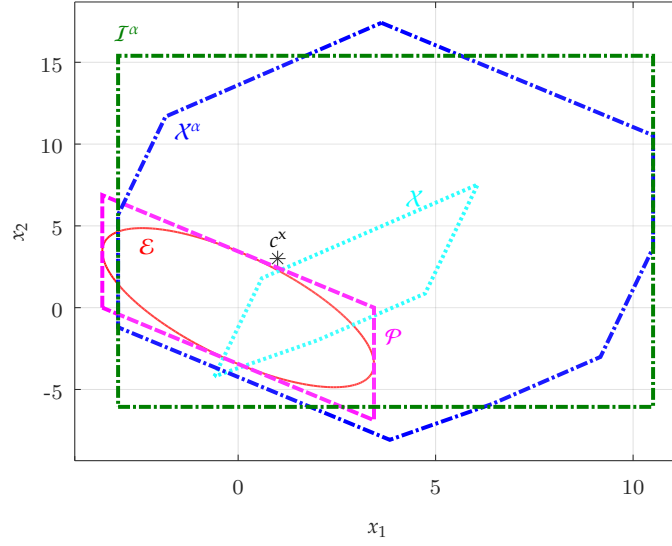


Figure 2.9: Illustration of the application of Definitions 2.4.3 and 2.4.4 to a mixed-uncertainty vector, whose results are a confidence CZ $\mathcal{X}^\alpha = c^x \oplus \mathcal{X} \oplus \mathcal{P}$ and a confidence box $\mathcal{I}^\alpha = c^x \oplus \square \mathcal{X} \oplus \square \mathcal{E}$.

Proposition 2.5.1 (Bernstein (2009)). Let K, M, N, O , and P be matrices with appropriate dimensions. Then, the following relations hold:

$$\text{tr}(M) = \text{tr}(M^\top) : M \in \mathbb{R}^{n \times n}, \quad (2.53)$$

$$\text{tr}(MN) = \text{tr}(NM) : M \in \mathbb{R}^{n \times m}, N \in \mathbb{R}^{m \times n}, \quad (2.54)$$

$$\text{tr}(M + N) = \text{tr}(M) + \text{tr}(N) : M \in \mathbb{R}^{n \times n}, N \in \mathbb{R}^{n \times n}, \quad (2.55)$$

$$\frac{\partial \text{tr}(MK^\top N)}{\partial K} = M^\top N^\top : K \in \mathbb{R}^{l \times m}, M \in \mathbb{R}^{n \times m}, N \in \mathbb{R}^{l \times n}, \quad (2.56)$$

$$\frac{\partial \text{tr}(MKNK^\top)}{\partial K} = NK^\top M + N^\top K^\top M^\top : K \in \mathbb{R}^{n \times m}, M \in \mathbb{R}^{n \times n}, N \in \mathbb{R}^{m \times m}, \quad (2.57)$$

$$\text{tr}(MN) = \text{vec}(M^\top)^\top \text{vec}(N) : M \in \mathbb{R}^{n \times m}, N \in \mathbb{R}^{m \times n}, \quad (2.58)$$

$$\text{tr}(MNO) = \text{vec}(M^\top)^\top (\mathbf{I}_n \otimes N) \text{vec}(O) : M \in \mathbb{R}^{n \times m}, N \in \mathbb{R}^{m \times l}, O \in \mathbb{R}^{l \times n}, \quad (2.59)$$

$$\text{vec}(MNO) = (O^\top \otimes M) \text{vec}(N) : M \in \mathbb{R}^{n \times m}, N \in \mathbb{R}^{m \times l}, O \in \mathbb{R}^{l \times k}, \quad (2.60)$$

$$\text{vec}(M^\top) = P \text{vec}(M) : M \in \mathbb{R}^{n \times m}, P \in \mathbb{R}^{nm \times nm}, \quad (2.61)$$

where $\text{tr}(\cdot)$ is the matrix trace, $\text{vec}(\cdot)$ is the matrix vectorization, $\otimes$ is the Kronecker product, and $P = \sum_{i=1}^n \sum_{j=1}^m E_{i,j,n \times m} E_{i,j,n \times m}^\top$ is the vectorized transpose matrix, with $E_{i,j,n \times m} \triangleq (\mathbf{I}_n)_{:,i} (\mathbf{I}_m)_{:,j}^\top$.

The general solution for linear systems can be parametrized as follows.

Lemma 2.5.1 (Wang et al. (2018c)). Consider the matrices $A \in \mathbb{R}^{a \times b}$, $B \in \mathbb{R}^{b \times c}$, and $C \in \mathbb{R}^{a \times c}$, where $\text{rank}(B) = c$, and A is an unknown matrix. The general solution of $AB = C$ is given by

$$A = CB^+ + S(\mathbf{I}_b - BB^+), \quad (2.62)$$

where $S \in \mathbb{R}^{a \times b}$ is an arbitrary matrix, and $(\cdot)^+$ is the pseudo-inverse operator, that is,

$$B^+ \triangleq (B^\top B)^{-1} B^\top. \quad (2.63)$$

The prior result is useful to formulate optimization problems based on matrix inequalities. The H_2 and H_∞ norms of an uncertain linear time-invariant system are formulated as follows.

Lemma 2.5.2 (Boyd et al. (1994)). Consider the discrete-time state-space system described by

$$x_k = Ax_{k-1} + Bw_{k-1}, \quad (2.64)$$

$$z_k = Cx_k, \quad (2.65)$$

where $A \in \mathbb{R}^{a \times a}$, $B \in \mathbb{R}^{a \times b}$, and $C \in \mathbb{R}^{c \times a}$ are constant matrices, and $w_{k-1} \in \mathbb{R}^b$ is the noise input. Consider also a symmetric and positive-definite matrix $P = P^\top > 0_{a \times a}$ and a variable $\psi > 0$. Then, by solving the optimization problem

$$\begin{aligned} \min \quad & \psi \\ \text{s.t.} \quad & A^\top P A - P + C^\top C < 0_{a \times a}, \quad B^\top P B < H, \quad \text{tr}(H) < \psi, \end{aligned} \quad (2.66)$$

the optimal value $\check{\psi}$ is obtained, which represents the square H_2 norm of (2.64)-(2.65).

Lemma 2.5.3 (Xu & Lam (2006)). Consider the discrete-time system (2.64)-(2.65). Consider also a symmetric and positive-definite matrix $P = P^\top > 0_{a \times a}$ and a variable $\gamma > 0$. Then, by solving the optimization problem

$$\begin{aligned} \min \quad & \gamma \\ \text{s.t.} \quad & \begin{bmatrix} A^\top P A + C^\top C - P & A^\top P B \\ B^\top P A & B^\top P B - \gamma I_b \end{bmatrix} < 0_{(a+b) \times (a+b)}, \end{aligned} \quad (2.67)$$

the optimal value $\check{\gamma}$ is obtained, which represents the square H_∞ norm of (2.64)-(2.65).

The following lemma uses the power spectral density of a white noise to minimize the steady-state variance of the estimation error. In Khargonekar et al. (1996), a similar result is formulated for continuous-time stochastic processes, but here it is formulated for discrete-time stochastic processes.

Lemma 2.5.4 (Adapted from Khargonekar et al. (1996)). Consider the discrete-time error system described as (2.64)-(2.65), where $w_{k-1} \in \mathbb{R}^b$ is a realization of the white noise $\{W\}$ at time $k-1$. Assume that the power spectral density of $\{W\}$ is $S^w = I_b$, and denote $G \in \mathbb{R}^{c \times b}$ the transfer matrix from the noise input w to the error output z , that is, $z = Gw$. Then,

the steady-state variance of the estimation error is given by

$$\begin{aligned} \lim_{k \rightarrow \infty} \mathbb{E}[z_k^\top z_k] &= \frac{1}{2\pi} \int_{-\pi}^{\pi} \text{tr}(GS^w G^*) d\omega \\ &= \|G\|_{\mathcal{H}_2}^2, \end{aligned} \tag{2.68}$$

where $(\cdot)^*$ is the complex conjugate transpose, and $\|\cdot\|_{\mathcal{H}_2}$ is the $\mathcal{H}_2$ norm of system.

Note that the covariance matrix of the white noise is assumed to be $Q = I_b$ in Lemma 2.5.4. In order to consider cases where this assumption is not true, the positive-definite matrix $Q > 0_{b \times b}$ can be decomposed as $Q = LL^\top$, where L is the Cholesky decomposition of Q . After, the matrix L is merged with the input matrix B in (2.64), to yield $\bar{B} = BL$. Then, without loss of generality, the matrix Q used in Lemma 2.5.4 is seen as an identity matrix.

2.6 Final Remarks

This chapter reviewed some contents to support and clarify developments showed in the next chapters. Regarding the random vectors, we focused on GRVs and chi-square variables. Regarding the sets, we defined ellipsoids, boxes, strips, zonotopes, and CZs. Both stochastic and set-based frameworks are considered to propose or extend algorithms with mixed-uncertainty vectors. A mixed-uncertainty vector is a combination of different representations for uncertainties, which allows to incorporate some specific characteristics of each approach. GRVs introduce confidence due to the stochastic nature of uncertainties, while sets attribute robustness due to the bounded nature. Although uncertainties are bounded in practice, sets are here devoted to represent uncertainties whose statistical knowledge is unknown. These characteristics will be explored in the next chapters.

3

Extensions of Stochastic and Set-based Approaches

In this chapter, we present some proposals to extend state estimators described in Appendices A and B. The main characteristics of the new algorithms are highlighted in Table 3.1. Part of the contributions is addressed to descriptor systems by means of a new stochastic filter (DGF) and a new performance criterion (DRZF). In comparison with the motivational algorithms, we may improve precision and/or accuracy at the cost of a larger complexity. Other part of the contributions is addressed to state-space systems, in which we highlight the proposal of interval (ZKF) and set-membership (IZMV and CZDC) observers. The ZKF extends a predictor (ZKP in Subsection B.1.1) to filter, whose motivation is to reach better precision and accuracy. This is expected because filters assimilate more measurements than predictors. The IZMV extends the ZMV (Subsection B.2.2), based on zonotopes and mean value extension, by additionally enforcing state inequality constraints and reducing the computational burden of the volume criterion. In turn, CZDC combines CZs and DC programming to build a new set-membership filter, which is able to treat constrained nonlinear systems.

Table 3.1: Overview of the stochastic and set-based algorithms proposed in this chapter. The numeration of each algorithm is presented between brackets in the first column, with our proposals being differed from the existing approaches (just above) through boldface. The remaining columns characterize the algorithms regarding: uncertainty — Gaussian random vector (GRV), zonotope (Zon), constrained zonotope (CZ), strip —, system — state space (SS), descriptor (D), linear time invariant (LTI), linear time varying (LTV) —, approximation — mean value extension (MV), difference of convex functions programming (DC) —, criterion — minimum trace (MT), minimum radius (MR), H_2 , H_∞ , segment minimization (SM), volume minimization (VM) —, category — predictor (P), filter (F) —, and constraints — equality constraints (EC), inequality constraints (IC) —. The proposed state estimators are divided in two groups: (i) stochastic, and (ii) set based.

Algorithm	Uncertainty	System	Approx.	Criterion	Cat.	Constraints
KP [A.1]	GRV	SS, LTV		MT	P	
DGF [3.1]	GRV	D , LTV		MT	F	EC
DH₂GF [3.2]	GRV	D , LTI		H₂	F	EC
DTZF [B.3]	Zon	D, LTV		MT	F	EC
DRZF [3.3]	Zon	D, LTV		MR	F	EC
ZKP [B.2]	Zon	SS, LTV		MT	P	
ZKF [3.4]	Zon	SS, LTV		MT	F	
NH_∞ZF [3.5]	Zon	SS, LTI		H_∞	F	
ZMV [B.13]	Zon, Strip	SS	MV	SM, VM1	F	
IZMV [3.6]	Zon, Strip	SS	MV	SM, VM3	F	IC
CZLF [B.5]	CZ	SS, LTV			F	
CZDC [3.8]	CZ	SS	DC		F	EC, IC

3.1 Linear Systems

3.1.1 Gaussian Filter for Descriptor Systems

In this subsection, we propose the Gaussian filter for descriptor LTV systems (DGF). This algorithm extends the Kalman predictor (KP) (Kalman, 1960) (see Subsection A.1.1), since its estimator structure is more general. Therefore, the KP is a special case of the DGF. Our proposal is motivated by the minimum-trace zonotopic filter for descriptor LTV systems (DTZF) (Tang et al., 2020) (see Subsection B.1.2), in which time-varying extra matrices are introduced to tighten the state estimates. This algorithm composes part of the paper de Paula et al. (2022b).

Remark 3.1.1. We use the term descriptor to make explicit the presence of invariants in a state space. In this case, the state trajectories evolve on hyperplanes.

Consider the discrete-time descriptor LTV dynamical system

$$E_k x_k = A_{k-1} x_{k-1} + B_{k-1} u_{k-1} + w_{k-1}, \quad (3.1)$$

$$y_k = C_k x_k + v_k, \quad (3.2)$$

where $E_k \in \mathbb{R}^{n \times n}$, $A_{k-1} \in \mathbb{R}^{n \times n}$, $B_{k-1} \in \mathbb{R}^{n \times p}$, and $C_k \in \mathbb{R}^{m \times n}$ are the known system matrices, $u_{k-1} \in \mathbb{R}^p$ is the deterministic input vector, $y_k \in \mathbb{R}^m$ is the measured output vector, $w_{k-1} \in \mathbb{R}^n$ is the process noise, $v_k \in \mathbb{R}^m$ is the measurement noise, and $x_k \in \mathbb{R}^n$ is the state vector to be estimated. The terms w_{k-1} and v_k are characterized by the known GRVs $W_{k-1} \sim \mathcal{N}(0_{n \times 1}, Q_{k-1})$ and $V_k \sim \mathcal{N}(0_{m \times 1}, R_k)$, respectively, such that $E[W_k W_j^\top] = 0_{n \times n}$ and $E[V_k V_j^\top] = 0_{m \times m}$ for all $k \neq j$. We assume that $Q_{k-1} > 0_{n \times n}$ and $R_k > 0_{m \times m}$; the KP requires that $Q_{k-1} \geq 0_{n \times n}$. The initial state x_0 is characterized by the known GRV $\hat{X}_0 \sim \mathcal{N}(\hat{x}_0, \hat{P}_0^{\text{xx}})$, with $\hat{P}_0^{\text{xx}} \geq 0_{n \times n}$, and we assume that $\hat{X}_0$, W_{k-1} , and V_k are uncorrelated for $k \in \mathbb{Z}_+$.

The design of DGF starts with the following assumption.

Assumption 3.1.1. The matrices E_k and C_k of the system (3.1)-(3.2) compose the full-rank block matrix $\begin{bmatrix} E_k \\ C_k \end{bmatrix}$.

Since Assumption 3.1.1 holds, there exists a full-rank matrix $\begin{bmatrix} T_k & N_k \end{bmatrix}$, where $T_k \in \mathbb{R}^{n \times n}$ and $N_k \in \mathbb{R}^{n \times m}$ are extra matrices, such that

$$T_k E_k + N_k C_k = I_n. \quad (3.3)$$

This existence is ensured by Lemma 2.5.1.

Post-multiplying (3.3) by x_k and using (3.1)-(3.2), we obtain the following non-descriptor representation:

$$x_k = T_k A_{k-1} x_{k-1} + T_k B_{k-1} u_{k-1} + T_k w_{k-1} + N_k y_k - N_k v_k. \quad (3.4)$$

By employing (3.3), a state-space format was obtained in (3.4). Differently from usual Kalman filtering, note that (3.4) relies on measurements. Then, the predictor/corrector structure is here a slight different since the predictions of x_k combine the knowledge of both the process model (3.1) and the measurement model (3.2). Conversely, this fact does not prohibit the use of that procedure to define a state estimator, in which predictions come from a non-descriptor representation while corrections are weighted by means of a Kalman-like gain.

Therefore, we propose the following mean estimate:

$$\hat{x}_k = (T_k A_{k-1} - K_{k-1} C_{k-1}) \hat{x}_{k-1} + T_k B_{k-1} u_{k-1} + K_{k-1} y_{k-1} + N_k y_k, \quad (3.5)$$

where $K_{k-1} \in \mathbb{R}^{n \times m}$ is the estimator gain. The gain matrix K is here stated at $k-1$ because the predictor/corrector structure is based on y_{k-1} . Note that, when $E = I_n$, $T_k = I_n$, and $N_k = 0_{n \times m}$, (3.5) is reduced to the KP (Kalman, 1960). Then, in comparison with the KP, equation (3.3): (i) adds consistency to the estimator (equality constraints); (ii) allows to

treat descriptor systems; and (iii) confers the filter characteristic for the DGF whenever $N_k \neq 0_{n \times m}$.

By defining the estimation error as $e_k \triangleq x_k - \hat{x}_k$, we obtain

$$e_k = (T_k A_{k-1} - K_{k-1} C_{k-1})(x_{k-1} - \hat{x}_{k-1}) + T_k w_{k-1} - K_{k-1} v_{k-1} - N_k v_k, \quad (3.6)$$

whose expected value is given by

$$E[e_k] = (T_k A_{k-1} - K_{k-1} C_{k-1})E[e_{k-1}] + T_k E[w_{k-1}] - K_{k-1} E[v_{k-1}] - N_k E[v_k]. \quad (3.7)$$

From the problem formulation, we have that $E[w_{k-1}] = 0_{n \times 1}$, $E[v_{k-1}] = 0_{m \times 1}$, and $E[v_k] = 0_{m \times 1}$. Then, $E[e_k] = (T_k A_{k-1} - K_{k-1} C_{k-1})E[e_{k-1}]$. Recall that $\hat{x}_k$ is a mean estimate for a realization of x_k . Actually, $\hat{x}_k$ may not coincide with the true mean $\bar{x}_k = E[x_k]$ in every realization. However, all mean estimates $\hat{x}_k$ are evaluated at $E[e_k]$, and we expect that, in average, $\bar{x}_k$ is reached. Then, by assuming that, in average, $\hat{x}_0$ coincides with the mean $\bar{x}_0$, we guarantee that $E[e_{k-1}] = 0_{n \times 1}$ for all $k \in \mathbb{Z}_+$, and thereby, the estimator (3.5) is *unbiased*. An analogous result is found in Simon (2006, Page 84) for Kalman filtering by assuming that $\hat{x}_0 = \bar{x}_0$. Even though $E[e_0] \neq 0_{n \times 1}$, we guarantee that $\hat{x}_k$ is asymptotically unbiased as follows.

Property 3.1.1. Let (3.7) be the bias of the mean estimator (3.5). Assume that the eigenvalues of $(T_k A_{k-1} - K_{k-1} C_{k-1})$ have module smaller than one. Then, since $E[w_{k-1}] = 0_{n \times 1}$, $E[v_{k-1}] = 0_{m \times 1}$, and $E[v_k] = 0_{m \times 1}$, (3.5) is asymptotically unbiased.

Proof. In Property 3.1.1, we assume that the deterministic matrix $(T_k A_{k-1} - K_{k-1} C_{k-1})$ is Schur stable. It means that $E[e_k] = (T_k A_{k-1} - K_{k-1} C_{k-1})E[e_{k-1}]$ converges to zero when $k \rightarrow \infty$. Then, (3.5) is asymptotically stable, and thereby, asymptotically unbiased. ■

Let $x_k = \bar{x}_k + x_k^g$ be the decomposition of x_k in terms of both the mean $\bar{x}_k$ and the realization x_k^g , with this latter term being characterized by the GRV $X_k^g \sim \mathcal{N}(0_{n \times 1}, P_k^{xx})$. The mean $\bar{x}_k$ is actually unknown in practice. Then, the estimate $\hat{x}_k$ may not coincide with the mean $\bar{x}_k$ in separate realizations, but in average. Therefore, there may exist a mean deviation $(\bar{x}_k - \hat{x}_k)$ in e_k . Since the prior deviation is *unknown*, it should be treated as an additional uncertainty in x_k^g .

Assume that the mean deviation in e_k is shifted to x_k^g , yielding a new variable $\hat{x}_k^g$ with representation $\hat{X}_k^g \sim \mathcal{N}(0_{n \times 1}, \hat{P}_k^{xx})$ such that $\hat{P}_k^{xx} \geq P_k^{xx}$. In this case, we set zero mean because this is the expected value with no deviation. Moreover, the transformed variable preserves the uncorrelation among random variables. Then, $e_k = \hat{x}_k^g$ is given by

$$\hat{x}_k^g = (T_k A_{k-1} - K_{k-1} C_{k-1})\hat{x}_{k-1}^g + T_k w_{k-1} - K_{k-1} v_{k-1} - N_k v_k, \quad (3.8)$$

whose associated covariance matrix is given by

$$\hat{P}_k^{\text{xx}} = (T_k A_{k-1} - K_{k-1} C_{k-1}) \hat{P}_{k-1}^{\text{xx}} (T_k A_{k-1} - K_{k-1} C_{k-1})^\top + T_k Q_{k-1} T_k^\top + K_{k-1} R_{k-1} K_{k-1}^\top + N_k R_k N_k^\top. \quad (3.9)$$

It is noteworthy that the mean deviation $(\bar{x}_k - \hat{x}_k)$ leads the covariance matrix $\hat{P}_k^{\text{xx}}$ to become larger than P_k^{xx} . This is expected because $\hat{X}_k^{\text{g}}$ must represent both X^{g} and the mean deviation, as illustrated next in Example 3.1.1.

Example 3.1.1. Consider that the GRV $X \sim \mathcal{N}(2, 4)$ will be represented by the GRV $\hat{X} \sim \mathcal{N}(\hat{x}, \hat{P}^{\text{xx}})$. Assume that $\hat{x} = 1$ is given, and that the goal is to find the smallest $\hat{P}^{\text{xx}}$ that guarantees a consistent estimate.

First, decompose the realizations of X as $x = 2 + x^{\text{g}}$, with x^{g} being the realizations of $X^{\text{g}} \sim \mathcal{N}(0, 4)$. According to Lefebvre* et al. (2004), $\hat{x}$ will be a consistent estimate if $\mathbb{E}[(x - \hat{x})(x - \hat{x})^\top] \leq \hat{P}^{\text{xx}}$. Therefore,

$$\hat{P}^{\text{xx}} \geq \mathbb{E}[(1 + x^{\text{g}})^2] = \mathbb{E}[1 + 2x^{\text{g}} + (x^{\text{g}})^2] = 1 + 2\mathbb{E}[x^{\text{g}}] + \mathbb{E}[(x^{\text{g}})^2] = 5.$$

Then, $\hat{P}^{\text{xx}} = 5$ is the desired solution. This example illustrates that $\hat{X}^{\text{g}} \sim \mathcal{N}(0, \hat{P}^{\text{xx}})$ requires $\hat{P}^{\text{xx}} > P^{\text{xx}}$ to characterize the error $e = x - \hat{x}$, for a given realization, when $\hat{x} \neq \bar{x}$.

The design matrices T_k , N_k , and K_{k-1} are now specified according to Problem 1.2.1 considering state linear equality constraints, from which we formulate the constrained optimization problem $\min_{T_k E_k + N_k C_k = I_n} \text{tr}(P_k^{\text{xx}})$. In order to solve an unconstrained optimization problem instead, we include the Lagrange multipliers $\Lambda_k \in \mathbb{R}^{n \times n}$ to define the objective function

$$J_k^{\text{eqmv}}(T_k, N_k, K_{k-1}, \Lambda_k) \triangleq \text{tr}(P_k^{\text{xx}}) + \text{tr}(\Lambda_k (E_k^\top T_k^\top + C_k^\top N_k^\top - I_n)). \quad (3.10)$$

The optimal solution of the prior problem is presented next as a proposition. By convenience, we define the following augmented design matrix:

$$\Psi_k \triangleq \begin{bmatrix} T_k & N_k & K_{k-1} \end{bmatrix}. \quad (3.11)$$

Proposition 3.1.1. Consider the objective function J_k^{eqmv} in (3.10) and the augmented design matrix Ψ_k in (3.11). Solving the unconstrained optimization problem $\min_{T_k, N_k, K_{k-1}, \Lambda_k} J_k^{\text{eqmv}}$ yields the optimal solutions

$$\check{\Lambda}_k = -(\Pi_k^\top \Sigma_k^{-1} \Pi_k)^{-1}, \quad (3.12)$$

$$\check{\Psi}_k = -\check{\Lambda}_k \Pi_k^\top \Sigma_k^{-1}, \quad (3.13)$$

where

$$\Pi_k = \begin{bmatrix} E_k^\top & C_k^\top & 0_{n \times m} \end{bmatrix}^\top, \quad (3.14)$$

$$\Sigma_k = 2 \begin{bmatrix} \hat{P}_{k|k-1}^{xx} & 0_{n \times m} & -\hat{P}_{k-1}^{xy} \\ 0_{m \times n} & R_k & 0_{m \times m} \\ \left(-\hat{P}_{k-1}^{xy}\right)^\top & 0_{m \times m} & \hat{P}_{k-1}^{yy} \end{bmatrix}, \quad (3.15)$$

$$\hat{P}_{k|k-1}^{xx} = A_{k-1} \hat{P}_{k-1}^{xx} A_{k-1}^\top + Q_{k-1}, \quad (3.16)$$

$$\hat{P}_{k-1}^{xy} = A_{k-1} \hat{P}_{k-1}^{xx} C_{k-1}^\top, \quad (3.17)$$

$$\hat{P}_{k-1}^{yy} = C_{k-1} \hat{P}_{k-1}^{xx} C_{k-1}^\top + R_{k-1}, \quad (3.18)$$

from which we extract the optimal matrices

$$\check{T}_k = \check{\Psi}_{(:,1:n),k}, \quad (3.19)$$

$$\check{N}_k = \check{\Psi}_{(:,n+1:n+m),k}, \quad (3.20)$$

$$\check{K}_{k-1} = \check{\Psi}_{(:,n+m+1:n+2m),k}. \quad (3.21)$$

Proof. According to Proposition 2.5.1, calculating the partial derivative of J_k^{eqmv} in (3.10) with respect to T_k , N_k , K_{k-1} , and Λ_k implies

$$\frac{\partial J_k^{\text{eqmv}}}{\partial T_k} = 2T_k \hat{P}_{k|k-1}^{xx} - 2K_{k-1} C_{k-1} \hat{P}_{k-1}^{xx} A_{k-1}^\top + \Lambda_k E_k^\top, \quad (3.22)$$

$$\frac{\partial J_k^{\text{eqmv}}}{\partial N_k} = 2N_k R_k + \Lambda_k C_k^\top, \quad (3.23)$$

$$\frac{\partial J_k^{\text{eqmv}}}{\partial K_{k-1}} = 2K_{k-1} \hat{P}_{k-1}^{yy} - 2T_k A_{k-1} \hat{P}_{k-1}^{xx} C_{k-1}^\top, \quad (3.24)$$

$$\frac{\partial J_k^{\text{eqmv}}}{\partial \Lambda_k} = T_k E_k + N_k C_k - I_n, \quad (3.25)$$

where $P_{k|k-1}^{xx}$ and P_{k-1}^{yy} are given by (3.16) and (3.18), respectively. Doing each partial derivative equals to zero yields the following result written as a linear equation system:

$$\Psi_k \Sigma_k + \Lambda_k \Pi_k^\top = 0_{n \times (n+2m)}, \quad (3.26)$$

$$\Psi_k \Pi_k = I_n, \quad (3.27)$$

where Ψ_k , Π_k , and Σ_k are given by (3.11), (3.14), and (3.15), respectively. We can still rewrite Σ_k as follows:

$$\Sigma_k = \Xi_k + \Gamma_k,$$

where

$$\Xi_k = 2 \begin{bmatrix} Q_{k-1} & 0_{n \times m} & 0_{n \times m} \\ 0_{m \times n} & R_k & 0_{m \times m} \\ 0_{m \times n} & 0_{m \times m} & R_{k-1} \end{bmatrix}, \quad \Gamma_k = 2 \begin{bmatrix} A_{k-1} \\ 0_{m \times n} \\ -C_{k-1} \end{bmatrix} \hat{P}_{k-1}^{xx} \begin{bmatrix} A_{k-1} \\ 0_{m \times n} \\ -C_{k-1} \end{bmatrix}^\top.$$

As $\Xi_k > 0_{(n+2m) \times (n+2m)}$ and $\Gamma_k \geq 0_{(n+2m) \times (n+2m)}$, $\Sigma_k > 0_{(n+2m) \times (n+2m)}$. In other words, Σ_k is invertible. Thus, from (3.26), $\check{\Psi}_k$ is uniquely given by (3.13). After substituting this result in (3.27),

we obtain $\check{\Lambda}_k(-\Pi_k^\top \Sigma_k^{-1} \Pi_k) = \mathbf{I}_n$. Since Π_k has full rank, $-\Pi_k^\top \Sigma_k^{-1} \Pi_k$ is invertible. Then, $\check{\Lambda}_k$ is uniquely determined by (3.12). Thereby, we obtain the desired matrices in (3.19)-(3.21). ■

As well as the KP and the KF (Kalman, 1960), the DGF employs the minimum-trace criterion to compute design matrices. Since the trace of a covariance matrix is optimally minimized in the three algorithms, these are called minimum variance. However, each algorithm solves a different optimization problem, resulting in different mean and covariance estimates. In comparison with the KP (Kalman, 1960), the optimization problem solved in Proposition 3.1.1 contains extra information of linear equality constraints, namely: state constraint (embedded in the descriptor representation) and consistency equation (related to the degrees of freedom T_k and N_k in (3.3)). Then, given that $N_k = \mathbf{0}_{n \times m}$ is an unlikely solution, the filter here proposed may return more precise solutions than the KP. Therefore, the KP can be realized as a *feasible* solution, but not optimal, of the DGF. To support this statement in a formal way, the reader is referred to Hu et al. (2022), which uses entropy as measure to denote that more constraint information used by the filter implies reduction of uncertainty of the posterior PDF.

Next, we present the DGF in Algorithm 3.1, which solves Problem 1.2.1 for descriptor systems, therefore enforcing state linear equality constraints. Its complexity order is given by

$$O(\text{DGF}) = O((n+m)^3 + n^2 p).$$

According to Subsection A.1.2, $O(\text{DGF}) > O(\text{KF})$.

Algorithm 3.1: $\hat{X}_k = \text{DGF}(\hat{X}_{k-1}, E_k, A_{k-1}, B_{k-1}, W_{k-1}, C_{k-1}, C_k, V_{k-1}, V_k, u_{k-1}, y_{k-1}, y_k)$

- 1 Compute the matrices (3.14)-(3.18) to obtain the optimal matrices $\check{\Lambda}_k$ in (3.12) and $\check{\Psi}_k$ in (3.13)
 - 2 Extract the optimal design matrices $\check{T}_k$, $\check{N}_k$, and $\check{K}_{k-1}$ using (3.19)-(3.21)
 - 3 Determine the estimates of mean $\hat{x}_k$ in (3.5) and covariance $\hat{P}_k^{\text{xx}}$ in (3.9) to obtain $\hat{X}_k \sim \mathcal{N}(\hat{x}_k, \hat{P}_k^{\text{xx}})$
-

Remark 3.1.2. In this thesis, we assume that the terms x_0 , w_{k-1} , and v_k are uncorrelated. In several problems, it is reasonable to expect that w_{k-1} and v_k are uncorrelated. However, when there exists correlation between w_{k-1} and v_k , the state error $e_k = x_k - \hat{x}_k$ may become correlated to the noises over $k \in \mathbb{Z}_+$. Therefore, some methodologies are addressed next to accord with the correlations. To the predictors of Kalman (1960), Combastel (2015), Combastel (2015) and Combastel (2016) that use y_{k-1} , the unique existing correlation is between the noises w_{k-1} and v_{k-1} , and this is readily treated by joining the common noise terms in the estimator equations. Such predictors are revisited in Appendices A-C. The filter version of the prior algorithms considers y_k , and because v_k is embedded in e_k , e_k

becomes correlated to w_k at forward steps; this correlation can be removed by defining extra vectors as in [Pan et al. \(2022\)](#) or by augmenting the state vector x_k as indicated in [Kalman \(1960\)](#) and [Lefebvre* et al. \(2004\)](#). Such filters are presented over Chapters 3-5. Finally, filters as DGF and DTZF that use both y_{k-1} and y_k can incorporate the aforementioned strategies to treat the correlations among e_k , w_k , and v_k .

3.1.2 Minimum H_2 Norm Gaussian Filter for Descriptor Systems

Here, we propose the minimum H_2 norm Gaussian filter for descriptor LTI systems (DH₂GF). This algorithm corresponds to the steady-state version of the DGF. It means that we can anticipate the constant design matrices $\check{T}$, $\check{N}$, and $\check{K}$ given by (3.19)-(3.21). The main advantage of the DH₂GF in comparison with the DGF is to keep the steady-state variance measure with a smaller processing time, since no matrix inversion is required. The state-space version ($E = I_n$) of the DH₂GF, abbreviated by H₂GF, was published in [de Paula et al. \(2020\)](#).

Regarding the estimator structure, the H₂GF is close to the minimum H_∞ norm zonotopic filter for LTI systems (H_∞ZF) ([Tang et al., 2019a](#)) (see Subsection B.1.3). The main difference is the norm employed to develop the algorithm. For the stochastic framework, the H_2 norm of system corresponds to the steady-state variance measure; see Lemma 2.5.4 for details. Then, minimizing such norm means minimizing the steady-state variance. To achieve that, we assume that the stochastic processes related to W_{k-1} and V_k are white noise.

Consider the LTI version of (3.1)-(3.2) given by:

$$Ex_k = Ax_{k-1} + Bu_{k-1} + w_{k-1}, \quad (3.28)$$

$$y_k = Cx_k + v_k, \quad (3.29)$$

where $E \in \mathbb{R}^{n \times n}$, $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times p}$, and $C \in \mathbb{R}^{m \times n}$ are known constant matrices. The process and measurement noises are expressed in terms of their linear mapping, that is, $w_{k-1} = F^w v_{k-1}^w$ and $v_k = F^v v_k^v$, where $F^w \in \mathbb{R}^{n \times n}$ and $F^v \in \mathbb{R}^{m \times m}$ define covariance matrices $Q = F^w (F^w)^\top$ and $R = F^v (F^v)^\top$ of constant GRVs $W \sim \mathcal{N}(0_{n \times 1}, Q)$ and $V \sim \mathcal{N}(0_{m \times 1}, R)$, respectively, and $v_{k-1}^w \in \mathbb{R}^n$ and $v_k^v \in \mathbb{R}^m$ are realizations from normalized GRVs with zero mean and identity covariance. This is necessary to impart the influence of covariance matrices to H_2 norm.

For the system (3.28)-(3.29), (3.3) is reduced to

$$TE + NC = I_n. \quad (3.30)$$

Following similar steps to (3.5)-(3.9), we propose the center estimate

$$\hat{x}_k = (TA - KC) \hat{x}_{k-1} + TBu_{k-1} + Ky_{k-1} + Ny_k \quad (3.31)$$

with estimation error

$$\hat{x}_k^g = (TA - KC)\hat{x}_{k-1}^g + TF^w v_{k-1}^w - KF^v v_{k-1}^v - NF^v v_k^v \quad (3.32)$$

and associated covariance matrix

$$\hat{P}_k^{xx} = (TA - KC)\hat{P}_{k-1}^{xx}(TA - KC)^\top + TQT^\top + KRK^\top + NRN^\top. \quad (3.33)$$

By convenience, we rewrite (3.32) as

$$\hat{x}_k^g = \tilde{A}\hat{x}_{k-1}^g + \tilde{B}d_{k-1}, \quad (3.34)$$

where

$$\tilde{A} = TA - KC, \quad (3.35)$$

$$\tilde{B} = \begin{bmatrix} TF^w & -KF^v & -NF^v \end{bmatrix}, \quad (3.36)$$

$$d_{k-1} = \begin{bmatrix} (v_{k-1}^w)^\top & (v_{k-1}^v)^\top & (v_k^v)^\top \end{bmatrix}^\top. \quad (3.37)$$

According to (3.34), we formulate a constrained optimization problem in Theorem 3.1.1 to specify the matrices T , N , and K . To support the developments, we apply Lemma 2.5.1 to (3.30) to rewrite the matrices T and N in function of a matrix $S \in \mathbb{R}^{n \times (n+m)}$ as follows:

$$T = \Theta^\dagger \alpha_1 + S\Psi\alpha_1, \quad (3.38)$$

$$N = \Theta^\dagger \alpha_2 + S\Psi\alpha_2, \quad (3.39)$$

where

$$\Theta = \begin{bmatrix} E \\ C \end{bmatrix}, \quad \Theta^\dagger = (\Theta^\top \Theta)^{-1} \Theta^\top, \quad \alpha_1 = \begin{bmatrix} I_n \\ 0_{m \times n} \end{bmatrix}, \quad \alpha_2 = \begin{bmatrix} 0_{n \times m} \\ I_m \end{bmatrix}, \quad \Psi = I_{(n+m)} - \Theta\Theta^\dagger. \quad (3.40)$$

Theorem 3.1.1. Consider a parameter $\psi > 0$ and matrices $P \in \mathbb{R}^{n \times n}$, $Y_1 \in \mathbb{R}^{n \times (n+m)}$, $Y_2 \in \mathbb{R}^{n \times m}$, $Y_3 \in \mathbb{R}^{(n+m) \times (n+m)}$, and $Y_4 \in \mathbb{R}^{m \times m}$, where $P = P^\top > 0_{n \times n}$, $Y_3 = Y_3^\top > 0_{(n+m) \times (n+m)}$, and $Y_4 = Y_4^\top > 0_{m \times m}$. Consider also the LMIs

$$\text{tr}(\Omega_1) + \text{tr}(\Omega_2) + \text{tr}\left((F^v)^\top Y_4 F^v\right) < \psi, \quad (3.41)$$

$$\begin{bmatrix} I_n - P & \Omega_3^\top \\ \Omega_3 & -P \end{bmatrix} < 0_{2n \times 2n}, \quad (3.42)$$

$$\begin{bmatrix} -Y_3 & Y_1^\top \\ Y_1 & -P \end{bmatrix} < 0_{(2n+m) \times (2n+m)}, \quad (3.43)$$

$$\begin{bmatrix} -Y_4 & Y_2^\top \\ Y_2 & -P \end{bmatrix} < 0_{(n+m) \times (n+m)}, \quad (3.44)$$

where

$$\Omega_1 = (F^w)^\top \alpha_1^\top (\Theta^\dagger)^\top (P\Theta^\dagger \alpha_1 + Y_1 \Psi \alpha_1) F^w + (F^w)^\top \alpha_1^\top \Psi^\top (Y_1^\top \Theta^\dagger \alpha_1 + Y_3 \Psi \alpha_1) F^w, \quad (3.45)$$

$$\Omega_2 = (F^v)^\top \alpha_2^\top (\Theta^\dagger)^\top (P\Theta^\dagger \alpha_2 + Y_1 \Psi \alpha_2) F^v + (F^v)^\top \alpha_2^\top \Psi^\top (Y_1^\top \Theta^\dagger \alpha_2 + Y_3 \Psi \alpha_2) F^v, \quad (3.46)$$

$$\Omega_3 = P\Theta^\dagger \alpha_1 A + Y_1 \Psi \alpha_1 A - Y_2 C, \quad (3.47)$$

with Θ , $\Theta^\dagger$, α_1 , α_2 , and Ψ given by (3.40). By solving the constrained optimization problem

$$\min \psi \quad \text{s.t.} \quad (3.41)-(3.44), \quad (3.48)$$

the following optimal matrices are obtained:

$$\check{T} = \Theta^\dagger \alpha_1 + \check{P}^{-1} \check{Y}_1 \Psi \alpha_1, \quad (3.49)$$

$$\check{N} = \Theta^\dagger \alpha_2 + \check{P}^{-1} \check{Y}_1 \Psi \alpha_2, \quad (3.50)$$

$$\check{K} = \check{P}^{-1} \check{Y}_2. \quad (3.51)$$

Proof. First, we relate (3.34) to Lemma 2.5.2. Doing $\check{A} = \tilde{A}$, $\check{B} = \tilde{B}$, and $\check{C} = I_n$, we obtain

$$\tilde{B}^\top P \tilde{B} = \begin{bmatrix} (F^w)^\top T^\top P T F^w & -(F^w)^\top T^\top P K F^v & -(F^w)^\top T^\top P N F^v \\ -(F^v)^\top K^\top P T F^w & (F^v)^\top K^\top P K F^v & (F^v)^\top K^\top P N F^v \\ -(F^v)^\top N^\top P T F^w & (F^v)^\top N^\top P K F^v & (F^v)^\top N^\top P N F^v \end{bmatrix}, \quad (3.52)$$

$$(TA - KC)^\top P (TA - KC) - P + I_n < 0_{n \times n}. \quad (3.53)$$

Next, Schur's complement is applied to obtain

$$\begin{bmatrix} I_n - P & (TA - KC)^\top P \\ P(TA - KC) & -P \end{bmatrix} < 0_{2n \times 2n}. \quad (3.54)$$

To enforce $TE + NC = I_n$, T and N are rewritten in terms of a common matrix S as in (3.38) and (3.39). After, the nonlinear products due to PS and PK are linearized by defining the transformations $Y_1 = PS$ and $Y_2 = PK$. Since (3.38) is substituted in $P(TA - KC)$, (3.47) is obtained. Then, (3.54) becomes the LMI (3.42).

Now, define the upper bounds

$$S^\top P S = Y_1^\top P^{-1} Y_1 < Y_3, \quad (3.55)$$

$$K^\top P K = Y_2^\top P^{-1} Y_2 < Y_4. \quad (3.56)$$

Applying Schur's complement, LMIs (3.43) and (3.44) are obtained. Also, substituting

(3.38) and (3.39) in (3.52) yields a block-diagonal matrix given by

$$H = \text{blkdiag}(\Omega_1, (F^\vee)^\top Y_4 F^\vee, \Omega_2).$$

Take the trace of each matrix within H and bound their summation by ψ . By defining $\psi > 0$ to be minimized, the constrained optimization problem is finished. ■

By solving the optimization problem (3.48), we guarantee that the eigenvalues of the matrix $\tilde{A}$ in (3.35) have module smaller than one. Then, Property 3.1.1 is also applicable to (3.31).

Next, we present the DH₂GF in Algorithm 3.2, which solves Problem 1.2.1 for descriptor systems, therefore enforcing state linear equality constraints. This algorithm assumes that Theorem 3.1.1 is offline employed, since the resulting matrices are constant. Its complexity order is given by

$$O(\text{DH}_2\text{GF}) = O(n^3 + n^2p + n^2m + nm^2).$$

In comparison with the stationary version of KP (SKP) and KF (SKF) discussed in Subsection A.1.3, note that $O(\text{DH}_2\text{GF}) > O(\text{SKF})$ and $O(\text{SKF}) = O(\text{SKP})$. However, by setting $\tilde{D} = TB$ in DH₂GF, the same complexities are achieved.

Algorithm 3.2: $\hat{X}_k = \text{DH}_2\text{GF}(\hat{X}_{k-1}, \check{T}, \check{N}, \check{K}, E, A, B, W, C, V, u_{k-1}, y_{k-1}, y_k)$

1 Compute the estimates of mean $\hat{x}_k$ in (3.31) and covariance $\hat{P}_k^{\text{xx}}$ in (3.33) to obtain $\hat{X}_k \sim \mathcal{N}(\hat{x}_k, \hat{P}_k^{\text{xx}})$

Remark 3.1.3. For LTI systems, the steady state ($k \rightarrow \infty$) of both DGF and DH₂GF is marked by the following fixed matrices: $\check{T}_\infty$, $\check{N}_\infty$, $\check{K}_\infty$, and $\hat{P}_\infty^{\text{xx}}$. These constant matrices can be anticipated as follows. First, Theorem 3.1.1 is employed to yield the steady-state design matrices $\check{T}_\infty$, $\check{N}_\infty$, and $\check{K}_\infty$. After, the steady-state covariance matrix $\hat{P}_\infty^{\text{xx}}$ is obtained by solving the discrete Lyapunov equation $\tilde{A}\hat{P}_\infty^{\text{xx}}\tilde{A}^\top - \hat{P}_\infty^{\text{xx}} + M = 0_{n \times n}$, where $\tilde{A} = (\check{T}_\infty A - \check{K}_\infty C)$ and $M = (\check{T}_\infty Q \check{T}_\infty^\top + \check{N}_\infty R \check{N}_\infty^\top + \check{K}_\infty R \check{K}_\infty^\top)$. The knowledge of $\hat{P}_\infty^{\text{xx}}$ can be used to reduce computational cost since only the mean estimates given by (3.5) vary at steady state.

3.1.3 Numerical Results

Now, we present an example to illustrate the usage of KP, KF, DGF, and their special case, that is, SKP, SKF, and H₂GF, respectively. In order to compare the performance of the algorithms, three indexes are employed, namely: (i) the *mean processing time* (T^{CPU}), given by

$$T^{\text{CPU}} \triangleq \frac{1}{m_s} \frac{1}{k_f} \sum_{j=1}^{m_s} \sum_{k=1}^{k_f} t_{k,j}, \quad (3.57)$$

where $k_f \in \mathbb{Z}_+$ is the number of time steps, $m_s \in \mathbb{Z}_+$ is the number of Monte Carlo simulations, and $t_{k,j}$ is the time interval to execute the k th iteration of a given algorithm in the j th Monte Carlo simulation; (ii) the *root mean square error* of the mean estimate of the i th state (RMSE_i), given by

$$\text{RMSE}_i \triangleq \frac{1}{m_s} \sum_{j=1}^{m_s} \sqrt{\frac{1}{(k_f - k_0 + 1)} \sum_{k=k_0}^{k_f} (x_{i,k,j} - \hat{x}_{i,k,j})^2}, \quad i = 1, \dots, n, \quad (3.58)$$

where n is the state vector dimension, and $k_0 \in \mathbb{Z}_+$ is defined to disregard the initialization effect; and (iii) the *mean trace* (MT), given by

$$\text{MT} \triangleq \frac{1}{m_s} \frac{1}{k_f} \sum_{j=1}^{m_s} \sum_{k=1}^{k_f} \text{tr}(\hat{P}_{k,j}^{\text{xx}}). \quad (3.59)$$

The simulations were performed in MATLAB 9.11 (Release) with the parameters $k_f = 60$ and $m_s = 1 \times 10^5$. Since both stationary-gain and varying-gain algorithms are compared, there exist differences during the transient to be taken into account, then we set $k_0 = 1$. The following computer configuration was used: 8 GB RAM 1333 MHz, Windows 10 Pro, and AMD FX-6300 CPU 3.50 GHz.

Consider a state-space LTI system with the form (3.28)-(3.29), whose parameters are defined as follows (Tang et al., 2019a):

$$A = \begin{bmatrix} -0.54 & 0.45 & 0.36 & 0 & 0 \\ 0.63 & 0.45 & 0.18 & 0.36 & 0 \\ 0.09 & 0.45 & 0.27 & 0.09 & 0.18 \\ 0 & 0 & 0.25 & 0.25\sqrt{2} & -0.25\sqrt{2} \\ 0 & 0 & 0 & 0.25\sqrt{2} & 0.25\sqrt{2} \end{bmatrix}, \quad B = 0_{5 \times 1}, \quad F^w = \begin{bmatrix} -1 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \quad (3.60)$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}, \quad F^v = 0.1 \times I_2. \quad (3.61)$$

It is simulated with initial state $x_0 = [4 \quad -4 \quad 4 \quad -4 \quad 4]^T$. The noise realizations w_{k-1} and v_k take values from the GRVs $W \sim \mathcal{N}(0_{5 \times 1}, F^w (F^w)^T)$ and $V \sim \mathcal{N}(0_{2 \times 1}, F^v (F^v)^T)$, respectively.

To estimate states, we additionally set $\hat{x}_0 = 0_{5 \times 1}$ and $\hat{P}_0^{\text{xx}} = \frac{25}{\varsigma} I_5$, with $\varsigma = 18.2051$ being the greatest realization of the chi-square variable, with five degrees of freedom, whose associated confidence ellipsoid encloses the true states x_0 with 99.73% confidence level.

Table 3.2: Numerical results from the state estimation of the system (3.60)-(3.61). The processing time to solve the optimization problems (3.48), (A.29) and (A.37) was disregarded.

Estimators	$T^{\text{CPU}} \times 10^{-4} (\text{s})$	MT	RMSE ₁	RMSE ₂	RMSE ₃	RMSE ₄	RMSE ₅	$\frac{1}{5} \sum_{i=1}^5 \text{RMSE}_i$
KP (Kalman, 1960)	0.166	2.14	1.12	0.549	0.533	0.525	1.13	0.771
SKP (Khargonekar et al., 1996)	0.0844	2.17	1.12	0.590	0.548	0.626	1.14	0.805
KF (Kalman, 1960)	0.165	0.215	0.526	0.534	0.528	0.520	0.564	0.534
SKF (Khargonekar et al., 1996)	0.107	0.273	0.526	0.530	0.527	0.554	0.677	0.563
DGF	0.802	0.212	0.526	0.548	0.539	0.520	0.580	0.543
H ₂ GF	0.113	0.233	0.526	0.536	0.534	0.576	0.577	0.550

After solving (3.48), we obtain the matrices

$$T = \begin{bmatrix} 0.0099 & 0 & 0 & 0.0005 & 0 \\ 0.0006 & 1 & 0 & -0.1588 & 0 \\ -0.0012 & 0 & 1 & 0.0036 & 0 \\ 0.0005 & 0 & 0 & 0.8691 & 0 \\ 0.9867 & 0 & 0 & 0.1384 & 1 \end{bmatrix}, \quad N = \begin{bmatrix} 0.9901 & -0.0005 \\ -0.0006 & 0.1588 \\ 0.0012 & -0.0036 \\ -0.0005 & 0.1309 \\ -0.9867 & -0.1384 \end{bmatrix}, \quad K^{\text{H}_2\text{GF}} = \begin{bmatrix} -0.0026 & 0.0007 \\ 0.2854 & 0.0909 \\ -0.0435 & 0.0509 \\ 0.1523 & 0.0726 \\ -0.4161 & 0.0705 \end{bmatrix}, \quad (3.62)$$

with the minimal H₂ norm equal to 0.1755. Also, by solving (A.29) and (A.37), the corresponding matrices are found:

$$K^{\text{SKP}} = \begin{bmatrix} -0.5343 & 0.1241 \\ 0.6234 & 0.2068 \\ -0.0878 & 0.1011 \\ 0.3479 & 0.1655 \\ -0.3480 & -0.0045 \end{bmatrix}, \quad K^{\text{SKF}} = \begin{bmatrix} 0.9901 & -0.0009 \\ -0.0010 & 0.2793 \\ 0.0023 & -0.0058 \\ -0.0009 & 0.2298 \\ -0.9834 & -0.2424 \end{bmatrix}, \quad (3.63)$$

whose minimal H₂ norm is equal to 1.4208 and 0.2015, respectively.

In Table 3.2, the computed indexes are presented. First, note that the T^{CPU} of KP, KF, and DGF is larger than the T^{CPU} of their stationary version. This is expected because the varying-gain estimators require online matrix inversions, while the fixed-gain estimators do not. Among KP, KF, and DGF, the most significant difference of T^{CPU} is realized with DGF, since it executes more matrix inversions and involves higher-order matrices. Regarding MT and RMSE, the predictors KP and SKP return the largest values. This is also expected because filters use both past and current measurements to reach more precise and accurate estimates (Teixeira, 2008).

In this example, we illustrated that the difference of MT and RMSE is not significant among the pairs of related algorithms and that the stationary versions return large result. This is expected as in Teixeira et al. (2009) because the state estimates satisfy the dynamical equations. Both DGF and DH₂GF are actually motivated to descriptor systems, since the other algorithms cannot properly enforcing state linear equality constraints. This will be exploited afterwards in Subsection 4.3.3. Anyway, we showed that DGF and DH₂GF

are more accurate and precise than their predictor version (KP and SKP), illustrating the effectiveness of our proposals.

3.1.4 Minimum-Radius Zonotopic Filter for Descriptor Systems

Here, we propose the minimum-radius zonotopic filter for descriptor LTV systems (DRZF). This algorithm presents a similar structure to the DTZF (see Subsection B.1.2), with the difference being a new cost criterion, from which we aim at effectively minimizing the radius of zonotopes. The proposed minimum-radius criterion is not equivalent to the exact volume approach, but it directly influences the volume measure since the considered box is the so-called interval hull of the zonotope. The resulting optimization problem is nonlinear, but we rewrite it as an LP to enable the use of linear solvers. The DRZF is presented in de Paula et al. (2022b).

Consider the discrete-time descriptor LTV uncertain dynamical system given by (3.1)-(3.2). Assume that the terms w_{k-1} , v_k , and x_0 are bounded by the known zonotopes $\mathcal{W}_{k-1} = \{G_{k-1}^w, 0_{n \times 1}\}$, $\mathcal{V}_k = \{G_k^v, 0_{m \times 1}\}$, and $\hat{\mathcal{X}}_0 = \{\hat{G}_0^x, \hat{x}_0\}$, with $G_{k-1}^w \in \mathbb{R}^{n \times n_g^w}$, $G_k^v \in \mathbb{R}^{m \times n_g^v}$, and $\hat{G}_0^x \in \mathbb{R}^{n \times n_g}$. No rank assumption is required for the generator matrices. Since $\text{rank}(E_k) \leq n$, the descriptor representation (3.1) already embeds the state linear equality constraints to x_k .

Following similar steps to (3.5)-(3.6), we propose the center estimate

$$\hat{x}_k = (T_k A_{k-1} - K_{k-1} C_{k-1}) \hat{x}_{k-1} + T_k B_{k-1} u_{k-1} + K_{k-1} y_{k-1} + N_k y_k \quad (3.64)$$

with estimation error

$$\begin{aligned} e_k &\triangleq x_k - \hat{x}_k \\ &= (T_k A_{k-1} - K_{k-1} C_{k-1}) e_{k-1} + T_k w_{k-1} - K_{k-1} v_{k-1} - N_k v_k \\ &= (T_k A_{k-1} - K_{k-1} C_{k-1}) \hat{x}_{k-1}^z + T_k w_{k-1} - K_{k-1} v_{k-1} - N_k v_k \\ &\in (T_k A_{k-1} - K_{k-1} C_{k-1}) \hat{\mathcal{X}}_{k-1}^z \oplus T_k \mathcal{W}_{k-1} \oplus (-K_{k-1}) \mathcal{V}_{k-1} \oplus (-N_k) \mathcal{V}_k \\ &= \hat{\mathcal{X}}_k^z \\ &= \{\hat{G}_k^x, 0_{n \times 1}\}. \end{aligned}$$

Although the center of $\hat{\mathcal{X}}_k^z$ is assumed to be zero, its true value is treated as an uncertainty, justifying the change of variable performed. From Properties 2.2.2 and 2.2.3, the generator matrix $\hat{G}_k^x$ is given by

$$\hat{G}_k^x = \begin{bmatrix} (T_k A_{k-1} - K_{k-1} C_{k-1}) \hat{G}_{k-1}^x & T_k G_{k-1}^w & -K_{k-1} G_{k-1}^v & -N_k G_k^v \end{bmatrix}, \quad (3.65)$$

with $G_{k-1}^v \in \mathbb{R}^{m \times n_g^{k-1}}$. As in Example 3.1.1, we next illustrate that the center deviation $(\bar{x}_k - \hat{x}_k)$ implies $\hat{\mathcal{X}}_k^z \supset \mathcal{X}_k^z$. Although *bias* is a stochastic concept, Property 3.1.1 is also

applicable to (3.64) by assuming that x_0 , w_{k-1} , v_{k-1} , and v_k are random terms defined in zonotopes. In this case, centers play the role of means.

Example 3.1.2. Consider that the zonotope $\mathcal{X} = \{2, 3\}$ will be approximated by the zonotope $\hat{\mathcal{X}} = \{\hat{G}^x, \hat{x}\}$. Assume that $\hat{x} = 2$ is given, and that the goal is to find the smallest set such that $\mathcal{X} \subseteq \hat{\mathcal{X}}$.

To solve this problem, the bounds of $\hat{\mathcal{X}}$ must coincide with or dominate the bounds of $\mathcal{X} = [1, 5]$. For $\hat{x} = 2$, there are two candidate solutions: (i) $\hat{\mathcal{X}} = [1, 3]$ and (ii) $\hat{\mathcal{X}} = [-1, 5]$. Since $[-1, 5] \supset [1, 5]$, the desired solution is $\hat{G}^x = 3$.

Our goal is to determine the design matrices T_k , N_k , and K_{k-1} , for $k \in \mathbb{Z}_+$, by minimizing the radius of the interval hull of $\hat{\mathcal{X}}_k$ given by $\square \hat{\mathcal{X}}_k$. Let $F \triangleq A_{k-1} \hat{G}_{k-1}^x$ and $H \triangleq C_{k-1} \hat{G}_{k-1}^x$. According to (3.65), the i th radius of $\square \hat{\mathcal{X}}_k$, for $i = 1, \dots, n$, is given by

$$\begin{aligned} \zeta_{i,k} &\triangleq |\hat{G}_{(i,:),k}^x| 1_{\left(n_g + n_g^w + n_{g,k-1}^v + n_{g,k}^v\right) \times 1} \\ &= \sum_{j=1}^{n_g} |T_{(i,:),k} F_{:,j} - K_{(i,:),k-1} H_{:,j}| + \sum_{j=1}^{n_g^w} |T_{(i,:),k} G_{:,j}^w| + \sum_{j=1}^{n_{g,k-1}^v} |-K_{(i,:),k-1} G_{(i,:),k-1}^v| + \sum_{j=1}^{n_{g,k}^v} |-N_{(i,:),k} G_{(i,:),k}^v|, \end{aligned} \quad (3.66)$$

with $T_{(i,:),k} E_k + N_{(i,:),k} C_k = (I_n)_{i,:}$ being the corresponding linear equality constraint (3.3). Thus, the minimization of ζ_k consists of n constrained optimization problems, from which the matrices T_k , N_k , and K_{k-1} are optimally specified. In principle, due to the fact that $\zeta_{i,k}$ is nonlinear, the prior problem should be solved using nonlinear solvers. However, by employing some procedures presented in Bertsimas & Tsitsiklis (1997, Section 1.3), we transform the original optimization problem in an LP. In doing so, efficient solvers such as CPLEX and Gurobi can be used. The derived LP is presented next in Proposition 3.1.2. Recall that the possible state linear equality constraints are already embedded in the descriptor representation. Then, the linear inequality and equality constraints present in the derived LP are related only to the reformulation of the original problem.

Proposition 3.1.2. The constrained optimization problem given by the minimization of (3.66) subject to $T_{(i,:),k} E_k + N_{(i,:),k} C_k = (I_n)_{i,:}$ is equivalent to the LP

$$\min_z \psi^\top z \quad \text{s.t.} \quad D^i z \leq d^i, \quad D^e z = d^e, \quad (3.67)$$

where $z \in \mathbb{R}^{n^z}$ is the variable vector such that

$$z_{1:n}^\top \triangleq T_{(i,:),k} \quad (3.68)$$

$$z_{n+1:n+m}^\top \triangleq N_{(i,:),k}, \quad (3.69)$$

$$z_{n+m+1:n+2m}^\top \triangleq K_{(i,:),k-1}, \quad (3.70)$$

$n^z = (n + 2m + n_g + n_g^w + n_{g,k-1}^v + n_{g,k}^v)$, $\psi \in \mathbb{R}^{n^z}$ is the cost vector, $D^i \in \mathbb{R}^{n^i \times n^z}$ and $d^i \in \mathbb{R}^{n^i}$

compose the inequality constraints, $n^i = 2(n_g + n_g^w + n_{g,k-1}^v + n_{g,k}^v)$, and $D^e \in \mathbb{R}^{n \times n^z}$ and $d^e \in \mathbb{R}^n$ compose the equality constraints. These parameters are defined as

$$\psi \triangleq \begin{bmatrix} 0_{1 \times (n+2m)} & 1_{1 \times (n_g + n_g^w + n_{g,k-1}^v + n_{g,k}^v)} \end{bmatrix}^\top, \quad (3.71)$$

$$D_{1:n_g,:}^i \triangleq \begin{bmatrix} -F^\top & 0_{n_g \times m} & H^\top & -I_{n_g} & 0_{n_g \times (n_g^w + n_{g,k-1}^v + n_{g,k}^v)} \end{bmatrix}, \quad (3.72)$$

$$D_{n_g+1:2n_g,:}^i \triangleq \begin{bmatrix} F^\top & 0_{n_g \times m} & -H^\top & -I_{n_g} & 0_{n_g \times (n_g^w + n_{g,k-1}^v + n_{g,k}^v)} \end{bmatrix}, \quad (3.73)$$

$$D_{2n_g+1:2n_g+n_g^w,:}^i \triangleq \begin{bmatrix} -(G_{k-1}^w)^\top & 0_{n_g^w \times (2m+n_g)} & -I_{n_g^w} & 0_{n_g^w \times (n_{g,k-1}^v + n_{g,k}^v)} \end{bmatrix}, \quad (3.74)$$

$$D_{2n_g+n_g^w+1:2(n_g+n_g^w),:}^i \triangleq \begin{bmatrix} (G_{k-1}^w)^\top & 0_{n_g^w \times (2m+n_g)} & -I_{n_g^w} & 0_{n_g^w \times (n_{g,k-1}^v + n_{g,k}^v)} \end{bmatrix}, \quad (3.75)$$

$$D_{2(n_g+n_g^w)+1:2(n_g+n_g^w)+n_{g,k-1}^v,:}^i \triangleq \begin{bmatrix} 0_{n_{g,k-1}^v \times (n+2m)} & (G_{k-1}^v)^\top & 0_{n_{g,k-1}^v \times (n_g + n_g^w)} & -I_{n_{g,k-1}^v} & 0_{n_{g,k-1}^v \times n_{g,k}^v} \end{bmatrix}, \quad (3.76)$$

$$D_{2(n_g+n_g^w)+n_{g,k-1}^v+1:2(n_g+n_g^w)+n_{g,k-1}^v+n_{g,k}^v,:}^i \triangleq \begin{bmatrix} 0_{n_{g,k-1}^v \times (n+2m)} & -(G_{k-1}^v)^\top & 0_{n_{g,k-1}^v \times (n_g + n_g^w)} & -I_{n_{g,k-1}^v} & 0_{n_{g,k-1}^v \times n_{g,k}^v} \end{bmatrix}, \quad (3.77)$$

$$D_{2(n_g+n_g^w+n_{g,k-1}^v)+1:2(n_g+n_g^w+n_{g,k-1}^v)+n_{g,k}^v,:}^i \triangleq \begin{bmatrix} 0_{n_{g,k}^v \times n} & (G_k^v)^\top & 0_{n_{g,k}^v \times (m+n_g+n_g^w+n_{g,k-1}^v)} & -I_{n_{g,k}^v} \end{bmatrix}, \quad (3.78)$$

$$D_{2(n_g+n_g^w+n_{g,k-1}^v)+n_{g,k}^v+1:2(n_g+n_g^w+n_{g,k-1}^v)+n_{g,k}^v+n_{g,k}^v,:}^i \triangleq \begin{bmatrix} 0_{n_{g,k}^v \times n} & -(G_k^v)^\top & 0_{n_{g,k}^v \times (m+n_g+n_g^w+n_{g,k-1}^v)} & -I_{n_{g,k}^v} \end{bmatrix}, \quad (3.79)$$

$$d^i \triangleq 0_{n^i \times 1}, \quad (3.80)$$

$$D^e \triangleq \begin{bmatrix} E_k^\top & C_k^\top & 0_{n \times (m+n_g+n_g^w+n_{g,k-1}^v+n_{g,k}^v)} \end{bmatrix}, \quad (3.81)$$

$$d^e \triangleq (I_n)_{:,i}. \quad (3.82)$$

Proof. Since the objective function (3.66) is composed of piecewise linear convex functions, and the constraints (3.3) are linear, we here employ the methodology discussed in Bertsimas & Tsitsiklis (1997, Example 1.5) to replace each absolute value by a new variable and two additional linear inequality constraints. To achieve that, we first define the variable vector $z \in \mathbb{R}^{n^z}$ whose first $(n+2m)$ elements correspond the i th row of matrices T_k , N_k , and K_{k-1} , respectively, whereas its $(n_g + n_g^w + n_{g,k-1}^v + n_{g,k}^v)$ remaining elements correspond to the absolute values. Thereby, only the additional variables contribute to $\zeta_{i,k}$, yielding the cost vector ψ in (3.71).

From (3.66), we define the following inequality constraints:

$$\left. \begin{aligned} & -z_{1:n}^\top F_{:,j} + z_{n+m+1:n+2m}^\top H_{:,j} - z_{n+2m+j} \leq 0 \\ & z_{1:n}^\top F_{:,j} - z_{n+m+1:n+2m}^\top H_{:,j} - z_{n+2m+j} \leq 0 \end{aligned} \right\}, \quad j = 1, \dots, n_g,$$

$$\left. \begin{aligned} & -z_{1:n}^\top G_{(:,j),k-1}^w - z_{n+2m+n_g+j} \leq 0 \\ & z_{1:n}^\top G_{(:,j),k-1}^w - z_{n+2m+n_g+j} \leq 0 \end{aligned} \right\}, \quad j = 1, \dots, n_g^w,$$

$$\left. \begin{aligned} & z_{n+m+1:n+2m}^\top G_{(:,j),k-1}^v - z_{n+2m+n_g+n_g^w+j} \leq 0 \\ & -z_{n+m+1:n+2m}^\top G_{(:,j),k-1}^v - z_{n+2m+n_g+n_g^w+j} \leq 0 \end{aligned} \right\}, \quad j = 1, \dots, n_{g,k-1}^v,$$

$$\left. \begin{aligned} z_{n+1:n+m}^\top G_{(:,j),k}^v - z_{n+2m+n_g+n_g^w+n_g^v}^\top + j &\leq 0 \\ -z_{n+1:n+m}^\top G_{(:,j),k}^v - z_{n+2m+n_g+n_g^w+n_g^v}^\top + j &\leq 0 \end{aligned} \right\}, \quad j = 1, \dots, n_{g,k}^v.$$

After transposing the prior inequalities, we extract the matrix D^i in (3.72)-(3.79) and the vector d^i in (3.80). Finally, by transposing the equality constraint $z_{1:n}^\top E_k + z_{n+1:n+m}^\top C_k = (I_n)_{i,:}$, we obtain the matrix D^e in (3.81) and the vector d^e in (3.82). ■

Next, we present in Algorithm 3.3 the steps to execute a loop of state estimation with DRZF. As usual in zonotopic filtering, we add an order-reduction step to limit the number of generators n_g of $\hat{G}_{k-1}^x$ in φ_g . This algorithm solves Problem 1.2.3 enforcing state linear equality constraints. Assuming that LPs are solved with the same computational efficiency than Kelner & Spielman (2006), the complexity order of DRZF is presented in Table 3.3.

Algorithm 3.3: $\hat{X}_k = \text{DRZF}(\hat{X}_{k-1}, E_k, A_{k-1}, B_{k-1}, C_{k-1}, C_k, \mathcal{W}_{k-1}, \mathcal{V}_{k-1}, \mathcal{V}_k, u_{k-1}, y_{k-1}, y_k, \varphi_g)$

Initialization: $\check{T}_k \in \mathbb{R}^{n \times n}$, $\check{N}_k \in \mathbb{R}^{n \times m}$, $\check{K}_{k-1} \in \mathbb{R}^{n \times m}$

- 1 Compute $\hat{X}_{k-1}^\downarrow = \text{ZR}(\hat{X}_{k-1}, \varphi_g)$, where ZR is discussed in Subsection B.1.1.1
 - 2 Define $F = A_{k-1} \hat{G}_{k-1}^x$, $H = C_{k-1} \hat{G}_{k-1}^x$, ψ in (3.71), D^i in (3.72)-(3.79), d^i in (3.80), and D^e in (3.81)
 - 3 **for** $i = 1, \dots, n$ **do**
 - 4 Define d^e in (3.82) and solve the LP (3.67) to obtain the optimal vector $\check{z}$
 - 5 Do $\check{T}_{(i,:),k} = \check{z}_{1:n}^\top$, $\check{N}_{(i,:),k} = \check{z}_{n+1:n+m}^\top$, and $\check{K}_{(i,:),k-1} = \check{z}_{n+m+1:n+2m}^\top$
 - 6 **end**
 - 7 Calculate the generator matrix $\hat{G}_k^x$ in (3.65) and the center $\hat{x}_k$ in (3.64) to obtain the estimated zonotope $\hat{X}_k = \{\hat{G}_k^x, \hat{x}_k\}$
-

3.1.5 Numerical Examples

In this subsection, we experiment the DRZF algorithm in two three-state numerical examples. In order to fairly compare it with literature results, we implement two algorithms that use a similar observer structure to DRZF. The first one is the H_∞ ZF described in Subsection B.1.3, which is here modified with the matrix $\Theta = \begin{bmatrix} E \\ C \end{bmatrix}$ to enable the treatment for descriptor systems; the resulting algorithm is called DH_∞ ZF. The second one is the DTZF. These algorithms are executed in different examples. For comparison purposes, we compute two performance indexes, namely: (i) the *average area ratio of box* ($A^\square$), given by

$$A^\square \triangleq \frac{1}{m_s} \frac{1}{k_f} \sum_{j=1}^{m_s} \sum_{k=1}^{k_f} \prod_{i=1}^n \text{diam}([x]_{i,k,j}), \quad (3.83)$$

Table 3.3: Complexity $O(\cdot)$ of DRZF.

Step	$O(\cdot)$
1	$n_g(n + \log(n_g))$
2	$n^2\varphi_g + mn\varphi_g + \varphi_g^2 + nn_g^w + (n_g^w)^2 + mn_{g,k-1}^v + (n_{g,k-1}^v)^2 + mn_{g,k}^v + (n_{g,k}^v)^2$
3-6	$n(n + m + \varphi_g + n_g^w + n_{g,k-1}^v + n_{g,k}^v)(n + \varphi_g + n_g^w + n_{g,k-1}^v + n_{g,k}^v)^3$
7	$n^2(n + p + m + \varphi_g + n_g^w) + nm(n_{g,k-1}^v + n_{g,k}^v)$
Total	$n_g(n + \log(n_g)) + n^2p + n(n + m + \varphi_g + n_g^w + n_{g,k-1}^v + n_{g,k}^v)(n + \varphi_g + n_g^w + n_{g,k-1}^v + n_{g,k}^v)^3$

with the time horizon $k_f = 80$ and the number of Monte Carlo simulations $m_s = 1000$; and (ii) the *average largest radius ratio of box* ($r^\square$), given by

$$r^\square \triangleq \frac{1}{m_s} \frac{1}{k_f} \sum_{j=1}^{m_s} \sum_{k=1}^{k_f} \max_i \text{rad}(\square \hat{\mathcal{X}}_{i,k,j}). \quad (3.84)$$

The reduction order $\varphi_g = 4$ is set to become the state estimation more challenging.

The following computer configuration was used: 8 GB RAM 1333 MHz, Windows 10 Pro, and AMD FX-6300 CPU 3.50 GHz. All implementations were executed in MATLAB 9.11 (Release) with Gurobi 9.1 and MPT3 (Herceg et al., 2013). Specifically, Gurobi is used to solve LPs while MPT is employed to sketch sets.

3.1.5.1 LTI System

Consider an LTI descriptor system (Rego et al., 2020b) of the form (3.1)-(3.2), where

$$E = \text{diag}([1 \quad 1 \quad 0]^\top), \quad A = \begin{bmatrix} 0.5 & 0 & 0 \\ 0.8 & 0.95 & 0 \\ -1 & 0.5 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 & 1 \\ 1 & -1 & 0 \end{bmatrix},$$

$G^w = \text{diag}([0.1 \quad 1.5 \quad 0.6]^\top)$, and $G^v = \text{diag}([0.5 \quad 1.5]^\top)$, where $\text{diag}(\cdot)$ returns a diagonal matrix from a vector. The elements w_{k-1} and v_k are taken from uniform distributions, while $u_k = \cos(0.15k) \mathbf{1}_{2 \times 1}$. The simulations are executed with $x_0 = [0.5 \quad 0.5 \quad 0.25]^\top \in \hat{\mathcal{X}}_0 = \{\mathbf{I}_3, \mathbf{0}_{3 \times 1}\}$.

In Figure 3.1, we illustrate the guaranteed state estimation using the $\text{DH}_\infty\text{ZF}$ and DRZF algorithms in one separate simulation; interval hulls are sketched by computational simplicity. As expected, DRZF returns the smallest sets, mainly during the transient, since the design matrices T_k , N_k , K_{k-1} are computed online and since the minimum-radius criterion aims at reducing $\square \hat{\mathcal{X}}_k$. Then, as shown in Table 3.4, DRZF returns better precision (smaller values of $A^\square$ and $r^\square$).

In order to briefly illustrate the DRZF together with a set-membership filter, we also implemented the algorithm of Rego et al. (2020b). This algorithm properly addresses the state-estimation problem with CZs for descriptor systems. However, it requires the

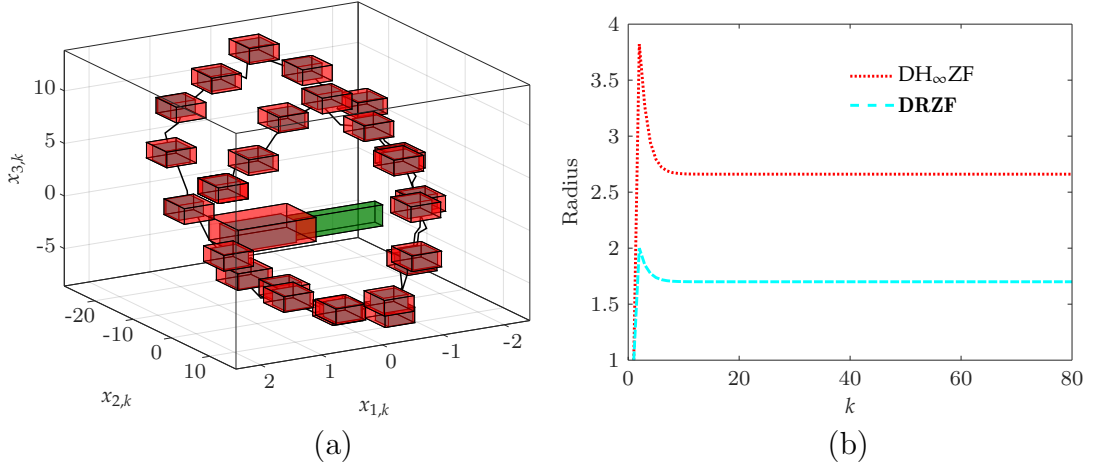


Figure 3.1: Graph (a): Time evolution of states (black —) and interval hulls computed by $\text{DH}_\infty\text{ZF}$ (red box) and DRZF (cyan box). Green box is the initial set $\hat{\mathcal{X}}_0$. Graph (b): Time evolution of the largest radius of the prior boxes.

Table 3.4: Results of $A^\square$ and $r^\square$ for the system of Subsection 3.1.5.1.

Algorithms	$A^\square$	$r^\square$
$\text{DH}_\infty\text{ZF}$	4.27	2.67
DRZF	2.12 ($\downarrow 50.4\%$)	1.70 ($\downarrow 36.3\%$)

previous knowledge of a state admissible set. For the current case study, we set the same admissible set that Rego et al. (2020b), that is, $\mathcal{X}^F = \{50\mathbf{I}_3, 0_{3 \times 1}\}$.

Many different simulations were executed with the prior algorithms, but we next depicted only two of them. In the first one, no order reduction was applied to zonotopes and CZs. In the second one, order reduction was employed with $\varphi_c = 1$ and $\varphi_g = 4$, and ZR (discussed in Subsection B.1.1.1) was used to reduce generators.

In Figure 3.2, we illustrate a common simulation involving the performed tests. According to graph (a), the precision of the method of Rego et al. (2020b) may be better than or equal to the DRZF one. This occurs because algebraic equations imply asymmetry for convex polytopes, and zonotopes cannot properly incorporate this information. Other reason is that DRZF minimizes the interval hull of zonotopes, not actually their volume. Conversely, DRZF established a tight upper bound to limit the amplitudes of CZs. This is a curious fact because, although DRZF yields zonotopes, it sometimes provides interval hulls competitive with those of CZs. Moreover, note that the initial evolution of CZs is affected by $\mathcal{X}^F$; this set is not required by DRZF.

In practice, order-reduction algorithms may be employed to control computational cost, and as depicted in graph (b), the precision of the method of Rego et al. (2020b) can be significantly deteriorated. Other reduction orders (φ_c and φ_g) and/or other generator reductions can be tested and may improve the convergence with CZs, but the dependency of u_{k-1} and y_k to compute the CZs makes each simulation be different, whose precision can get better or worse. Differently, the uncertainty of DRZF is not affected by u_{k-1} and y_k ,

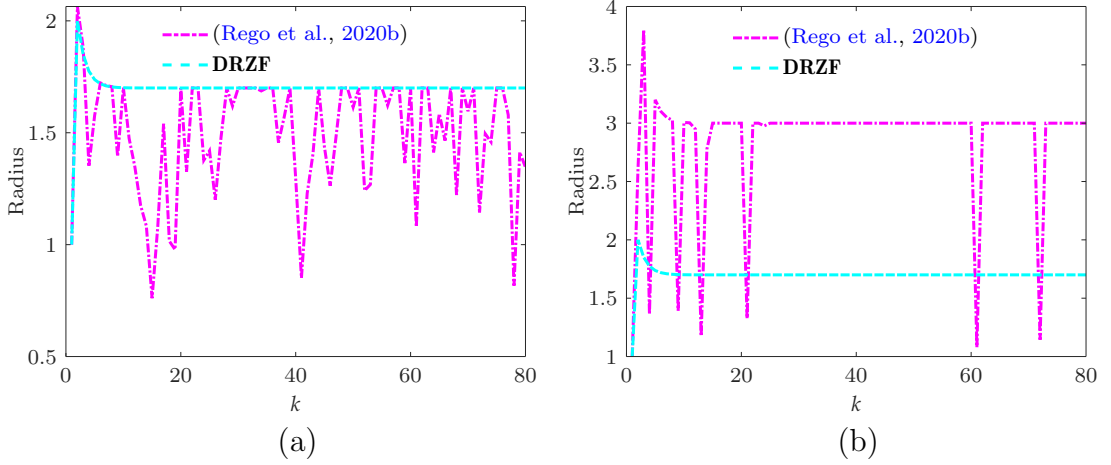


Figure 3.2: Time evolution of the largest radius of interval hulls computed by the method of Rego et al. (2020b) (magenta $- \cdot -$) and by DRZF (cyan $--$). Graph (a): no reduction order is employed. Graph (b): Order reduction is applied to sets.

Table 3.5: Results of $A^\square$ and $r^\square$ for the system of Subsection 3.1.5.2.

Algorithms	$A^\square$	$r^\square$
DTZF	0.304	0.431
DRZF	0.0453 ($\downarrow 85.1\%$)	0.257 ($\downarrow 40.4\%$)

and thereby, its precision is not significantly modified with change of order.

3.1.5.2 LTV System

Consider an LTV descriptor system (Tang et al., 2020) of the form (3.1)-(3.2), where

$$E_k = \begin{bmatrix} 1 + 0.2 \sin(0.1k) & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad A_k = \begin{bmatrix} 0.6 - 0.1 \sin(0.1k) & 0 & 0.3 \\ -0.2 \sin(0.2k) & 0.4 & 0 \\ 0 & 1 & 1 \end{bmatrix}, \quad B_k = \begin{bmatrix} 1 \\ \sin(0.1k) \\ 0 \end{bmatrix},$$

$C_k = [0 \quad 1 + 0.2 \sin(0.2k) \quad 0.5]$, $G^w = 0.04I_3$ and $G^v = 0.01$. The elements w_{k-1} and v_k are taken from uniform distributions, while $u_k = 0.75 \sin(0.15k)$. The simulations are executed with $x_0 = [0.1 \quad 0 \quad 0]^T \in \hat{X}_0 = \{0.2I_3, 0_{3 \times 1}\}$.

In Figure 3.3, we illustrate both the application of DRZF in an LTV case and the increase of precision with respect to DTZF. Both algorithms compute online design matrices, but the trace minimization (DTZF) is not as efficient as the radius minimization (DRZF) to reduce the uncertainty of zonotopes, implying more conservative results as shown in Table 3.5.

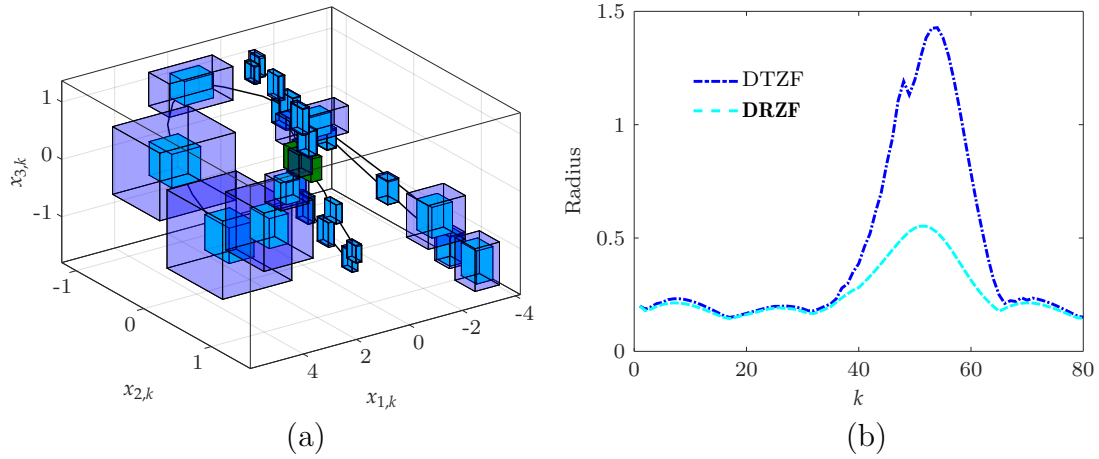


Figure 3.3: Graph (a): Time evolution of states (black —) and interval hulls computed by DTZF (blue box) and DRZF (cyan box). Green box is the common initial box. Graph (b): Time evolution of the largest radius of the prior boxes.

3.1.6 Zonotopic Kalman Filter

3.1.6.1 Time-Varying Case

Here, we propose the zonotopic Kalman filter for LTV systems (ZKF). The estimator structure of the ZKF is similar to the KF one (see Subsection A.1.2), but the uncertainties are assumed to be bounded and characterized by zonotopes. This algorithm improves the ZKP (see Subsection B.1.1) since a filter version is provided. Then, we expect to obtain better accuracy (exactness of centers) and precision (size of uncertainty) than the ZKP (Combastel, 2015). The ZKF was published in de Paula et al. (2019) as a special case of mixed-uncertainty filtering.

In comparison with the DRZF, ZKF is a faster algorithm because the minimum-trace optimization problems are explicitly solved, and for state-space systems we expect better accuracy and competitive precision. Regarding improvement of accuracy, it occurs because DRZF aims at minimizing the size of uncertainty (interval hull of zonotope), while ZKF aims at minimizing the variation of uncertainty around centers. Concerning precision, significant differences may appear between the algorithms in the presence of state equality constraints, since the covariation matrix of zonotopes for ZKF may be initialized without satisfying those constraints; this issue will be discussed in Subsection 4.3.3 with more details.

Consider the discrete-time LTV uncertain dynamical system

$$x_k = A_{k-1}x_{k-1} + B_{k-1}u_{k-1} + w_{k-1}, \quad (3.85)$$

$$y_k = C_k x_k + v_k, \quad (3.86)$$

where $A_{k-1} \in \mathbb{R}^{n \times n}$, $B_{k-1} \in \mathbb{R}^{n \times p}$, and $C_k \in \mathbb{R}^{m \times n}$ are the known system matrices, $u_{k-1} \in \mathbb{R}^p$ is the deterministic input vector, $y_k \in \mathbb{R}^m$ is the measured output vector, $w_{k-1} \in \mathbb{R}^n$ is the process

noise, $v_k \in \mathbb{R}^m$ is the measurement noise, and $x_k \in \mathbb{R}^n$ is the state vector to be estimated. The terms w_{k-1} , v_k , and x_0 are characterized by the known zonotopes $\mathcal{W}_{k-1} = \{G_{k-1}^w, 0_{n \times 1}\}$, $\mathcal{V}_k = \{G_k^v, 0_{m \times 1}\}$, and $\hat{X}_0 = \{\hat{G}_0^x, \hat{x}_0\}$, respectively, with $G_{k-1}^w \in \mathbb{R}^{n \times n_g^w}$, $G_k^v \in \mathbb{R}^{m \times n_g^v}$, and $\hat{G}_0^x \in \mathbb{R}^{n \times n_g}$. We assume that $G_k^v (G_k^v)^\top > 0_{m \times m}$.

For (3.85)-(3.86), we propose the center estimate

$$\hat{x}_k = \hat{x}_{k|k-1} + K_k (y_k - \hat{y}_{k|k-1}), \quad (3.87)$$

where

$$\hat{x}_{k|k-1} = A_{k-1} \hat{x}_{k-1} + B_{k-1} u_{k-1}, \quad (3.88)$$

$$\hat{y}_{k|k-1} = C_k \hat{x}_{k|k-1}, \quad (3.89)$$

and with $K_k \in \mathbb{R}^{n \times m}$ being the gain matrix. Unlike Subsection 3.1.1, K is here stated at k because y_k is employed.

From (3.87), we have:

$$\begin{aligned} e_k &\triangleq x_k - \hat{x}_k \\ &= (I_n - K_k C_k) (A_{k-1} e_{k-1} + w_{k-1}) - K_k v_k \\ &= (I_n - K_k C_k) (A_{k-1} \hat{x}_{k-1}^z + w_{k-1}) - K_k v_k \\ &\in (I_n - K_k C_k) (A_{k-1} \hat{X}_{k-1}^z \oplus \mathcal{W}_{k-1}) \oplus (-K_k) \mathcal{V}_k \\ &= \hat{X}_k^z \\ &= \{\hat{G}_k^x, 0_{n \times 1}\}. \end{aligned}$$

Using Properties 2.2.2 and 2.2.3, the generator matrix $\hat{G}_k^x$ is given by

$$\hat{G}_{k|k-1}^x = \begin{bmatrix} A_{k-1} \hat{G}_{k-1}^x & G_{k-1}^w \end{bmatrix}, \quad (3.90)$$

$$\hat{G}_k^x = \begin{bmatrix} (I_n - K_k C_k) \hat{G}_{k|k-1}^x & -K_k G_k^v \end{bmatrix}. \quad (3.91)$$

Both $\hat{x}_k$ and $\hat{G}_k^x$, given by (3.87) and (3.91), have been obtained by assuming a known matrix K_k . In order to optimally yield this design matrix, we propose the following theorem.

Theorem 3.1.2. Let $\hat{G}_k^x$ in (3.91) be the instance of the cost function J_k^{rad} in (1.8). Then, minimizing J_k^{rad} results in the optimal gain

$$\check{K}_k = \hat{C}_{k|k-1}^{\text{xy}} \left(\hat{C}_{k|k-1}^{\text{yy}} \right)^{-1}, \quad (3.92)$$

where

$$\hat{C}_{k|k-1}^{xx} = \hat{G}_{k|k-1}^x (\hat{G}_{k|k-1}^x)^\top, \quad (3.93)$$

$$C_k^{vv} = G_k^v (G_k^v)^\top, \quad (3.94)$$

$$\hat{C}_{k|k-1}^{xy} = \hat{C}_{k|k-1}^{xx} C_k^\top, \quad (3.95)$$

$$\hat{C}_{k|k-1}^{yy} = C_k \hat{C}_{k|k-1}^{xx} C_k^\top + C_k^{vv}. \quad (3.96)$$

Proof. According to Proposition 2.5.1, carrying out $\frac{\partial \text{tr}(\hat{C}_k^x (\hat{C}_k^x)^\top)}{\partial K_k} = 0$ yields

$$\hat{C}_{k|k-1}^{xx} C_k^\top = K_k \hat{C}_{k|k-1}^{yy},$$

where $\hat{C}_{k|k-1}^{xx}$ and $\hat{C}_{k|k-1}^{yy}$ are given by (3.93) and (3.96), respectively. Since $C_k^{vv} > 0_{m \times m}$, we guarantee the inversion of $\hat{C}_{k|k-1}^{yy}$, and thereby, the optimal gain (3.92) is obtained. ■

By assuming that x_0 , w_{k-1} , and v_k are white noise realizations defined in zonotopes, a similar result to Property 3.1.1 is next derived to the ZKF.

Property 3.1.2. Let $E[e_k] = (I_n - K_k C_k)(A_{k-1} E[e_{k-1}] + E[w_{k-1}]) - K_k E[v_k]$ be the bias of (3.87). Assume that the eigenvalues of $(I_n - K_k C_k) A_{k-1}$ have module smaller than one. Then, since $E[w_{k-1}] = 0_{n \times 1}$ and $E[v_k] = 0_{m \times 1}$, (3.87) is asymptotically unbiased.

Proof. This proof is similar to the proof of Property 3.1.1, with the difference being that the system matrix is $(I_n - K_k C_k) A_{k-1}$, and centers play the role of means. ■

Note that the number of generators for $\hat{G}_k^x$ in (3.91) is larger than the initial value n_g . This order growth is originated from Property 2.2.3 and implies an increasing demand of computational resources. In order to limit the order of (3.91) in a desired value φ_g , we apply the ZR to $\hat{G}_{k-1}^x$; see Subsection B.1.1.1 for details.

Next, we describe the ZKF in Algorithm 3.4, which solves Problem 1.2.3 without enforcing state constraints. Its complexity order is given by

$$O(\text{ZKF}) = O(n_g (n + \log(n_g)) + n(p + mn_g^v) + n^2(n + n_g^w + m + \varphi_g) + m^2(n + m + n_g^v)).$$

With respect to ZKP (see Subsection B.1.1), $H_\infty\text{ZF}$ (Subsection B.1.3), and ZKF, we have the following complexity analyses:

$$\begin{aligned} O(\text{ZKF}) - O(\text{ZKP}) &= n^2 n_g^w, \\ O(\text{ZKF}) - O(H_\infty\text{ZF}) &= n_g (n + \log(n_g) - n^2) + p(n - n^2) + n^2(\varphi_g - n_g^v) + m^2(n + m + n_g^v), \\ O(\text{ZKP}) - O(H_\infty\text{ZF}) &= n_g (n + \log(n_g) - n^2) + p(n - n^2) + n^2(\varphi_g - n_g^w - n_g^v) + m^2(n + m + n_g^v). \end{aligned}$$

Then, ZKP may be computationally cheaper than ZKF, while $H_\infty\text{ZF}$ may be the fastest algorithm.

Algorithm 3.4: $\hat{\mathcal{X}}_k = \text{ZKF}(\hat{\mathcal{X}}_{k-1}, A_{k-1}, B_{k-1}, \mathcal{W}_{k-1}, C_k, \mathcal{V}_k, u_{k-1}, y_k, \varphi_g)$

- 1 Compute $\hat{\mathcal{X}}_{k-1}^\downarrow = \text{ZR}(\hat{\mathcal{X}}_{k-1}, \varphi_g)$, where ZR is discussed in Subsection B.1.1.1
 - 2 Compute the centers $\hat{x}_{k|k-1}$ in (3.88) and $\hat{y}_{k|k-1}$ in (3.89), and the covariation matrices $\hat{C}_{k|k-1}^{xx}$ in (3.93), C_k^{vv} in (3.94), $\hat{C}_{k|k-1}^{xy}$ in (3.95), and $\hat{C}_{k|k-1}^{yy}$ in (3.96)
 - 3 Calculate the optimal gain (3.92), and thereby, the estimates of generator matrix $\hat{G}_k^x$ in (3.91) and center $\hat{x}_k$ in (3.87) to obtain $\hat{\mathcal{X}}_k = \{\hat{G}_k^x, \hat{x}_k\}$
-

Remark 3.1.4. Unlike stochastic algorithms as DGF and DH₂GF, set-based estimators as ZKF and ZKF work on representations of uncertainty that can vary their complexity over time. For instance, the zonotopes $\mathcal{W}_{k-1} \ni w_{k-1}$ and $\mathcal{W}_k \ni w_k$ have $n_{g,k-1}^w$ and $n_{g,k}^w$ generators, respectively, and maybe $n_{g,k-1}^w \neq n_{g,k}^w$. However, whenever a direct comparison of worst-case complexity order is here performed among algorithms that involve time-varying orders, the same order of uncertainty will be assumed. In this case, $n_g^w = n_{g,k-1}^w = n_{g,k}^w$.

3.1.6.2 Time-Invariant Case

Now, we propose an LTI algorithm with the same structure of the ZKF, whose difference is that H_∞ norm is chosen as cost criterion. The resulting algorithm is called *new H_∞ zonotopic filter* (NH_∞ZF).

Consider the dynamical system described by (3.85)-(3.86), whose matrices are now constant. No rank assumption for generators matrices is required. By defining the center estimate

$$\hat{x}_k \triangleq (I_n - KC)(A\hat{x}_{k-1} + Bu_{k-1}) + Ky_k, \quad (3.97)$$

we obtain the state-estimation error

$$\begin{aligned} e_k &\triangleq x_k - \hat{x}_k \\ &= (\bar{x}_k - \hat{x}_k) + x_k^z \\ &= \hat{x}_k^z \\ &= (I_n - KC)(A\hat{x}_{k-1}^z + w_{k-1}) - Kv_k \\ &= (I_n - KC)(A\hat{x}_{k-1}^z + G^w \xi_{k-1}^w) - KG^v \xi_k^v, \end{aligned} \quad (3.98)$$

with $x_k^z \in \mathcal{X}_k^z = \{G_k^x, 0_{n \times 1}\}$, $e_k \in \mathcal{X}_k^{z+} = \{G_k^x, (\bar{x}_k - \hat{x}_k)\}$, $\hat{x}_k^z \in \hat{\mathcal{X}}_k^z = \{\hat{G}_k^x, 0_{n \times 1}\} \supseteq \mathcal{X}_k^{z+}$, $\xi_{k-1}^w \in \mathcal{B}_s^{n_g^w}$, and $\xi_k^v \in \mathcal{B}_s^{n_g^v}$. In order to calculate the constant gain $K \in \mathbb{R}^{n \times m}$ via H_∞ norm, we rewrite $\hat{x}_k^z$ as follows:

$$\hat{x}_k^z = \tilde{A}\hat{x}_{k-1}^z + \tilde{B}\tilde{\xi}_{k-1}, \quad (3.99)$$

$$z_k = \hat{x}_k^z, \quad (3.100)$$

where $\tilde{A} = (I_n - KC)A$, $\tilde{B} = [(I_n - KC)G^w \quad -KG^v]$, and $\tilde{\xi}_{k-1} = [(\xi_{k-1}^w)^\top \quad (\xi_k^v)^\top]^\top$. Next, we apply

Lemma 2.5.3 to obtain the optimal gain $\check{K}$. This step is presented in Theorem 3.1.3 using linear matrix inequalities.

Theorem 3.1.3. Consider variables $P = P^\top > 0_{n \times n}$, $Y \in \mathbb{R}^{n \times m}$, and $\gamma > 0$, with γ being the square H_∞ norm of the zonotopic system (3.99)-(3.100). By solving the optimization problem

$$\min \quad \gamma \quad (3.101)$$

$$\text{s.t.} \quad \begin{bmatrix} I_n - P & 0_{n \times (n_g^w + n_g^v)} & \Omega_1^\top \\ 0_{(n_g^w + n_g^v) \times n} & -\gamma I_{(n_g^w + n_g^v)} & \Omega_2^\top \\ \Omega_1 & \Omega_2 & -P \end{bmatrix} < 0_{(2n + n_g^w + n_g^v) \times (2n + n_g^w + n_g^v)}, \quad (3.102)$$

where

$$\Omega_1 = PA - YCA, \quad (3.103)$$

$$\Omega_2 = [(P - YC)G^w \quad -YG^v], \quad (3.104)$$

we obtain the optimal gain $\check{K} = \check{P}^{-1}\check{Y}$.

Proof. By relating the error system (3.99)-(3.100) to (2.67), the following matrix inequality is yielded:

$$\begin{bmatrix} \tilde{A}^\top P \tilde{A} + I_n - P & \tilde{A}^\top P \tilde{B} \\ \tilde{B}^\top P \tilde{A} & \tilde{B}^\top P \tilde{B} - \gamma I_{(n_g^w + n_g^v)} \end{bmatrix} < 0_{(n + n_g^w + n_g^v) \times (n + n_g^w + n_g^v)}.$$

This inequality can be rewritten as

$$\begin{bmatrix} I_n - P & 0_{n \times (n_g^w + n_g^v)} \\ 0_{(n_g^w + n_g^v) \times n} & -\gamma I_{(n_g^w + n_g^v)} \end{bmatrix} + \begin{bmatrix} \tilde{A}^\top P \\ \tilde{B}^\top P \end{bmatrix} P^{-1} \begin{bmatrix} P \tilde{A} & P \tilde{B} \end{bmatrix} < 0_{(n + n_g^w + n_g^v) \times (n + n_g^w + n_g^v)}.$$

Defining the linearizing transformation $Y = PK$ and applying Schur's complement, we obtain the linear matrix inequality in (3.102). The minimization of γ subject to the prior constraint yields the optimal matrices $\check{P}$ and $\check{Y}$, and thereby, $\check{K}$. ■

By employing Theorem 3.1.3, we guarantee that $NH_\infty ZF$ is asymptotically stable. Then, assuming that x_0 , w_{k-1} , and v_k are random terms defined in zonotopes, Property 3.1.2 is also applicable to (3.97).

Since the design matrix K was specified, we next derive a reachable set analysis to specific the $NH_\infty ZF$ exploiting both precision and low-cost computation with zonotopes. Firstly, note that the error equation $\hat{x}_k^z$, given by (3.98), accords with the following time evolution:

$$\begin{aligned} \hat{x}_1^z &= \tilde{A} \hat{x}_0^z + (I_n - KC) w_0 - K v_1, \\ \hat{x}_2^z &= \tilde{A}^2 \hat{x}_0^z + \tilde{A} (I_n - KC) w_0 + (I_n - KC) w_1 - \tilde{A} K v_1 - K v_2, \end{aligned}$$

$$\begin{aligned}
& \vdots \\
\hat{x}_k^z &= \tilde{A}^k \hat{x}_0^z + \sum_{i=0}^{k-1} \tilde{A}^{(k-1-i)} [(\mathbf{I}_n - KC) w_i - K v_{i+1}] \\
&= \tilde{A}^k \hat{x}_0^z + \sum_{i=0}^{k-1} \tilde{A}^i [(\mathbf{I}_n - KC) w_{k-1-i} - K v_{k-i}].
\end{aligned}$$

Replacing each element by its corresponding zonotope and applying Property 2.2.1 yields

$$\square \hat{X}_k^z = \square \left\{ \tilde{A}^k \hat{G}_0^x \mathcal{B}^{n_g} \right\} \oplus \bigoplus_{i=0}^{k-1} \left[\square \left\{ \tilde{A}^i (\mathbf{I}_n - KC) G^w \mathcal{B}^{n_g^w} \right\} \oplus \square \left\{ -\tilde{A}^i K G^v \mathcal{B}^{n_g^v} \right\} \right].$$

Finally, by applying Property 2.2.2 together with the row sum in (2.26), we make explicit the radius of $\square \hat{X}_k^z$ as

$$\zeta_k = \text{rs} \left(\tilde{A}^k \hat{G}_0^x \right) + \sum_{i=0}^{k-1} \left[\text{rs} \left(\tilde{A}^i (\mathbf{I}_n - KC) G^w \right) + \text{rs} \left(-\tilde{A}^i K G^v \right) \right].$$

Then, we should store products and sums in auxiliary variables to support the next iterations.

The $\text{NH}_\infty\text{ZF}$ is summarized in Algorithm 3.5, which solves Problem 1.2.3 without enforcing state constraints. We assume that Theorem 3.1.3 is offline solved. The complexity order of the proposed algorithm is given by

$$O(\text{NH}_\infty\text{ZF}) = O\left(n^2 \left(n_g + n_g^w + n_g^v\right) + n(p + m)\right).$$

Therefore, $\text{NH}_\infty\text{ZF}$ is faster than $O(\text{H}_\infty\text{ZF})$ and may imply lower cost computations than ZKP and ZKF.

Since

Algorithm 3.5: $\left[\square \hat{X}_k, \hat{x}_k, \Pi_k^x, \Pi_k^w, \Pi_k^v, \zeta_k^w, \zeta_k^v \right] = \text{NH}_\infty\text{ZF}(\hat{x}_{k-1}, \Pi_{k-1}^x, \Pi_{k-1}^w, \Pi_{k-1}^v, \zeta_{k-1}^w, \zeta_{k-1}^v, \tilde{A}, \tilde{D}, \tilde{K}, u_{k-1}, y_k)$

Initialization: $\tilde{A} = (\mathbf{I}_n - \tilde{K}C)A$, $\tilde{D} = (\mathbf{I}_n - \tilde{K}C)B$, $\Pi_0^x = \hat{G}_0^x$, $\Pi_0^w = \tilde{A}^{-1}(\mathbf{I}_n - \tilde{K}C)G^w$,
 $\Pi_0^v = -\tilde{A}^{-1}\tilde{K}G^v$, $\zeta_0^w = \zeta_0^v = \mathbf{0}_{n \times 1}$

- 1 Do $\Pi_k^x = \tilde{A}\Pi_{k-1}^x$, $\Pi_k^w = \tilde{A}\Pi_{k-1}^w$, $\Pi_k^v = \tilde{A}\Pi_{k-1}^v$, $\zeta_k^w = \text{rs}(\Pi_k^w) + \zeta_{k-1}^w$, and $\zeta_k^v = \text{rs}(\Pi_k^v) + \zeta_{k-1}^v$
 - 2 Calculate the radius $\zeta_k = \text{rs}(\Pi_k^x) + \zeta_k^w + \zeta_k^v$
 - 3 Compute the center estimate $\hat{x}_k = \tilde{A}\hat{x}_{k-1} + \tilde{D}u_{k-1} + \tilde{K}y_k$
 - 4 Determine the box $\square \hat{X}_k = [\hat{x}_k - \zeta_k, \hat{x}_k + \zeta_k]$
-

Remark 3.1.5. Algorithms as the $\text{NH}_\infty\text{ZF}$ are commonly called *interval* because their output is an interval (Meslem et al., 2018; Tang et al., 2019a,b; Martinez et al., 2020). However, sets more general than intervals are operated each time step, even though indirectly, with the interval output being adopted by particular reasons such as low-cost

computation and good precision (due to the absence of wrapping effect caused by order-reduction algorithms). Then, interval observers are here named according to the most general class being manipulated, justifying why the $\text{NH}_\infty\text{ZF}$ is called zonotopic.

3.1.7 Numerical Example

Now, we exemplify the use of the H_∞ZF , $\text{NH}_\infty\text{ZF}$, ZKP, and ZKF algorithms. Consider the five-state dynamical system discussed in Subsection 3.1.3. Let $w_{k-1} \in \mathcal{W} = \{[F^w \ 0_{5 \times 4}], 0_{5 \times 1}\}$, $v_k \in \mathcal{V} = \{F^v, 0_{2 \times 1}\}$, and $x_0 \in \hat{\mathcal{X}}_0 = \{5 \times \text{I}_5, 0_{5 \times 1}\}$, where F^w and F^v are given by (3.60) and (3.61), respectively. To estimate states, we set $x_0 = [4 \ -4 \ 4 \ -4 \ 4]^\top$, $m_s = 1000$, and $k_f = 60$; the noise terms are taken from uniform distributions.

By solving the constrained optimization problem given by (B.45), the following constant matrices are obtained for H_∞ZF :

$$\check{T} = \begin{bmatrix} 0.0099 & 0 & 0 & 0.0034 & 0 \\ 0.0006 & 1 & 0 & -0.3180 & 0 \\ -0.0037 & 0 & 1 & 0.1806 & 0 \\ 0.0023 & 0 & 0 & 0.5714 & 0 \\ 0.9817 & 0 & 0 & 0.5318 & 1 \end{bmatrix}, \check{N} = \begin{bmatrix} 0.9901 & -0.0034 \\ -0.0006 & 0.3180 \\ 0.0037 & -0.1806 \\ -0.0023 & 0.4286 \\ -0.9817 & -0.5318 \end{bmatrix}, \check{K} = \begin{bmatrix} -0.0039 & 0.0009 \\ 0.4213 & 0.2376 \\ -0.1554 & -0.0262 \\ 0.3268 & 0.3034 \\ -0.7587 & -0.0940 \end{bmatrix}, \quad (3.105)$$

whose minimal H_∞ norm is equal to 0.1787. By employing Theorem 3.1.3, the gain matrix

$$\check{K} = \begin{bmatrix} 0.9903 & -0.0064 \\ -0.0002 & 0.5007 \\ 0.0106 & -0.6218 \\ -0.0064 & 0.9958 \\ -0.9695 & -1.2971 \end{bmatrix} \text{ is returned for } \text{NH}_\infty\text{ZF}, \text{ whose minimal } \text{H}_\infty \text{ norm is equal to } 0.2427.$$

To execute both ZKP and ZKF, we set $\varphi_g = 32$. For comparison purposes, we employ the indexes of T^{CPU} in (3.57), RMSE in (3.58), and $r^\square$ in (3.84). We also set $k_0 = 1$ to compare the RMSE of both stationary-gain and varying-gain algorithms. The following computer configuration was used: 8 GB RAM 1333 MHz, Windows 10 Pro, and AMD FX-6300 CPU 3.50 GHz.

In Figure 3.4, the non-measured state x_5 is involved by interval hulls $\square\hat{\mathcal{X}}_k$ and approximated by punctual estimates $\hat{x}_k$ in (a), while the largest radius of interval hulls is illustrated in (b). As expected from varying-gain algorithms as ZKP and ZKF, their reduction of radius is faster than that realized in fixed-gain algorithms as H_∞ZF and $\text{NH}_\infty\text{ZF}$ during the transient. Moreover, since filters as ZKF, H_∞ZF , and $\text{NH}_\infty\text{ZF}$ assimilate more current measurements than predictors, both their general accuracy and precision are better than the ZKP ones. These results are corroborated by Table 3.6.

By convenience we set $\varphi_g = 32$, since such value makes ZKF return steady-state precision competitive in comparison with H_∞ZF . For $\varphi_g > 32$, the precision of ZKF is going

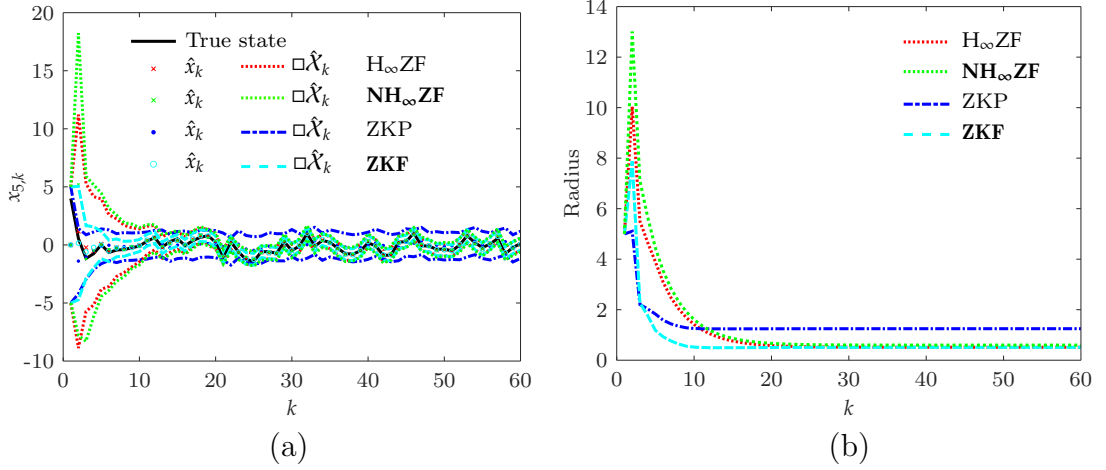


Figure 3.4: Graph (a): Time evolution of state $x_{5,k}$ (black —) and of the estimates of center $\hat{x}_k$ and box $\hat{X}_k$ computed by H_∞ ZF (red color), NH_∞ ZF (green color), ZKP (blue color), and ZKF (cyan color). Graph (b): Time evolution of the largest radius of interval hulls.

Table 3.6: Numerical results from the state estimation of the system (3.60)-(3.61) using guaranteed approaches.

Estimators	$T^{\text{CPU}} \times 10^{-4}(\text{s})$	$r^\square$	RMSE ₁	RMSE ₂	RMSE ₃	RMSE ₄	RMSE ₅	$\frac{1}{5} \sum_{i=1}^5 \text{RMSE}_i$
H_∞ ZF (Tang et al., 2019a)	0.309	1.14	0.520	0.520	0.521	0.540	0.546	0.529
NH_∞ ZF	0.247	1.35	0.520	0.537	0.540	0.520	0.817	0.587
ZKP (Combastel, 2015)	0.786	1.42	0.778	0.541	0.524	0.523	0.796	0.632
ZKF	0.820	0.781	0.520	0.528	0.539	0.517	0.537	0.528

to improve at the cost of a larger T^{CPU} . Anyway, H_∞ ZF stands out for its smaller T^{CPU} . In this example, H_∞ ZF is more precise and accurate than NH_∞ ZF because its degrees of freedom confer a smaller H_∞ norm. Conversely, since ZKF has the same estimator structure than NH_∞ ZF and returns better solutions than H_∞ ZF, there may exist examples in which NH_∞ ZF achieves a performance better than or competitive with H_∞ ZF. Anyway, H_∞ ZF motivates the introduction of degrees of freedom in NH_∞ ZF to reduce the H_∞ norm.

3.2 Nonlinear Systems

3.2.1 Improvements for the Zonotopic Filter

In this subsection, we present two contributions to the zonotopic filter based on mean value extension (ZMV) presented in Subsection B.2.2. The first one is to extend the ZMV to the interval zonotopic filter based on mean value extension (IZMV) given by Algorithm 3.6. This algorithm allows to enforce the state interval constraints

$$a_{i,k} \leq x_{i,k} \leq b_{i,k}, \quad i = 1, \dots, n, \quad (3.106)$$

where $a_k \in \mathbb{R}^n$ and $b_k \in \mathbb{R}^n$ are the lower and upper bounds of the states, using the intersection between strip and zonotope proposed in the ZMV. Although we focus on interval constraints, this procedure allows for enforcing up to zonotopic constraints. Therefore, we expect to improve both Frobenius norm and volume measure with respect to the ZMV in the presence of (3.106). The second contribution is to improve the efficiency of the original volume minimization (VM1) discussed in Subsection B.2.2.1. To achieve that, we rewrite the convex nonlinear optimization problem as an LP. Therefore, the goal is to solve Problem 1.2.2 enforcing the interval constraints (3.106). These improvements were published in de Paula et al. (2022c).

3.2.1.1 A New Interval Zonotopic Filter

Consider the problem formulation (1.1)-(1.2), where the terms w_{k-1} , v_k , and x_0 are characterized by the zonotopes $\mathcal{W}_{k-1} = \{G_{k-1}^w, c_{k-1}^w\}$, $\mathcal{V}_k = \{G_k^v, c_k^v\}$, and $\hat{\mathcal{X}}_0 = \{\hat{G}_0^x, \hat{x}_0\}$, respectively. In order to deal with the interval constraints (3.106), we propose to append a new step to the ZMV. Basically, after the reduced-order set $\hat{\mathcal{X}}_k$ being computed, a constrained set can be obtained by the intersection between the state interval constraints (3.106) and the reduced-order set $\hat{\mathcal{X}}_k$. Such an intersection is the same one proposed in Alamo et al. (2005), except for the way how each strip is determined, such that strips are generated by the interval constraints. The proposed step reminds the PDF truncation of TUKF (see Subsection A.2.2.1), since the interval constraints are individually enforced.

Next, we present the new ZMV algorithm with interval constraints, here called IZMV, in Algorithm 3.6. For completeness, the IZMV is illustrated in Figure 3.5 with the following performance criteria: segment minimization (SM) and volume minimization (VM). Its complexity order when employing SM is given by

$$O(\text{IZMV}) = O\left((n + n_s + m)(mn^2 + \log(n + n_s + m)) + n^3\varphi_s\right).$$

For the sake of simplicity, state interval constraints were considered in Algorithm 3.6. However, zonotopic constraints can also be incorporated by handling zonotopes in the halfspace format $Ax \leq b$. In this case, Steps 5–7 of the IZMV should be slightly modified to define the strips. That is, once the parallel hyperplanes $A_{i,:}x \leq b_i$ and $A_{j,:}x \leq b_j$ are identified, where $A_{i,:} = -A_{j,:}$ with i and j being indexes of rows, define $c^\tau = A_{i,:}$, $d = \frac{b_i - b_j}{2}$, and $\rho = b_i - d > 0$. The resulting algorithm reminds the ZMV+F employed in Rego et al. (2021) and yields similar results.

Example 3.2.1. Consider the constraint box $\mathcal{R} = \begin{bmatrix} 2.5 & 0 \\ 0 & 2.5 \end{bmatrix} \mathcal{B}^2 \oplus \begin{bmatrix} 2.5 \\ 2.5 \end{bmatrix}$ and the zonotope $\mathcal{X} = \begin{bmatrix} 1 & 2 \\ 3 & 2 \end{bmatrix} \mathcal{B}^2 \oplus \begin{bmatrix} 2 \\ 1 \end{bmatrix}$. Let $\mathcal{X}$ be the zonotope returned by the IZMV at step 1, and let $\mathcal{R}$ be the interval constraints (3.106). Figure 3.6 illustrates that the set $\mathcal{X}$ (—) exceeds the

Algorithm 3.6: $\check{\mathcal{X}}_k = \text{IZMV}(\check{\mathcal{X}}_{k-1}, \mathcal{W}_{k-1}, \mathcal{V}_k, u_{k-1}, y_k, \varphi_g, a_k, b_k, f, h)$

```

1  $\hat{\mathcal{X}}_k = \text{ZMV}(\check{\mathcal{X}}_{k-1}, \mathcal{W}_{k-1}, \mathcal{V}_k, u_{k-1}, y_k, \varphi_g, f, h)$ , where ZMV is discussed in Subsection
   B.2.2
2  $\mathcal{R} = [a_k, b_k]$  is the state interval constraints
3  $\check{\mathcal{X}}_{1,k} = \hat{\mathcal{X}}_k$  is the initialization zonotope
4 for  $i = 1, \dots, n$  do
5    $c_i = (\mathbf{I}_n)_{:,i}$  is the direction of a strip
6    $d_i = \text{mid}_i(\mathcal{R})$  is the shift of a strip
7    $\rho_i = \text{rad}_i(\mathcal{R})$  is the width of a strip
8    $\mathcal{S} = \{x \in \mathbb{R}^n : |c_i^\top x - d_i| \leq \rho_i\}$  is the  $i$ th strip
9    $\check{\mathcal{X}}_{i+1,k} = \check{\mathcal{X}}_{i,k} \cap \mathcal{S}_i$ , whose performance criterion can be SM or VM discussed in
      Subsection B.2.2.1
10 end
11  $\check{\mathcal{X}}_k = \check{\mathcal{X}}_{n+1,k}$  is the output zonotope

```

limits of the set $\mathcal{R}$ ($-\cdot-$). Therefore, the IZMV runs steps 2–11 with $n = 2$ to refine $\mathcal{X}$ by means of the intersection with $\mathcal{R}$. Given the SM and VM1 algorithms (see Subsection **B.2.2.1**), the corresponding zonotopes $\mathcal{X}_{\text{seg.min}}(\cdots)$ and $\mathcal{X}_{\text{vol.min}}(--)$ are obtained. As the exact intersection between $\mathcal{X}$ and $\mathcal{R}$ is not a zonotope, the generated sets over the proposed intersection procedure are more conservative than the exact set to obtain zonotopes. Note that the VM1 returns a more precise zonotope than does the SM.

3.2.1.2 Novel Computational Improvement for the Minimum-Volume Approach

The ZMV algorithm proposed by Alamo et al. (2005) performs an intersection between a zonotope and a strip in the corrected set step. The goal of this intersection is to yield a zonotope in (2.24)-(2.25) parametrized by means of the vector κ . This vector is determined through some criterion. In Alamo et al. (2005), two criteria are proposed, namely segment minimization and volume minimization, which are presented in Subsection **B.2.2.1**. The former minimizes the Frobenius norm (1.6), whose resulting function is nonlinear. The latter criterion minimizes the volume of zonotope (1.7), whose resulting function is piecewise linear. Both optimization problems are convex, but only the first one has explicit solution.

Due to the computational burden to run the volume minimization, an improvement for the processing time is proposed in Bravo et al. (2006), where the optimization problem is replaced by direct computations on candidate zonotopes. However, this improvement has a drawback when applied exactly, that is, using (1.7) to compute volume of zonotopes. Consider that there are many candidate zonotopes. Even if the first analyzed zonotope has the smallest volume, the algorithm must compute all remaining zonotopes to discard them at the end.

Due to the absolute value operator, the original volume approach is composed by

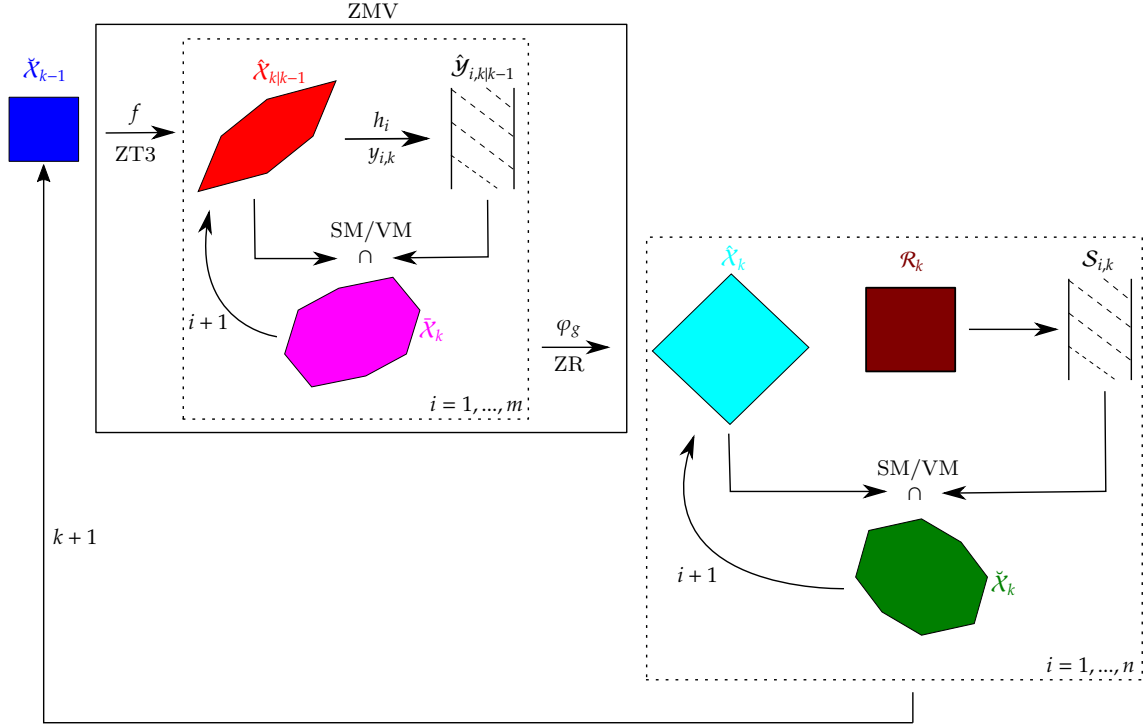


Figure 3.5: Illustrative diagram of the interval zonotopic filter based on mean value extension (IZMV) given by Algorithm 3.6. First, the ZMV is run with the zonotope $\hat{X}_{k-1}$ to yield the zonotope $\hat{X}_k$. This set is introduced in a loop indexed by i , whose goal is to assimilate the interval constraints $\mathcal{R}_k$ through the intersection between the zonotope $\hat{X}_k$ and the i th strip $\mathcal{S}_{i,k}$, for $i = 1, \dots, n$. Unlike the ZMV, the strips are uniquely defined by means of the intervals of $\mathcal{R}_k$. Choosing the SM or the VM to compose the intersection, the zonotope $\check{X}_k$ is obtained. At the end of each iteration, the zonotope $\hat{X}_k$ is replaced by the zonotope $\check{X}_k$ to hold the loop. After n iterations, the IZMV whose output $\check{X}_k$ becomes the input $\hat{X}_{k-1}$ at the next time step ($k+1$) is completed.

piecewise linear functions. Then, it cannot be directly solved via linear programming. An alternative way is to apply nonlinear solvers at the cost of low efficiency, or to apply the improvement from Bravo et al. (2006) instead. Therefore, LPs have not been motivated yet to exactly solve the volume minimization stated in Algorithm B.8.

This subsection proposes to formulate the original volume problem as an LP, whose result is called VM3. To treat the absolute values, some procedures presented in Bertsimas & Tsitsiklis (1997) are used. In doing so, efficient solvers such as CPLEX and Gurobi are used. To solve LPs rather than the algorithm proposed by Bravo et al. (2006) may be a more efficient way, since all candidate zonotopes do not need to be tested. Recall that the exact version of VM2 is being compared; the approximate version of VM2 is faster but deteriorates the precision, therefore it is not here discussed. The main differences among VM1, VM2, and VM3 are summarized in Figure 3.7.

Then, consider the original volume minimization problem (Alamo et al., 2005). Given the zonotope $\mathcal{X} = G^x \mathcal{B}^{n_g} \oplus \hat{x}$ and the strip $\mathcal{S} = \{x \in \mathbb{R}^n : |c^T x - d| \leq \rho\}$, we seek the smallest

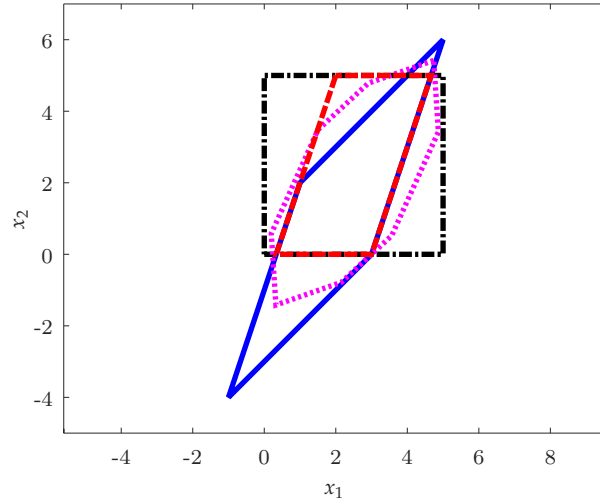


Figure 3.6: Example of the intersection between the constraint box $\mathcal{R} = 2.5 \times \mathbf{I}_2 \mathcal{B}^2 \oplus [2.5 \ 2.5]^\top$ (black dotted-dash line) and the zonotope $\mathcal{X} = [g_1^x \ g_2^x] \mathcal{B}^2 \oplus [2 \ 1]^\top$ (blue solid line), where $g_1^x = [1 \ 3]^\top$ and $g_2^x = [2 \ 2]^\top$, using strips. Assume that $\mathcal{X}$ is the zonotope returned by the IZMV at step 1. Since $\mathcal{X}$ exceeds the constraint box $\mathcal{R}$, intersections are performed between $\mathcal{X}$ and $\mathcal{R}$ at steps 2–11 to return a zonotope more precise than $\mathcal{X}$. To achieve that, each one-dimensional interval of $\mathcal{R}$ defines a strip, and each strip is used to intersect a zonotope. Another intersection is performed between the prior resulting zonotope and a new strip. This iterative process lasts until all strips are used. Each intersection depends on an optimal criterion, namely segment (magenta dotted line) and volume (red dashed line) minimizations. The former minimizes the Frobenius norm, while the latter minimizes the volume of the intersection.

zonotope $\tilde{\mathcal{X}}(\kappa)$, where the optimal vector $\kappa \in \mathbb{R}^n$ minimizes

$$\text{Vol}(\tilde{\mathcal{X}}(\kappa)) = 2^n \sum_{i=1}^{N(n_g, n)} |\det[(\mathbf{I}_n - \kappa c^\top) T_i^a]| + 2^n \sum_{i=1}^{N(n_g, n-1)} |\det[(\mathbf{I}_n - \kappa c^\top) T_i^b \ \rho \kappa]|, \quad (3.107)$$

where $T_i^a \in \mathbb{R}^{n \times n}$ denotes all matrices that can be obtained taking n columns from the matrix G^x , and $T_i^b \in \mathbb{R}^{n \times (n-1)}$ denotes all matrices that can be obtained taking $(n-1)$ columns from G^x .

Using properties of determinant, (3.107) is rewritten in Alamo et al. (2005) as

$$\text{Vol}(\tilde{\mathcal{X}}(\kappa)) = 2^n \sum_{i=1}^{N(n_g, n)} |\det(T_i^a)| |1 - c^\top \kappa| + 2^n \sum_{i=1}^{N(n_g, n-1)} \rho |\det([T_i^b \ v_i])| |v_i^\top \kappa|, \quad (3.108)$$

where $v_i^\top T_i^b = \mathbf{0}_{1 \times (n-1)}$ and $v_i^\top v_i = 1$. In other words, $v_i \in \mathbb{R}^n$ is any normalized non-null vector such that $(T_i^b)^\top v_i = \mathbf{0}_{(n-1) \times 1}$.

Note that $|1 - c^\top \kappa|$ and $|v_i^\top \kappa|$ are piecewise linear functions. Also, due to the absolute value involving the matrix determinants, these terms are nonnegative. In this case, (3.108) is composed by summation of piecewise linear convex functions. According to Bertsimas & Tsitsiklis (1997), given that the cost coefficients are nonnegative, the summation of

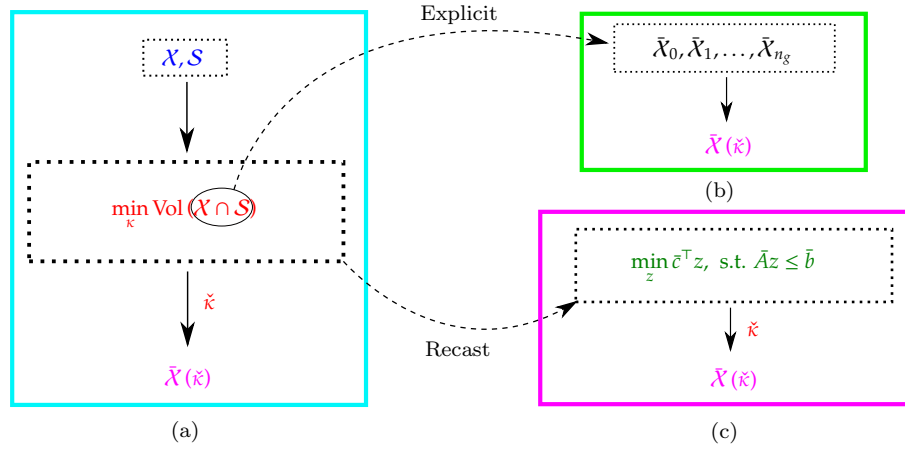


Figure 3.7: The different methods that compute the intersection between a strip ($\mathcal{S}$) and zonotope ($\mathcal{X}$) based on the volume minimization (VM) approach. Such methods are equivalent in the sense that the same smallest volume zonotope $\bar{\mathcal{X}}(\kappa)$ is yielded, with κ being the optimal parameter. (a) VM1 (proposed in Alamo et al. (2005)) determines the best zonotope $\bar{\mathcal{X}}(\kappa)$ through the solution κ of a nonlinear optimization problem. Although this problem is convex, it requires a nonlinear solver to yield solutions κ online. Therefore, the VM1 may demand a large processing time. (b) To improve this time issue, VM2 (proposed in Bravo et al. (2006)) replaces the search of parameter $\hat{\kappa}$ by the search of (n_g+1) known zonotopes $\bar{\mathcal{X}}_j$, for $j = 0, \dots, n_g$, which are possible solutions for the intersection between the strip $\mathcal{S}$ and the zonotope $\mathcal{X}$ of order n_g . The best zonotope $\bar{\mathcal{X}}(\kappa)$ is extracted by evaluating the volume of each candidate zonotope. (c) Motivated by the convexity of the optimization problem solved in the VM1, we recast such nonlinear problem as an LP. This reformulation is described in VM3, whose optimal solution κ can be efficiently computed using specific solvers.

piecewise linear convex functions is a piecewise linear convex function. This is called *cost criterion*. Note that the costs must be nonnegative to maintain convexity. Therefore, (3.108) can be rewritten as an LP, which guarantees the same optimal solution. The necessary transformations are well exemplified in Bertsimas & Tsitsiklis (1997, Example 1.5).

Now, the steps to transform (3.108) in an LP are presented.

Proposition 3.2.1. The unconstrained optimization problem given by the minimization of (3.108) is equivalent to the LP

$$\min_z \bar{c}^T z \quad \text{s.t.} \quad \bar{A}z \leq \bar{b}, \quad (3.109)$$

where $z \in \mathbb{R}^{\bar{n}}$ is the variable vector, $\bar{n} = n + 1 + N(n_g, n - 1)$, $\bar{c} \in \mathbb{R}^{\bar{n}}$ is the cost vector, $\bar{A} \in \mathbb{R}^{\bar{m} \times \bar{n}}$ and $\bar{b} \in \mathbb{R}^{\bar{m}}$ compose the inequality constraints, and $\bar{m} = 2(1 + N(n_g, n - 1))$. These

parameters are defined as

$$\bar{c} = \begin{bmatrix} 0_{1 \times n} & 2^n \sum_{i=1}^{N(n_g, n)} |\det(T_i^a)| & 2^n \rho \left| \det \begin{bmatrix} T_1^b & v_1 \end{bmatrix} \right| & \cdots & 2^n \rho \left| \det \begin{bmatrix} T_{N(n_g, n-1)}^b & v_{N(n_g, n-1)} \end{bmatrix} \right| \end{bmatrix}^\top, \quad (3.110)$$

$$\bar{A}_{1:2,1:\bar{n}} = \begin{bmatrix} -c^\top & -1 & 0_{1 \times N(n_g, n-1)} \\ c^\top & -1 & 0_{1 \times N(n_g, n-1)} \end{bmatrix}, \quad (3.111)$$

$$\bar{A}_{3:4,1:\bar{n}} = \begin{bmatrix} v_1^\top & 0 & -1 & 0_{1 \times (N(n_g, n-1)-1)} \\ -v_1^\top & 0 & -1 & 0_{1 \times (N(n_g, n-1)-1)} \end{bmatrix}, \quad (3.112)$$

$$\bar{A}_{5:6,1:\bar{n}} = \begin{bmatrix} v_2^\top & 0_{1 \times 2} & -1 & 0_{1 \times (N(n_g, n-1)-2)} \\ -v_2^\top & 0_{1 \times 2} & -1 & 0_{1 \times (N(n_g, n-1)-2)} \end{bmatrix}, \quad (3.113)$$

$$\vdots \quad (3.114)$$

$$\bar{A}_{\bar{m}-1:\bar{m},1:\bar{n}} = \begin{bmatrix} v_{N(n_g, n-1)}^\top & 0_{1 \times N(n_g, n-1)} & -1 \\ -v_{N(n_g, n-1)}^\top & 0_{1 \times N(n_g, n-1)} & -1 \end{bmatrix}, \quad (3.115)$$

and

$$\bar{b} = \begin{bmatrix} 1_{1 \times 2} & 0_{1 \times 2N(n_g, n-1)} \end{bmatrix}^\top. \quad (3.116)$$

Proof. Recall that there exists a summation of two parcels in (3.108). Let $z_{1:n} = \kappa$. Since κ does not appear without the absolute value, its cost is null, that is, $\bar{c}_{1:n} = 0_{n \times 1}$. The first parcel (related to T_i^a) is replaced by $\bar{c}_{n+1} z_{n+1}$, where

$$\bar{c}_{n+1} = 2^n \sum_{i=1}^{N(n_g, n)} |\det(T_i^a)|, \quad (3.117)$$

and two constraints are added to treat the absolute value, that is,

$$1 - \bar{c}_{1:n}^\top z_{1:n} \leq z_{n+1}, \quad (3.118)$$

$$-1 + \bar{c}_{1:n}^\top z_{1:n} \leq z_{n+1}. \quad (3.119)$$

The second parcel (related to T_i^b) is replaced by $\bar{c}_{(n+2):(n+1+N(n_g, n-1))}^\top z_{(n+2):(n+1+N(n_g, n-1))}$, where

$$\bar{c}_{n+1+i} = 2^n \rho \left| \det \begin{bmatrix} T_i^b & v_i \end{bmatrix} \right|, \quad i = 1, \dots, N(n_g, n-1), \quad (3.120)$$

and $2N(n_g, n-1)$ inequality constraints are enforced, that is,

$$v_i^\top z_{1:n} \leq z_{n+1+i}, \quad (3.121)$$

$$-v_i^\top z_{1:n} \leq z_{n+1+i}, \quad (3.122)$$

for $i = 1, \dots, N(n_g, n-1)$.

These parcels compose the LP (3.109), whose parameters are given by $\bar{c}$ in (3.110), $\bar{A}$ in (3.111)-(3.115), and $\bar{b}$ in (3.116). Ultimately, there are $\bar{n} = n + 1 + N(n_g, n-1)$ variables and $\bar{m} = 2(1 + N(n_g, n-1))$ linear inequality constraints. ■

After solving the LP (3.109), the desired vector $\check{x}$ is extracted from the optimal solution $\check{z}$ to define the smallest volume zonotope $\bar{\mathcal{X}}$. This procedure is described in Algorithm 3.7. Considering that LPs are solved with the same computational efficiency than Kelner & Spielman (2006), we have:

$$O(\text{VM3}) = O\left(n^3 [N(n_g, n) + N(n_g, n-1)] + [n + N(n_g, n-1)] [N(n_g, n-1)]^3 + n^2 n_g\right).$$

Although the order comparison between VM2 (Subsection B.2.2.1) and VM3 is not direct, the main difference is the way in which the optimal zonotope is obtained. VM2 builds a candidate zonotope and evaluates its volume using (1.7); this procedure occurs $(n_g + 1)$ times. VM3 uses (1.7) only once and solves an LP related to. However, the construction of the LP demands $N(n_g, n-1)$ additional combinations (not only $N(n_g, n)$) and the computation of null space (to define vectors v_i). Then, for $n_g \approx n$, VM2 may be faster, while VM3 may stand out with the increase of $n_g - n$. This fact has been verified during simulations.

Algorithm 3.7: $\bar{\mathcal{X}} = \text{VM3}(\mathcal{X}, \mathcal{S})$

- 1 Given the zonotope $\mathcal{X} = G^x \mathcal{B}^{n_g} \oplus \hat{x} \subset \mathbb{R}^n$ and strip $\mathcal{S} = \{x \in \mathbb{R}^n : |c^\top x - d| \leq \rho\}$, define the cost vector $\bar{c} \in \mathbb{R}^{\bar{n}}$ in (3.110), where $\bar{n} = n + 1 + N(n_g, n-1)$
 - 2 Define the matrix $\bar{A} \in \mathbb{R}^{\bar{m} \times \bar{n}}$ in (3.111)-(3.115) and the vector $\bar{b} \in \mathbb{R}^{\bar{m} \times 1}$ in (3.116) as inequality constraints, where $\bar{m} = 2 + 2N(n_g, n-1)$
 - 3 Solve the LP (3.109) using some standard solver to obtain the optimal solution $\check{z} \in \mathbb{R}^{\bar{n}}$
 - 4 Define the desired vector $\check{x} = \check{z}_{1:n}$
 - 5 Compute the zonotope $\bar{\mathcal{X}} = \bar{G}^x \mathcal{B}^{(n_g+1)} \oplus \bar{x}$, whose center $\bar{x}$ and generator matrix $\bar{G}^x$ are given by (2.25) and (2.24), respectively
-

Both VM2 (presented in Subsection B.2.2.1) and VM3 can become unfeasible with the growth of n and n_g due to the mathematical combination $N(n_g, n)$. The VM2 suffers from the computational burden (since each candidate zonotope is costly evaluated by (1.7)), whereas the VM3 suffers from the storage capacity (since the combinatorial effect of (3.108) yields a high-order LP containing many inequality constraints). This issue can be improved exploiting the sparse nature of the problem. However, the LP may demand an expressive time to be built. In this case, the SM should be preferred, since it explicitly computes a unique zonotope. Therefore, the SM demands the smallest computational burden and processing time. For all numerical examples discussed in Subsection 3.2.1,

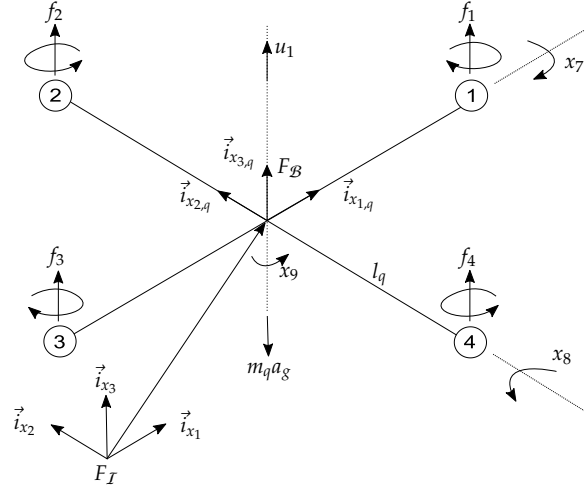


Figure 3.8: Illustrative diagram of a quadrotor. This UAV is assumed to be a rigid body, where the origin of the body frame F_B coincides with the center of mass. The symbols ①, ②, ③, and ④ represent the four coplanar rotors. Each rotor is distanced l_q from the center of mass, and the rotation of the i th rotor generates the i th thrust f_i , for $i = 1, \dots, 4$. The variable u_1 corresponds to the total thrust (that is, $u_1 = \sum_1^4 f_i$), while $m_q a_g$ corresponds to the weight force of the quadrotor UAV, with m_q being its total mass and a_g the gravitational acceleration. The notation $\vec{i}$ is used to indicate the increasing direction of each axis at a frame.

the volume minimization problem was solved using the VM3. In comparison with VM2, VM3 returned the same solutions with smaller T^{CPU} .

3.2.2 Quadrotor Unmanned Aerial Vehicle

Next, we propose a case study involving a quadrotor unmanned aerial vehicle (UAV). The scenario is designed to illustrate the application of the ZMV and IZMV, as well as the UKF and TUKF (see Appendix A), under a challenging and real application. Both UKF and TUKF are used for comparison purposes.

In Figure 3.8, a quadrotor UAV is illustrated. Its equations of motion correspond to a rotating rigid body with six degrees of freedom and twelve states (Mistler et al., 2001),

which are given by

$$\dot{x} = \begin{bmatrix} x_4 \\ x_5 \\ x_6 \\ \frac{u_1}{m_q} (\cos(x_9) \sin(x_8) \cos(x_7) + \sin(x_9) \sin(x_7)) + \frac{1}{m_q} w_1 \\ \frac{u_1}{m_q} (\sin(x_9) \sin(x_8) \cos(x_7) - \cos(x_9) \sin(x_7)) + \frac{1}{m_q} w_2 \\ -a_g + \frac{u_1}{m_q} \cos(x_8) \cos(x_7) + \frac{1}{m_q} w_3 \\ x_{10} + x_{11} \sin(x_7) \tan(x_8) + x_{12} \cos(x_7) \tan(x_8) \\ x_{11} \cos(x_7) - x_{12} \sin(x_7) \\ x_{11} \sin(x_7) \sec(x_8) + x_{12} \cos(x_7) \sec(x_8) \\ \frac{(I_{yy} - I_{zz})}{I_{xx}} x_{11} x_{12} + \frac{l_q}{I_{xx}} u_2 \\ \frac{(I_{zz} - I_{xx})}{I_{yy}} x_{10} x_{12} + \frac{l_q}{I_{yy}} u_3 \\ \frac{(I_{xx} - I_{yy})}{I_{zz}} x_{10} x_{11} + \frac{1}{I_{zz}} u_4 \end{bmatrix}, \quad (3.123)$$

where the continuous time t is omitted for brevity, with $[x_1 \ x_2 \ x_3]^\top$ being the linear position vector of the quadrotor with respect to the inertial frame F_I , $[x_4 \ x_5 \ x_6]^\top$ the linear speed vector expressed in the inertial frame F_I , $[x_7 \ x_8 \ x_9]^\top$ the Euler angles that describe the orientation of the quadrotor, and $[x_{10} \ x_{11} \ x_{12}]^\top$ the angular speed vector expressed in the body frame F_B . In addition, m_q , I_{xx} , I_{yy} , I_{zz} , and l_q are physical parameters, a_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, and u_4 is the difference of torques between clockwise and counter-clockwise turning propellers. Assume that the quadrotor flies in an environment whose known dimensions are $1.2 \times 1.2 \times 3.0 \text{ m}^3$. By means of a wind generator, exogenous forces $w = [w_1 \ w_2 \ w_3]^\top$ are applied to the UAV, such that their amplitudes are characterized by the GRV $W \sim \mathcal{N}(0_{3 \times 1}, \frac{1}{\varsigma_w} \mathbf{I}_3)$ bounded with 99.73% confidence level, where $\varsigma_w = 14.1563$ is the greatest realization of the chi-square variable with three degrees of freedom. In turn, the measurement model is given by

$$y_k = Cx_k + v_k, \quad (3.124)$$

where $C \in \mathbb{R}^{9 \times 12}$ is the output matrix whose rows individually select all states x_k , except the linear speed vector $[x_4 \ x_5 \ x_6]^\top$. In addition, v_k is the measurement noise whose realizations are characterized by the GRV $V \sim \mathcal{N}(0_{9 \times 1}, R)$ bounded by a box with the confidence level at least 99.73%, where

$$R = \frac{1}{\varsigma_v} \text{diag}(0.15^2 \times 1_{1 \times 2}, 0.51^2, 2.618^2 \times 10^{-6} 1_{1 \times 3}, 16.558^2 \times 10^{-6} 1_{1 \times 3})$$

and $\varsigma_v = 25.2567$ is the greatest realization of the chi-square variable with nine degrees of freedom. These values are defined based on [Rego et al. \(2020a\)](#), where a realistic scenario is considered.

It is desired that the quadrotor tracks a vertical helix trajectory defined by the reference values $x_1^{\text{tr}}(t) = \frac{1}{2} \cos\left(\frac{t}{2}\right)$, $x_2^{\text{tr}}(t) = \frac{1}{2} \sin\left(\frac{t}{2}\right)$, $x_3^{\text{tr}}(t) = 1 + \frac{t}{10}$, and $x_9^{\text{tr}} = \frac{\pi}{3}$, subject to the noise forces w_1 for $t \in [5, 15)$ s, w_2 for $t \in [8, 15)$ s, and w_3 for $t \in [10, 15)$ s; these wind forces are null otherwise. To control the system, the feedback linearization strategy proposed by [Mistler et al. \(2001\)](#) is used. However, the gains of the controller are defined according to the design of a linear-quadratic regulator (LQR). To reach a smooth performance, the infinite-horizon LQR weighting matrices are selected as $Q^y = \mathbf{I}_4$ and $R^u = 1 \times 10^{-3} \mathbf{I}_4$. The model parameters are set as $m_q = 0.7$ kg, $l_q = 0.3$ m, $I_{xx} = I_{yy} = I_{zz} = 1.2416$ kg $\times$ m², and $a_g = 9.81$ m/s².

The proposed scenario allows for setting the interval state constraints (3.106) with

$$\begin{aligned} a_k &= [-0.6 \ -0.6 \ 0 \ -0.5 \ -0.5 \ -0.5 \ -\pi/70 \ -\pi/70 \ \pi/3.1 \ -1 \ -1 \ -0.03]^\top \\ b_k &= [0.6 \ 0.6 \ 3 \ 0.5 \ 0.5 \ 0.5 \ \pi/70 \ \pi/70 \ \pi/2.9 \ 1 \ 1 \ 0.03]^\top. \end{aligned}$$

These constraints concern space, speed, and angle limitations to be fulfilled by the UAV. The system is simulated with the initial state $x(0) = [0.5 \ 0 \ 1 \ 0_{1 \times 5} \ \frac{\pi}{3} \ 0_{1 \times 3}]^\top$ and $T_s = 0.01$ s. Moreover, the Euler approximation is used to discretize the continuous dynamics (3.123).

3.2.2.1 Setup

We apply the UKF, TUKF, ZMV, and IZMV to estimate the states of the quadrotor UAV. Direct comparisons are performed among algorithms of same approach. Regarding different approaches, possible limitations and drawbacks of algorithms are discussed.

For the stochastic algorithms, the initial state estimate is set as $\hat{x}_0 = 0_{12 \times 1}$ with covariance

$$\hat{P}_0^{\text{xx}} = \frac{1}{\varsigma_x} \text{diag}\left(2^2, 2^2, 2^2, 1, 1, 1, \frac{\pi^2}{6^2}, \frac{\pi^2}{6^2}, \frac{\pi^2}{2^2}, \frac{\pi^2}{12^2}, \frac{\pi^2}{12^2}, \frac{\pi^2}{12^2}\right),$$

where $\varsigma_x = 30.097$ is the greatest realization of the chi-square variable with 12 degrees of freedom. The 99.73% confidence level is the criterion to compensate the discretization error. To achieve that, first define the additive GRV $\bar{W}_{k-1} \sim \mathcal{N}(0_{12 \times 1}, \bar{Q}_{k-1})$ with $\bar{Q}_{k-1} = 0_{12 \times 12}$. Second, execute some Monte Carlo simulations and verify if the true states x_k are included in ellipsoids, which are defined according to the *a posteriori* estimates $\hat{x}_k$ and $\hat{P}_k^{\text{xx}}$. If most of true states x_k are not included in ellipsoids, then the covariance matrix $\bar{Q}_{k-1}$ is slightly enlarged over test simulations. In doing so, the covariance matrix $\bar{Q}_{k-1} = \text{diag}(1 \times 10^{-4} \mathbf{1}_{1 \times 3}, 3 \times 10^{-4} \mathbf{1}_{1 \times 2}, 4 \times 10^{-4}, 1 \times 10^{-4} \mathbf{1}_{1 \times 6})$ is set.

For the zonotopic algorithms, the initial state estimate is set as $\hat{x}_0 = 0_{12 \times 1}$ with generator

matrix

$$\hat{G}_0^x = \text{diag}\left(2, 2, 2, 1, 1, 1, \frac{\pi}{6}, \frac{\pi}{6}, \frac{\pi}{2}, \frac{\pi}{12}, \frac{\pi}{12}, \frac{\pi}{12}\right),$$

such that $x_0 \in \hat{G}_0^x \mathcal{B}^{12} \oplus \hat{x}_0$. No additional noise is applied to model the discretization error. The noise terms w_{k-1} and v_k are characterized by the zonotopes $\mathcal{W}_{k-1} = G_{k-1}^w \mathcal{B}^3 \oplus 0_{3 \times 1}$ and $\mathcal{V}_k = G_k^v \mathcal{B}^9 \oplus 0_{9 \times 1}$, respectively, where $G_{k-1}^w = I_3$ and

$$G_k^v = \text{diag}(0.15 \times 1_{1 \times 2}, 0.51, 2.618 \times 10^{-3} 1_{1 \times 3}, 16.558 \times 10^{-3} 1_{1 \times 3}).$$

The reduction order $\varphi_g = 500$ is chosen based on offline simulations, when the order reduction procedure does not modify the Frobenius norm significantly. Both ZMV and IZMV are performed together with the DA approach (Algorithm D.2).

3.2.2.2 Numerical Results

The outputs of the UKF and TUKF are presented in Figure 3.9. The UKF yields the green ellipsoids, while the TUKF yields the red ellipsoids. All ellipsoids are illustrated with 99.73% confidence level. There still exists non-null probability of the true states x_k being outside the ellipsoids. In this application, the initial state estimate is close to the actual state x_0 . A slight difference can be realized between the UKF and TUKF when more than one dimension is seen at the same time. To illustrate the different performance of the two algorithms, Figure 3.9(c) indicates that the TUKF continually corrects its covariance estimates. This also occurs with the other linear speed states. It is illustrated then that for the same confidence level, the TUKF returns more precise estimates (ellipsoids with smaller volumes).

The outputs of the ZMV and IZMV are depicted in Figure 3.10. Both ZMV and IZMV estimate zonotopes ($\hat{X}_k$ and $\check{X}_k$ respectively) guaranteeing the true state x_k is included within the corresponding zonotope. The impact of enforcing interval constraints through IZMV is illustrated in Figures 3.10a, 3.10c, and 3.10d. Note that the blue boxes provided by the IZMV present smaller volume and Frobenius norm than the cyan boxes provided by the ZMV; these precision metrics are improved thanks to the intersection operation and segment minimization, respectively. However, (as in Figure 3.6) the segment minimization (Algorithm B.7) may yield low-precision solutions. This is the reason why the blue boxes still exceed the constraint box (dash-dotted box). In Figure 3.10(d), the Frobenius norm is computed for each *a posteriori* generator matrix. Note that, when the ZR (step 13 of Algorithm B.13) is used, the Frobenius norm starts to grow. This growth is more significant when $\varphi_g < 500$. When $\varphi_g > 500$, the Frobenius norm tends to the ideal scenario, where the norm converges to the smallest possible value. In this application, $\varphi_g = 500$ does not significantly modify the performance of the algorithms.

Additionally, the center estimates $\hat{x}_k$ (cyan dashed line) (ZMV) and $\check{x}_k$ (blue dotted

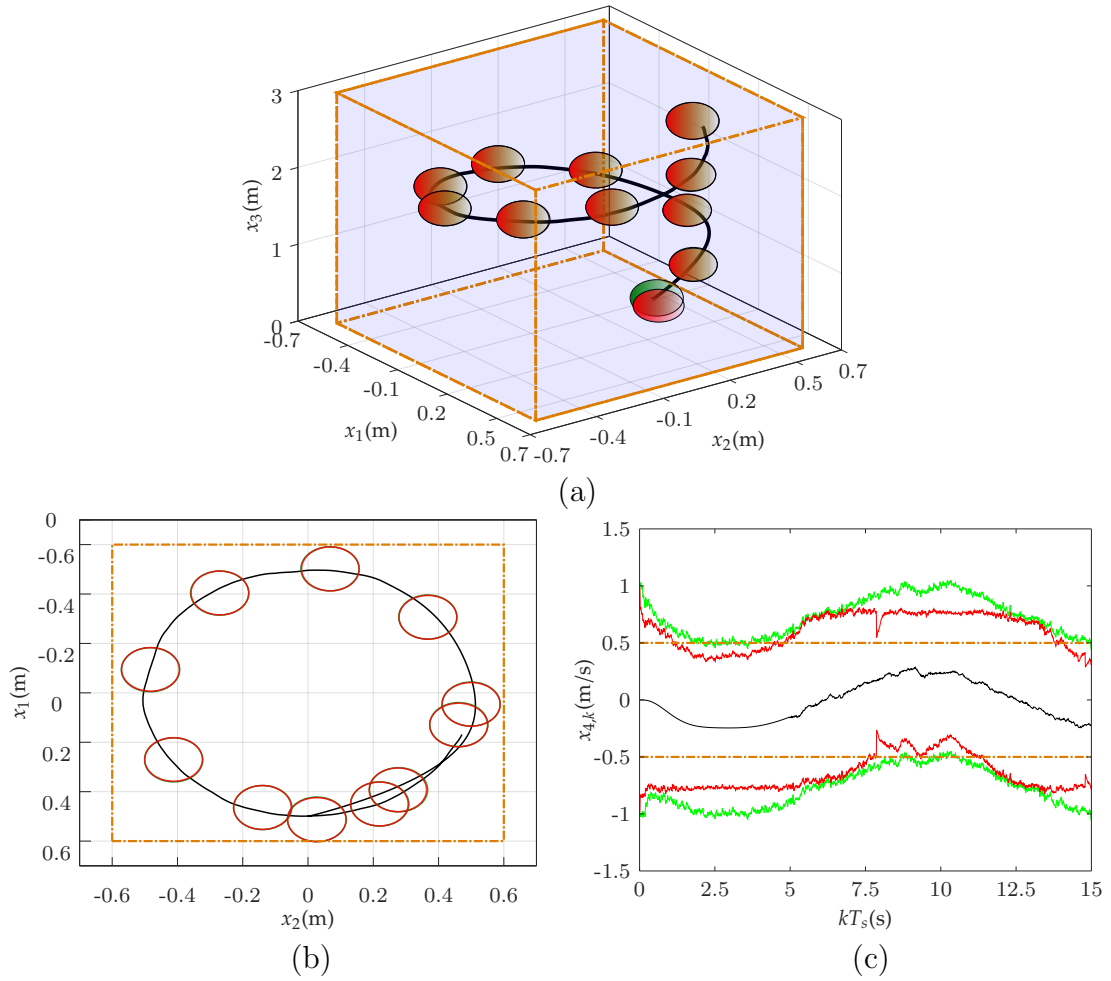


Figure 3.9: State estimation for the quadrotor UAV using the unscented Kalman filter (UKF) and truncated unscented Kalman filter (TUKF). In (a), the true states (black solid line), related to the linear position, are approximated by ellipsoids with 99.73% confidence level. The green and red ellipsoids represent the estimates provided by the UKF and TUKF, respectively. The large box containing both true states and ellipsoids is the available workspace for the quadrotor. In (b), the state constraints and ellipsoids are projected in 2D. The projections of UKF and TUKF almost coincide. In (c), only the confidence interval for the state x_4 is illustrated. In this case, the TUKF (red solid line) corrects its estimates based on the interval constraints, while the UKF (green solid line) does not.

line) (IZMV) are illustrated in Figure 3.10(b). When comparing the center of both ZMV and IZMV to the true states, $\hat{x}_k$ seems to be more accurate than $\check{x}_k$, despite the improved precision of IZMV. It means that, after the intersection of the zonotope $\hat{X}_k$ with the state interval constraints $\mathcal{R}_k = [a_k, b_k]$, the resulting zonotope $\check{X}_k$ has its center moved away from the constraint boundaries. Therefore, depending on the use of the estimated set, an appropriate metric should be used to choose a particular punctual state of the set rather than its center, aiming at mitigating the eventual loss of accuracy in applications that require a punctual estimate. Likewise, it is known that, for the stochastic approach, the mode of the *a posteriori* PDF may be chosen over the mean in some applications. For

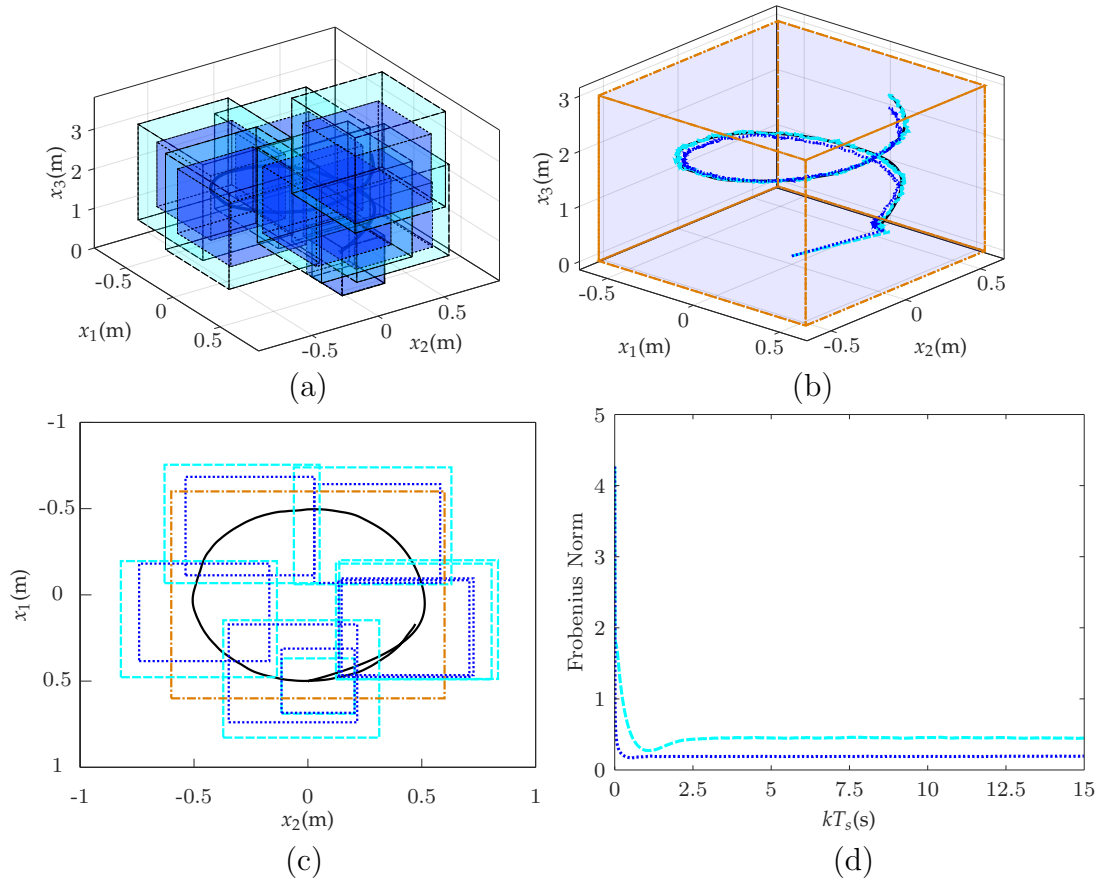


Figure 3.10: State estimation for the quadrotor UAV using the zonotopic filter based on mean value extension (ZMV) and its interval extension (IZMV) with segment minimization (Algorithm B.7). The number of generators of zonotopes is fixed by the reduction order $\varphi_g = 500$. In (a), the true states (black solid line), related to the linear position, are illustrated together with the interval hulls of zonotopes provided by the ZMV (cyan dashed boxes) and IZMV (blue dotted boxes). This sketch notes the improvement reached by the IZMV in relation to the ZMV, given that interval state constraints are enforced. In (b), the true states are illustrated together with center estimates. The large box containing both true states and center estimates is the available workspace for the quadrotor. This graph indicates the ZMV yields the most accurate centers. In (c), the prior boxes are illustrated in 2D, to highlight that the blue boxes can still exceed the constraints, since the segment minimization yields low-precision solutions. In (d), the Frobenius norm is used as criterion to quantify the precision of zonotopes.

instance, an alternative possibility is to formulate an optimization problem as in [Rego et al. \(2020a, Proposition 1\)](#) to find the optimal point (in the 1-norm sense) interior to $\check{\mathbf{X}}_k$ that is the closest point to the center $\hat{\mathbf{x}}_k$ returned in step 1 of Algorithm 3.6.

Consider again Figures 3.9 and 3.10, whose results were obtained using an unconstrained control law. These illustrate two different types of state estimates: confidence ellipsoids and boxes. The former provides a probability measure to estimate the true states, while the latter guarantees that the true states are included within. Also, these results note that the mean estimate of Kalman estimators may be a suitable option to feed controllers based on state feedback, as in this case study, as well the center of the ZMV.

On the other hand, the center of the IZMV is not the best choice, due to its displacement away from the state constraints.

3.2.3 Set-Membership Filter based on CZs and DC Programming

Consider the problem formulation (1.1)-(1.4), where the terms w_{k-1} , v_k , and x_0 are characterized by the CZs $\mathcal{W}_{k-1} = \{G_{k-1}^w, c_{k-1}^w, A_{k-1}^w, b_{k-1}^w\}$, $\mathcal{V}_k = \{G_k^v, c_k^v, A_k^v, b_k^v\}$, and $\mathcal{X}_0 = \{G_0^x, c_0^x, A_0^x, b_0^x\}$, respectively. Regarding (1.4), we make the following assumption to enable the direct use of convex polytopes.

Assumption 3.2.1. The inequality constraints given by (1.4), if present, define a compact feasibility set $\mathcal{X}_k^F \subset \mathbb{R}^n$.

Our set-membership filter aims at estimating the state vector x_k using CZs $\mathcal{X}_k$ over $k \in \mathbb{Z}_+$. To achieve this goal at each k , given u_{k-1} , we define five steps (similar to Rego et al. (2021)) as follows:

1. *Forecast:* $\mathcal{X}_{k|k-1} \supseteq \{f(x_{k-1}, u_{k-1}, w_{k-1}) : x_{k-1} \in \mathcal{X}_{k-1}, w_{k-1} \in \mathcal{W}_{k-1}\}$;
2. *Data assimilation:* $\check{\mathcal{X}}_k \supseteq \{x_k \in \mathcal{X}_{k|k-1} : h(x_k, v_k) = y_k, v_k \in \mathcal{V}_k\}$;
3. *Admissibility:* $\check{\mathcal{X}}_k = \check{\mathcal{X}}_k \cap \mathcal{X}_k^F$;
4. *Consistency:* $\tilde{\mathcal{X}}_k \supseteq \{x_k \in \check{\mathcal{X}}_k : g(x_k) = 0_{m_c \times 1}\}$;
5. *Reduction:* $\mathcal{X}_k \supset \tilde{\mathcal{X}}_k$, with $\mathcal{X}_k$ being a set with lower complexity than $\tilde{\mathcal{X}}_k$.

Next, we present the results to execute a loop of the novel set-membership filter based on CZs and DC programming, called CZDC. This algorithm extends the constrained zonotopic linear filter (CZLF) discussed in Subsection B.1.4 using DC programming to work on nonlinear systems and on state constraints. The general idea is to firstly linearize models. Then, operations (2.27)-(2.30) are performed on the linearized models. Finally, Lemma 2.3.1 is employed to enclosure the linearization error. The CZDC composes the paper de Paula et al. (2022d), which has been extended in Tottoli et al. (2022) to combine DC programming with mean value extension.

Theorem 3.2.1 (Forecast Step). Consider the CZs $\mathcal{X}_{k-1} \subset \mathbb{R}^n$ and $\mathcal{W}_{k-1} \subset \mathbb{R}^q$, and the deterministic input $u_{k-1} \in \mathbb{R}^p$. Let $f : \mathbb{R}^n \times \mathbb{R}^p \times \mathbb{R}^q \rightarrow \mathbb{R}^n$ (1.1) be rewritten as $\varrho^f : \mathbb{R}^{(n+q)} \times \mathbb{R}^p \rightarrow \mathbb{R}^n$ using the augmented vector $z_{k-1} = \begin{bmatrix} x_{k-1}^\top & w_{k-1}^\top \end{bmatrix}^\top$. Let also $\varrho^f = \varrho^{fa} - \varrho^{fb}$ be DC on the polytope $\mathcal{P}_{k-1} \supseteq \mathcal{Z}_{k-1} = \mathcal{X}_{k-1} \times \mathcal{W}_{k-1}$. Finally, let $\mathcal{R}_{k-1} \subset \mathbb{R}^n$ be the set returned by Lemma 2.3.1 to enclose the linearization error of ϱ^f for a given punctual estimate $\bar{z} \in \mathcal{P}_{k-1}$. Then, the exact image $\varrho^f(\mathcal{Z}_{k-1}, u_{k-1})$ is outer-approximated by the CZ

$$\mathcal{X}_{k|k-1} = (\varrho^f(\bar{z}, u_{k-1}) - F\bar{z}) \oplus F\mathcal{Z}_{k-1} \oplus \mathcal{R}_{k-1}, \quad (3.125)$$

with $F = \nabla_z \varrho^f(\bar{z}, u_{k-1})$ being the Jacobian matrix evaluated at $\bar{z} = [\bar{x}^\top \quad \bar{w}^\top]^\top$.

Proof. This proof is similar to Alamo et al. (2008, Proof of Theorem 1), with the difference being the propagation of CZs instead of zonotopes, and the introduction of deterministic input u_{k-1} . ■

Theorem 3.2.2 (Data-Assimilation Step). Consider the CZs $\mathcal{X}_{k|k-1} \subset \mathbb{R}^n$ and $\mathcal{V}_k \subset \mathbb{R}^r$, and the measured output $y_k \in \mathbb{R}^m$. Let $h : \mathbb{R}^n \times \mathbb{R}^r \rightarrow \mathbb{R}^m$ (1.2) be rewritten as $\varrho^h : \mathbb{R}^{(n+r)} \rightarrow \mathbb{R}^m$ using the augmented vector $z_k = [x_k^\top \quad v_k^\top]^\top$. Let also $\varrho^h = \varrho^{ha} - \varrho^{hb}$ be DC on the polytope $\mathcal{P}_k \supseteq \mathcal{Z}_k = \mathcal{X}_{k|k-1} \times \mathcal{V}_k$. Finally, let $\mathcal{R}_k \subset \mathbb{R}^m$ be the set returned by Lemma 2.3.1 to enclose the linearization error of ϱ^h for a given punctual estimate $\bar{z} \in \mathcal{P}_k$. Then, the exact set $\{x_k \in \mathcal{X}_{k|k-1} : y_k = h(x_k, v_k), v_k \in \mathcal{V}_k\}$ is overapproximated by the CZ

$$\check{\mathcal{X}}_k = \mathcal{X}_{k|k-1} \cap_{H^\times} \mathcal{Y}_k, \quad (3.126)$$

where $\mathcal{Y}_k = (y_k - \varrho^h(\bar{z}) + H\bar{z}) \oplus (-H^\vee \mathcal{V}_k) \oplus (-\mathcal{R}_k)$, with $H = [H^\times \quad H^\vee]$, $H^\times = \nabla_x \varrho^h(\bar{z})$, and $H^\vee = \nabla_v \varrho^h(\bar{z})$.

Proof. Let

$$y_k = \varrho^h(\bar{z}) + H(z_k - \bar{z}) + e_k^h$$

be the analytical linearization of the DC function $\varrho^h = \varrho^{ha} - \varrho^{hb}$ on $\mathcal{P}_k$, and let the CZ $\mathcal{R}_k \ni e_k^h$ be the linearization error enclosure given by Lemma 2.3.1. By making explicit the term $H^\times x_k$ from $H z_k = [H^\times \quad H^\vee] \begin{bmatrix} x_k \\ v_k \end{bmatrix}$, we obtain $H^\times x_k = y_k - \varrho^h(\bar{z}) + H\bar{z} - H^\vee v_k - e_k^h$, which implies the CZ $\mathcal{Y}_k = (y_k - \varrho^h(\bar{z}) + H\bar{z}) \oplus (-H^\vee \mathcal{V}_k) \oplus (-\mathcal{R}_k)$. Then, we employ the generalized intersection (2.29) to match $\mathcal{X}_{k|k-1}$ with $\mathcal{Y}_k$, yielding $\check{\mathcal{X}}_k$. ■

Remark 3.2.1. If functions f and h are affine in the noise terms w_{k-1} and v_k , respectively, then these terms are canceled during the computation of $\mathcal{R}$ in Theorems 3.2.1 and 3.2.2. It means that, instead of $2^{(n+q)}$ and $2^{(n+r)}$ vertices, we need to process 2^n vertices only.

The consistency step enforces the nonlinear invariants described in (1.3). In algebraic terms, (1.3) is a special case of (1.2). Then, consistency step is a direct consequence from Theorem 3.2.2, being next presented as a corollary. For brevity, the admissibility step is presented together with the consistency one.

Corollary 3.2.1 (Consistency Step). Consider the CZ $\check{\mathcal{X}}_k \subset \mathbb{R}^n$ (Theorem 3.2.2) and the feasible set $\mathcal{X}_k^F \subset \mathbb{R}^n$. Let $\check{\mathcal{X}}_k = \check{\mathcal{X}}_k \cap_{I_n} \mathcal{X}_k^F$ be the admissible set. Let $g : \mathbb{R}^n \rightarrow \mathbb{R}^{m_c}$ (1.3) be rewritten as $g = g^a - g^b$, where g^a and g^b are convex functions in the polytope $\check{\mathcal{P}}_k \supseteq \check{\mathcal{X}}_k$. Let also $\mathcal{R}_k \subset \mathbb{R}^{m_c}$ be the set returned by Lemma 2.3.1 to enclose the linearization error of g for a given punctual estimate $\bar{x} \in \check{\mathcal{P}}_k$. Then, the exact set $\{x_k \in \check{\mathcal{X}}_k : g(x_k) = 0_{m_c \times 1}\}$ is overapproximated by the CZ

$$\tilde{\mathcal{X}}_k = \check{\mathcal{X}}_k \cap_H \mathcal{C}_k, \quad (3.127)$$

where $H = \nabla_x g(\bar{x})$ and $C_k = (-g(\bar{x}) + H\bar{x}) \oplus (-\mathcal{R}_k)$.

Proof. This proof is similar to the proof of Theorem 3.2.2, whose difference is the replacement of y_k , h , and z_k by $0_{m_c \times 1}$, g , and x_k , respectively. ■

We summarize the steps of CZDC in Algorithm 3.8. We recall that DC programming is motivated to mitigate the wrapping and dependency effects related to interval arithmetic, but its computational challenge (control and manipulation of vertices) is transmitted to CZDC.

Algorithm 3.8: $\mathcal{X}_k = \text{CZDC}(f, f^a, f^b, \mathcal{X}_{k-1}, u_{k-1}, \mathcal{W}_{k-1}, y_k, h, h^a, h^b, \mathcal{V}_k, g, g^a, g^b, \mathcal{X}_k^F, \varphi_c, \varphi_g)$

- 1 Apply Theorem 3.2.1 to obtain the CZ $\mathcal{X}_{k|k-1}$
 - 2 Apply Theorem 3.2.2 to obtain the CZ $\check{\mathcal{X}}_k$
 - 3 Compute $\check{\mathcal{X}}_k = \check{\mathcal{X}}_k \cap_{\text{In}} \mathcal{X}_k^F$
 - 4 Apply Corollary 3.2.1 to obtain the CZ $\tilde{\mathcal{X}}_k$
 - 5 Apply CZR given by (2.31) to reduce the number of constraints n_h and generators n_g of $\tilde{\mathcal{X}}_k$ to φ_c and φ_g , respectively, yielding the CZ $\mathcal{X}_k$
-

3.2.3.1 Complexity Analysis

The worst-case computational complexity $O(\cdot)$ for each step of CZDC (Algorithm 3.8) is shown in Table 3.7. Such complexities were derived using basic operations among CZs (Rego et al., 2020a). Regarding the forecast, data assimilation, and consistency steps, the complexity order to obtain the linearization point $\bar{z}$ is not included since it depends on the employed methodology. As in Rego et al. (2020a), we also assume that the evaluation of nonlinear functions has complexity $O(1)$. In the second column of Table 3.7, the cubic term between parenthesis refers to either the computation of parallelotope via LPs or the order reduction. The term $2^{\bar{n}}$ is related to either the computation of vertices or the DC programming problems (2.40)-(2.41). In turn, the third column of Table 3.7 presents the amount of constraints and generators for the state CZ $\mathcal{X}$ over the different steps.

Table 3.7 makes the following assumptions: $\mathcal{X}_{k-1} = \{G_{k-1}^x, c_{k-1}^x, A_{k-1}^x, b_{k-1}^x\} \subset \mathbb{R}^n$, $\mathcal{W}_{k-1} = \{G_{k-1}^w, c_{k-1}^w, A_{k-1}^w, b_{k-1}^w\} \subset \mathbb{R}^q$, $\mathcal{V}_k = \{G_k^v, c_k^v, A_k^v, b_k^v\} \subset \mathbb{R}^r$, and $\mathcal{X}_k^F = \{G_k^{xF}, c_k^{xF}, A_k^{xF}, b_k^{xF}\} \subset \mathbb{R}^n$, where $G_{k-1}^x \in \mathbb{R}^{n \times n_g}$, $G_{k-1}^w \in \mathbb{R}^{q \times n_g^w}$, $G_k^v \in \mathbb{R}^{r \times n_g^v}$, $G_k^{xF} \in \mathbb{R}^{n \times n_g^{xF}}$, $c_{k-1}^x \in \mathbb{R}^n$, $c_{k-1}^w \in \mathbb{R}^q$, $c_k^v \in \mathbb{R}^r$, $c_k^{xF} \in \mathbb{R}^n$, $A_{k-1}^x \in \mathbb{R}^{n_h \times n_g}$, $A_{k-1}^w \in \mathbb{R}^{n_h^w \times n_g^w}$, $A_k^v \in \mathbb{R}^{n_h^v \times n_g^v}$, $A_k^{xF} \in \mathbb{R}^{n_h^{xF} \times n_g^{xF}}$, $b_{k-1}^x \in \mathbb{R}^{n_h}$, $b_{k-1}^w \in \mathbb{R}^{n_h^w}$, $b_k^v \in \mathbb{R}^{n_h^v}$, and $b_k^{xF} \in \mathbb{R}^{n_h^{xF}}$. These sets are evaluated over the functions $f : \mathbb{R}^n \times \mathbb{R}^p \times \mathbb{R}^q \rightarrow \mathbb{R}^n$, $h : \mathbb{R}^n \times \mathbb{R}^r \rightarrow \mathbb{R}^m$, and $g : \mathbb{R}^n \rightarrow \mathbb{R}^{m_c}$, considering the vectors $u_{k-1} \in \mathbb{R}^p$ and $y_k \in \mathbb{R}^m$. At the end of an iteration of CZDC, the desired CZ $\mathcal{X}_k$ is returned with φ_c constraints and φ_g generators.

Remark 3.2.2. According to Table 3.7, the output CZs obtained by Theorems 3.2.1-3.2.2 and Corollary 3.2.1 have smaller number of constraints and generators than those pointed out by Remarks 5, 7, and 10 from Rego et al. (2021). Exceptionally, the number of

Table 3.7: Complexity order of the forecast, data assimilation, admissibility, consistency, and reduction steps from CZDC using Proposition 2.3.2.

Step	$O(\cdot)$	Definition
Forecast	$\tilde{n}_h (\tilde{n}_h + \tilde{n}_g)^3 + \tilde{n} (\tilde{n} + \tilde{n}_g) (\tilde{n} + \tilde{n}_h + \tilde{n}_g)^3 + \tilde{n}^2 2^{\tilde{n}}$	$\tilde{n} \triangleq n + q, \tilde{n}_h \triangleq n_h + n_h^W, \tilde{n}_g \triangleq n_g + n_g^W$
Data Assimilation	$\tilde{n}_h (\tilde{n}_h + \tilde{n}_g)^3 + \tilde{n} (\tilde{n} + \tilde{n}_g) (\tilde{n} + \tilde{n}_h + \tilde{n}_g)^3 + (\tilde{n}^2 + m) 2^{\tilde{n}} + mn n_g^V + mn (\tilde{n}_g - n_g^V) + m\tilde{n} + m^2$	$\tilde{n} \triangleq n + r, \tilde{n}_h \triangleq n_h + n_h^W + n_h^V, \tilde{n}_g \triangleq n_g + n_g^W + n + n_g^V$
Admissibility	$n^2 \tilde{n}_g + mn_g^F$	$\tilde{n}_g \triangleq n_g + n_g^W + n + n_g^V + m$
Consistency	$\tilde{n}_h (\tilde{n}_h + \tilde{n}_g)^3 + n (n + \tilde{n}_g) (n + \tilde{n}_h + \tilde{n}_g)^3 + (n^2 + m_c) 2^n + m_c n \tilde{n}_g + m_c^2$	$\tilde{n}_h \triangleq n_h + n_h^W + n_h^V + m + n_h^{X^F} + n$ $\tilde{n}_g \triangleq n_g + n_g^W + n + n_g^V + m + n_g^{X^F}$
Reduction	$k_c (\tilde{n}_h + \tilde{n}_g)^3 + k_c n \tilde{n}_g^2 + (n + \varphi_c)^2 (\tilde{n}_g - k_c) + k_g (\tilde{n}_g - k_c) (n + \varphi_c)$	$k_c \triangleq \tilde{n}_h - \varphi_c, k_g \triangleq \tilde{n}_g - k_c - \varphi_g,$ $\tilde{n}_h \triangleq n_h + n_h^W + n_h^V + m + n_h^{X^F} + n + m_c,$ $\tilde{n}_g \triangleq n_g + n_g^W + n + n_g^V + m + n_g^{X^F} + m_c$

constraints for $\mathcal{X}_{k|k-1}$ coincides with the value indicated in Rego et al. (2021, Remark 5) for the CZMV algorithm.

3.2.4 Numerical Examples

Next, CZDC is experimented over two case studies. For comparison purposes, we also implement the state-of-the-art algorithms proposed in Rego et al. (2021), called CZFO (based on Taylor expansion) and CZMV (based on mean value extension). To yield punctual estimates $\bar{z}$, and thereby, to approximate the nonlinear models, we make the following choices: CZDC uses the center of the polytope $\mathcal{P}$ associated to Lemma 2.3.1, where $\mathcal{P}$ is a box or a parallelotope, whose procedure is $O(1)$; CZFO is run with metric Rego et al. (2021, C3) to minimize the diameter of an interval matrix; CZMV is run with metric Rego et al. (2021, C2) to minimize the diameter of an interval vector. Two performance indexes are computed: T^{CPU} in (3.57) and $A^\square$ in (3.83). The noise terms w_{k-1} and v_k are taken from uniform distributions defined in $\mathcal{W}_{k-1}$ and $\mathcal{V}_k$, while the initial state x_0 belongs to the initial set $\mathcal{X}_0$. The following computer configuration was used: 8 GB RAM 1333 MHz, Windows 10 Pro, and AMD FX-6300 CPU 3.50 GHz. All implementations were executed in MATLAB 9.11 (Release) with INTLAB 12 (Rump, 1999), MPT3 (Herceg et al., 2013), and Gurobi 9.1. Specifically, INTLAB is employed to compute interval estimates, MPT is used to sketch sets, and Gurobi is used to solve LPs.

Since the measurement y_0 is available, all three algorithms execute a first loop with $\mathcal{X}_{k|k-1} = \mathcal{X}_0$, whose goal is to improve the precision of the starting set $\mathcal{X}_0$. Soon after, the state estimators are normally executed. For all examples, CZFO employs order reduction with fixed values φ_c and φ_g at the end of each step (with exception of the admissibility step), as recommended in Rego et al. (2021).

3.2.4.1 Two-State Nonlinear Process

Consider the discrete-time system (Alamo et al., 2008)

$$\begin{bmatrix} x_{1,k} \\ x_{2,k} \end{bmatrix} = \begin{bmatrix} -0.7x_{2,k-1} + 0.1x_{2,k-1}^2 + 0.1x_{1,k-1}x_{2,k-1} + 0.1\exp(x_{1,k-1}) \\ x_{1,k-1} + x_{2,k-1} - 0.1x_{1,k-1}^2 + 0.2x_{1,k-1}x_{2,k-1} \end{bmatrix} + \begin{bmatrix} w_{1,k-1} \\ w_{2,k-1} \end{bmatrix}, \quad (3.128)$$

$$y_k = \begin{bmatrix} 1 & 1 \end{bmatrix} x_k + v_k, \quad (3.129)$$

where $w_{k-1} \in \mathcal{W} = \{0.1I_2, 0_{2 \times 1}\}$ and $v_k \in \mathcal{V} = \{0.2, 0\}$. To simulate this system, we set $x_0 = \begin{bmatrix} 1 & 1 \end{bmatrix}^\top \in \mathcal{X}_0 = \{3 \times I_2, 0_{2 \times 1}\}$, $k_f = 40$, and $m_s = 100$. This example aims at illustrating that CZDC is a promising option to substitute the use of CZFO and CZMV whenever the wrapping and dependency effects imply divergence of estimates, and that CZDC reaches a better precision than the zonotopic filter based on DC programming (ZDC) proposed in Alamo et al. (2008). To reduce order of CZs, we set $\varphi_c = 3$ and $\varphi_g = 8$. This latter value is also used to reduce order of zonotopes in ZDC with Method 4 of Yang & Scott (2018). To improve both the computational efficiency and the precision of the minimum-volume zonotopes computed in ZDC, we employ VM3 (Algorithm 3.7) and Bravo et al. (2006, Definition 8). Motivated by Alamo et al. (2008), we propose the DC function $x_k = f^a - f^b + w_{k-1}$ such that

$$\begin{aligned} f^a &= \begin{bmatrix} x_{k-1}^\top A_1 x_{k-1} \\ x_{k-1}^\top A_2 x_{k-1} \end{bmatrix} + \begin{bmatrix} 0.1\exp(x_{1,k-1}) \\ x_{1,k-1} + x_{2,k-1} \end{bmatrix} = \begin{bmatrix} 0.1x_{1,k-1}^2 + 0.1x_{1,k-1}x_{2,k-1} + 0.1x_{2,k-1}^2 + 0.1\exp(x_{1,k-1}) \\ 0.1x_{2,k-1}^2 + x_{1,k-1} + x_{2,k-1} \end{bmatrix}, \\ f^b &= \begin{bmatrix} x_{k-1}^\top B_1 x_{k-1} \\ x_{k-1}^\top B_2 x_{k-1} \end{bmatrix} + \begin{bmatrix} 0.7x_{2,k-1} \\ 0 \end{bmatrix} = \begin{bmatrix} 0.1x_{1,k-1}^2 + 0.7x_{2,k-1} \\ 0.1x_{1,k-1}^2 + 0.1x_{2,k-1}^2 - 0.2x_{1,k-1}x_{2,k-1} \end{bmatrix}, \end{aligned}$$

with A_1 , A_2 , B_1 , and B_2 being positive semidefinite matrices (not necessarily symmetric). Then, there exist different choices of matrices to ensure that f^a and f^b are convex. Although Hessian matrices could be here chosen as candidate, we present the following choices:

$$A_1 = \begin{bmatrix} 0.1 & 0.1 \\ 0 & 0.1 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 0 & 0 \\ 0 & 0.1 \end{bmatrix}, \quad B_1 = \begin{bmatrix} 0.1 & 0 \\ 0 & 0 \end{bmatrix}, \quad B_2 = \begin{bmatrix} 0.1 & -0.2 \\ 0 & 0.1 \end{bmatrix},$$

whose eigenvalues are nonnegative. Since DC functions were directly defined, Proposition 2.3.3 was not employed, and thereby, the polytope $\mathcal{P}_{k-1}$ associated to Theorem 3.2.1 is a parallelotope (given by Proposition 2.3.2) that contains the CZ $\mathcal{Z}_{k-1} = \mathcal{X}_{k-1}$.

In Figure 3.11(a), we point out that both CZFO and CZMV diverge due to the direct usage of interval arithmetic. In this case, both wrapping and dependency effects are present and lead to large forecasts. Therefore, the application of measurements is not enough to correct the enlargement of CZs over iterations, justifying why those algorithms diverge. Although the natural interval extension was used to experiment CZFO and CZMV in Rego et al. (2021), it is not enough to reach convergence in this case study.

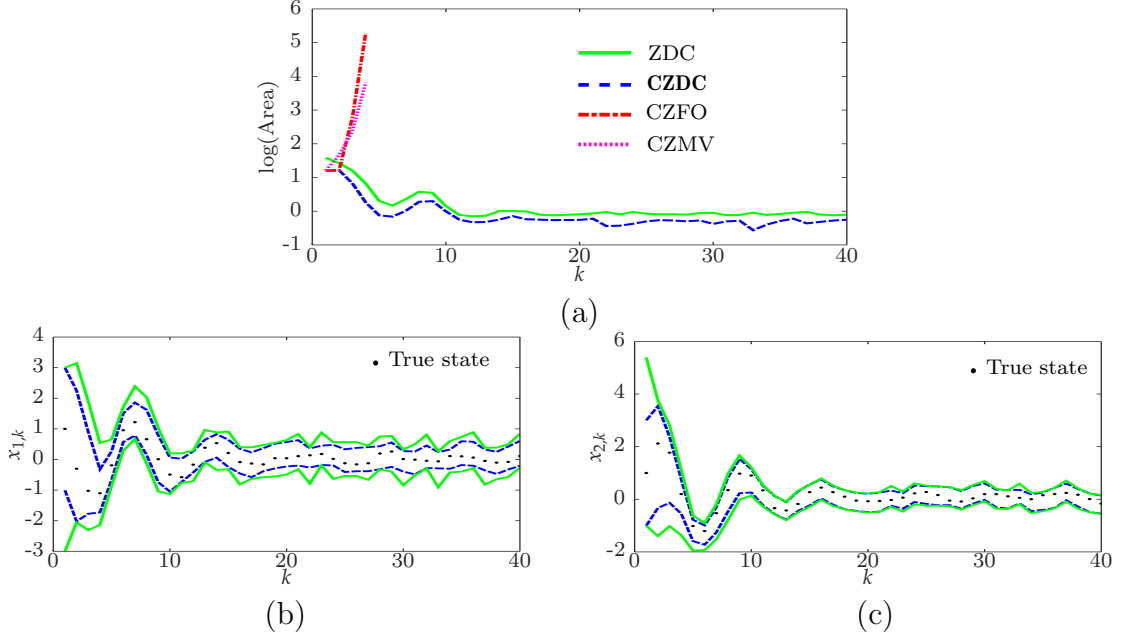


Figure 3.11: State estimation for the first case study (Subsection 3.2.4.1). Graph (a) depicts the time evolution of the area of boxes computed by ZDC, CZDC, CZFO, and CZMV. In (b) and (c), true states are involved by interval hulls of CZs computed by ZDC and CZDC.

Table 3.8: Results of T^{CPU} and $A^\square$ for the first example (Subsection 3.2.4.1).

Indexes	ZDC	CZDC
T^{CPU}	7.90 ms	12.6 ms ($\uparrow 59.5\%$)
$A^\square$	3.62	1.75 ($\downarrow 51.7\%$)

Differently, both ZDC and CZDC achieve convergent solutions because DC programming involves evaluation of elementary functions rather than inclusion functions. In Figure 3.11(b) and (c), one-dimensional intervals are sketched to illustrate that those algorithms provide guaranteed solutions. As shown in Table 3.8, CZDC provides a significantly better precision than ZDC at the cost of a larger T^{CPU} .

3.2.4.2 Attitude Estimation

Now, we show the application of CZDC to a more challenging and technological example, containing multiplicative process noise, nonlinear measurements, and state equality constraints. The considered system concerns the attitude estimation of a flying robot. By employing quaternion representation, the attitude is expressed as $x_k \in \mathbb{R}^4$ such that $\|x_k\|_2^2 = 1$. These states evolve at discrete time according to (Teixeira et al., 2009; Rego et al., 2021)

$$x_k = \left(\cos(p(\check{u}_{k-1})) I_4 - \frac{T_s}{2} \frac{\sin(p(\check{u}_{k-1}))}{p(\check{u}_{k-1})} \Omega(\check{u}_{k-1}) \right) x_{k-1}, \quad (3.130)$$

where $p(\check{u}_k) = \frac{T_s}{2} \|\check{u}_k\|_2$, $\Omega(\check{u}_k) = \begin{bmatrix} 0 & \check{u}_{3,k} & -\check{u}_{2,k} & \check{u}_{1,k} \\ -\check{u}_{3,k} & 0 & \check{u}_{1,k} & \check{u}_{2,k} \\ \check{u}_{2,k} & -\check{u}_{1,k} & 0 & \check{u}_{3,k} \\ -\check{u}_{1,k} & -\check{u}_{2,k} & -\check{u}_{3,k} & 0 \end{bmatrix}$, $\check{u}_k = \begin{bmatrix} 0.3 \sin((2\pi/12)kT_s) \\ 0.3 \sin((2\pi/12)kT_s - 6) \\ 0.3 \sin((2\pi/12)kT_s - 12) \end{bmatrix}$ is the

physical input that drives the actual system, and $T_s = 0.2$ s. For state-estimation purposes, we assume that $\check{u}_k$ is acquired by gyroscopes. Then, $\check{u}_k$ is corrupted by an additive noise $w_k \in \mathcal{W} = \{3 \times 10^{-3} \mathbf{I}_3, 0_{3 \times 1}\}$, whose result is the known signal $u_k = \check{u}_k + w_k$. The measurement is given by

$$y_k = \begin{bmatrix} C(x_k) r^{[1]} \\ C(x_k) r^{[2]} \end{bmatrix} + v_k, \quad (3.131)$$

where $r^{[1]} = [1 \ 0 \ 0]^\top$, $r^{[2]} = [0 \ 1 \ 0]^\top$,

$$C(x_k) = \begin{bmatrix} x_{1,k}^2 - x_{2,k}^2 - x_{3,k}^2 + x_{4,k}^2 & 2(x_{1,k}x_{2,k} + x_{3,k}x_{4,k}) & 2(x_{1,k}x_{3,k} - x_{2,k}x_{4,k}) \\ 2(x_{1,k}x_{2,k} - x_{3,k}x_{4,k}) & -x_{1,k}^2 + x_{2,k}^2 - x_{3,k}^2 + x_{4,k}^2 & 2(x_{1,k}x_{4,k} + x_{2,k}x_{3,k}) \\ 2(x_{1,k}x_{3,k} + x_{2,k}x_{4,k}) & 2(-x_{1,k}x_{4,k} + x_{2,k}x_{3,k}) & -x_{1,k}^2 - x_{2,k}^2 + x_{3,k}^2 + x_{4,k}^2 \end{bmatrix}$$

is a rotation matrix, and $v_k \in \mathcal{V} = \{0.15\mathbf{I}_6, 0_{6 \times 1}\}$.

To simulate the system, we consider the uncorrupted signal $\check{u}_k$, initial state $x_0 = [0 \ 1 \ 0 \ 0]^\top \in \mathcal{X}_0 = \{0.18\mathbf{I}_4, [0.1 \ 0.9 \ 0.1 \ 0.1]^\top\}$, realizations of uniform noise defined in $\mathcal{V}$ for v_k , $k_f = 200$, and $m_s = 10$. To estimate states, we consider the corrupted signal u_k , fixed values $\varphi_c = 10$ and $\varphi_g = 30$, the invariant $g(x_k) = x_k^\top x_k - 1$, and the feasible set $\mathcal{X}^F = \{\mathbf{I}_4, 0_{4 \times 1}\}$. Since $\check{u}_k$ is unknown, the algorithms replace $\check{u}_k$ by $(u_k - w_k)$. Due to the nonlinearity of $x_k = f(x_{k-1}, u_{k-1}, w_{k-1})$ in (3.130), we define the DC function $f = f^a - f^b$ such that $f_i^a = f_i + f_i^b$ and $f_i^b = \frac{\tilde{f}_i}{2} z_{k-1}^\top z_{k-1}$, with $z_{k-1} = [x_{k-1}^\top \ w_{k-1}^\top]^\top$, $i = 1, \dots, 4$, and with $\tilde{f}_i$ being given by Proposition 2.3.3 over each time step. In this case, the polytope $\mathcal{P}_{k-1}$ related to Theorem 3.2.1 is a box (given by Proposition 2.2.3) that contains the CZ $\mathcal{Z}_{k-1} = \mathcal{X}_{k-1} \times \mathcal{W}_{k-1}$. By exploiting the quadratic nature of both $y_k = h(x_k) + v_k$ in (3.131) and $g(x_k) = 0$, we propose the DC functions $y_k = h^a - h^b + v_k$ and $g = g^a - g^b$ such that

$$h^a = \begin{bmatrix} x_k^\top A_1 x_k \\ x_k^\top A_2 x_k \\ x_k^\top A_3 x_k \\ x_k^\top A_4 x_k \\ x_k^\top A_5 x_k \\ x_k^\top A_6 x_k \end{bmatrix} = \begin{bmatrix} x_{1,k}^2 + x_{4,k}^2 \\ x_{1,k}x_{2,k} - x_{3,k}x_{4,k} \\ x_{1,k}x_{3,k} + x_{2,k}x_{4,k} \\ x_{1,k}x_{2,k} + x_{3,k}x_{4,k} \\ x_{2,k}^2 + x_{4,k}^2 \\ x_{2,k}x_{3,k} - x_{1,k}x_{4,k} \end{bmatrix}, \quad h^b = \begin{bmatrix} x_k^\top B_1 x_k \\ x_k^\top B_2 x_k \\ x_k^\top B_3 x_k \\ x_k^\top B_4 x_k \\ x_k^\top B_5 x_k \\ x_k^\top B_6 x_k \end{bmatrix} = \begin{bmatrix} x_{2,k}^2 + x_{3,k}^2 \\ -x_{1,k}x_{2,k} + x_{3,k}x_{4,k} \\ -x_{1,k}x_{3,k} - x_{2,k}x_{4,k} \\ -x_{1,k}x_{2,k} - x_{3,k}x_{4,k} \\ x_{1,k}^2 + x_{3,k}^2 \\ x_{1,k}x_{4,k} - x_{2,k}x_{3,k} \end{bmatrix}, \quad g^a = g, \quad g^b = 0,$$

where

$$\begin{aligned}
 A_1 &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, A_2 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 \end{bmatrix}, A_3 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, A_4 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \\
 A_5 &= \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, A_6 = \begin{bmatrix} 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, B_1 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, B_2 = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \\
 B_3 &= \begin{bmatrix} 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, B_4 = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 \end{bmatrix}, B_5 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, B_6 = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.
 \end{aligned}$$

Note that Hessian matrices are not here candidate to replace the prior non-symmetric matrices. To achieve that, new functions h^a and h^b must be defined, but this procedure may enlarge the current eigenvalues, yielding more conservative DC approximations. Having in mind the choice of matrices here adopted, the polytopes $\mathcal{P}_k$ and $\check{\mathcal{P}}_k$, in Theorem 3.2.2 and Corollary 3.2.1, are parallelotopes (given by Proposition 2.3.2) that contain the CZs $\mathcal{Z}_k = \mathcal{X}_{k|k-1}$ and $\check{\mathcal{Z}}_k = \check{\mathcal{X}}_k$, respectively. We employ parallelotopes instead of boxes because the DC components are fixed, and thereby, $\mathcal{P}_k$ and $\check{\mathcal{P}}_k$ may reach better precision.

Figure 3.12 depicts a separate simulation with the CZDC, CZFO and CZMV algorithms. Boxes were sketched rather than CZs for computational simplicity. According to the figure, CZDC generates CZs with the smallest associated interval hulls. Moreover, a faster reduction of uncertainty is expected with CZDC during the initialization effect. Table 3.9 corroborates the improvement of precision caused by CZDC in comparison with both CZMV and CZFO. Since CZFO is, in general, more costly than CZMV (Rego et al., 2021, Table 1), it demands a larger T^{CPU} as shown in Table 3.9. Differently, CZDC can enhance the precision of CZMV using much less computational resource, and this advantage is related to both tight linearization remainder (Lemma 2.3.1) and low-dimension CZs (Remark 3.2.2). However, the quantity of operations involved with CZDC may be larger than the CZMV one, justifying the difference of T^{CPU} .

During the state estimation, CZMV and CZFO diverged for some simulations, whose results were discarded and not included in the computation of $A^\square$. In this case, the enlargement of CZs was increasing, but fixed by the admissibility step, resulting in the own feasible set $\mathcal{X}^F$. In principle, the prior results could be improved with the increase of φ_c and φ_s . However, the generator reduction can imply conservatism for some directions due to the wrapping effect. This effect is influenced by the evolution of CZs, which is

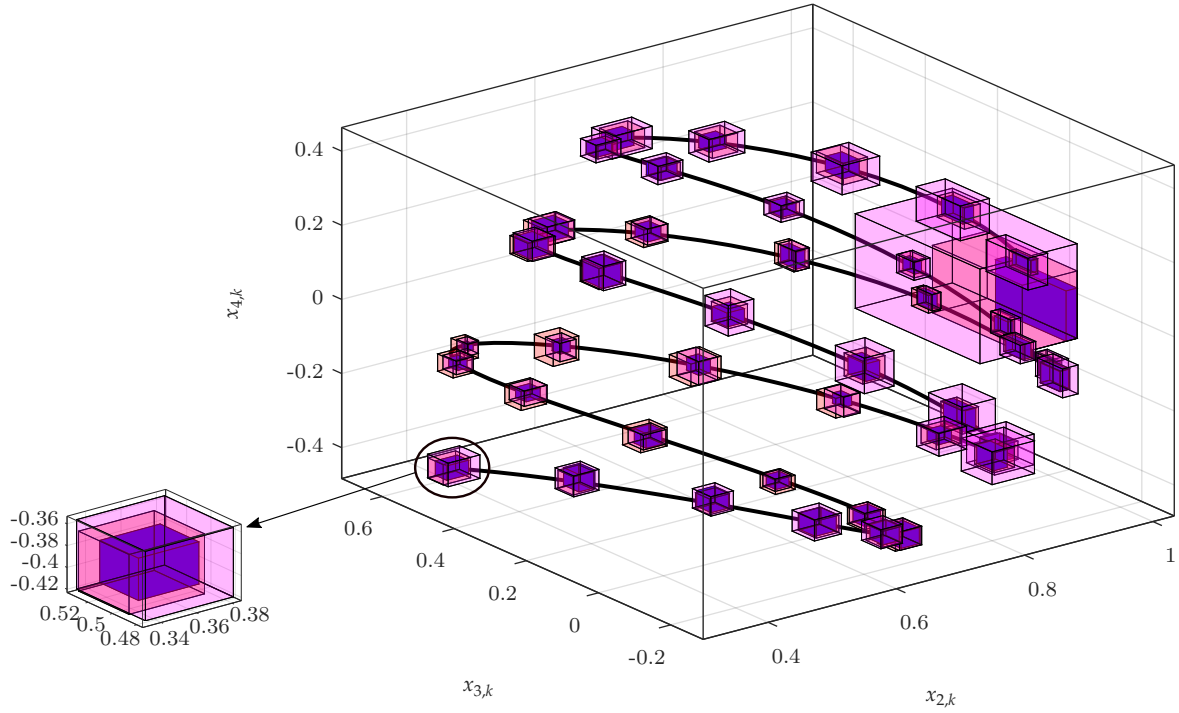


Figure 3.12: 3D projection with respect to $x_{1,k} = 0$ of the interval hulls of CZs, computed by CZDC (blue color), CZFO (red color), and CZMV (magenta color), for the second case study (Subsection 3.2.4.2). The true states are sketched in black solid line.

Table 3.9: Results of T^{CPU} and $A^\square$ for the second example (Subsection 3.2.4.2). The percentage reduction of $A^\square$ for CZDC and CZFO in comparison to CZMV is shown between parenthesis.

Indexes	CZDC	CZFO	CZMV
T^{CPU}	3.58 s	39.8 s	1.76 s
$A^\square(\times 10^{-4})$	0.0265 ($\downarrow 96.7\%$)	0.1714 ($\downarrow 78.4\%$)	0.7919

different for each Monte Carlo simulation since the noise realizations are present in u_k and y_k , affecting then the intersections. Therefore, we justified why CZMV and CZFO converged or diverged for some simulations.

Additional simulations were executed with CZDC to verify if the eigenvalues of Proposition 2.3.3 would be smaller by convexifying $-f$ instead of f , as discussed in Remark 2.3.4. Specifically, for each time step, we first computed the i th eigenvalue related to f_i and $-f_i$, with $i = 1, \dots, 7$. After, we compared which eigenvalues were smaller, deciding then the i th DC function to be used. However, the tests pointed out that convexifying f yielded the best solutions.

3.3 Final Remarks

In this chapter, extensions of state estimators based on either GRVs or sets were presented. The main motivation was to improve some performance measures as precision,

accuracy, and computational burden of existing algorithms in the literature. For instance, in comparison with the KP (Subsection A.1.1), the proposed DGF introduces analysis with state equality constraints, degrees of freedom (extra design matrices), and filter version, whose resulting algorithm improves both precision and accuracy of the KP at the cost of a larger processing time. Likewise, the proposed ZKF improves both the precision and accuracy of its predictor version (ZKP in Subsection B.1.1). The proposed DRZF introduces a new cost criterion under the same observer structure of DTZF (Subsection B.1.2), in which the interval radius of zonotopes is minimized instead of the F -radius. This procedure may enhance the precision of DTZF because the radius minimization affects the volume minimization, whereas there is no direct connection between trace and volume minimizations. Moreover, we showed that the minimum-radius optimization problem is convex and derived an LP, enabling the use of efficient solvers as CPLEX and Gurobi. For nonlinear systems, we proposed two set-membership observers: IZMV and CZDC. The IZMV extends the ZMV (Subsection B.2.2) by enforcing state inequality constraints and reducing the computational effort of VM1 (Subsection B.2.2.1). Specifically, IZMV can incorporate zonotopic constraints by means of intersection among zonotopes and strips, while VM1 is rewritten as an LP in VM3. In turn, the proposed CZDC extends the CZLF (Subsection B.1.4) to work on constrained nonlinear systems. In this case, the DC programming approach proposed in [Alamo et al. \(2008\)](#) is employed to enclose the analytical linearization remainder. Additionally, we showed how compact feasible sets and nonlinear invariants can be enforced by CZDC to tighten the sets obtained after the data-assimilation step. This procedure is relevant to mitigate the wrapping effect introduced by order-reduction algorithms over the time steps.

4

Novel Mixed-Uncertainty Approaches for Linear Systems

This chapter proposes new state estimators for linear systems with mixed uncertainties. One of our motivations is to extend separate approaches, proposed in Chapter 3 and/or revisited in Appendices A and B, to work on the mixed-uncertainty case. For instance, the proposed ZGKF, based on zonotopes and GRVs, is the direct extension of the ZGKP (Subsection C.1) to filter. The improvements of precision and accuracy with filters have been widely discussed in the literature (Jazwinski, 2007; Teixeira, 2008; Särkkä, 2013). The proposed LTVMF combines both DGF (Algorithm 3.1) and DTZF (Subsection B.1.2) to generate a novel mixed-uncertainty filter based on CZs and GRVs for LTV descriptor systems. In this algorithm, we introduce a Pareto analysis to justify the choice of solutions and propose two versions to enforce state constraints. In the first version, quadratic programs are solved including linear equality and inequality constraints. In the second version, we separate the moment in which the constraints are enforced to explicitly solve minimum-trace problems, and thereby, to mitigate the use of solvers. Finally, the proposed LTIMF combines both DH₂GF (Algorithm 3.2) and H_∞ZF (Subsection B.1.3) to yield a novel mixed-uncertainty filter based on CZs and GRVs for LTI descriptor systems. This algorithm is developed under a reachable set analysis, with the main motivation being the computational efficiency to compute interval hull of CZs; this approach disregards the use of order-reduction algorithms. In Table 4.1, the main attributes of the new algorithms are pointed out.

Table 4.1: Overview of the linear mixed-uncertainty algorithms here proposed. Each set of lines contains information about the extended estimator. The numeration of each algorithm is presented between brackets in the first column, with our proposals being differed from the existing approaches (just above) through boldface. The remaining columns characterize the algorithms regarding: uncertainty — Gaussian random vector (GRV), zonotope (Zon), constrained zonotope (CZ) —, system — state space (SS), descriptor (D), linear time invariant (LTI), linear time varying (LTV) —, criterion — minimum trace (MT), H_∞ , H_2/H_∞ —, category — predictor (P), filter (F) —, constraints — equality constraints (EC), inequality constraints (IC) —, and Pareto analysis ($\star$).

Algorithm	Uncertainty	System	Criterion	Cat.	Constraints	Pareto
ZGKP [C.1]	Zon, GRV	SS, LTV	MT	P		
ZGKF [4.1]	Zon, GRV	SS, LTV	MT	F		
DTZF [B.3]	Zon	D, LTV	MT	F	EC	
LTVMF [4.2]	CZ, GRV	D, LTV	MT	F	EC, IC	$\star$
H_∞ ZF [B.4]	Zon	SS, LTI	H_∞	F		
LTMF [4.4]	CZ, GRV	D , LTI	H_2/H_∞	F	EC, IC	$\star$

4.1 Zonotopic and Gaussian Kalman Filter

Here, we propose the zonotopic and Gaussian Kalman filter for LTV systems (ZGKF). This algorithm is the filter version of the ZGKP (see Subsection C.1) since it merges the KF (see Subsection A.1.2) with the ZKF (Algorithm 3.4) into a unique state estimator. Unlike predictors, filters assimilate current measurements, and then, both their precision and accuracy may be better than the predictors ones (Teixeira, 2008). Therefore, ZGKF is motivated to improve the performance of ZGKP. This result was published in de Paula et al. (2019).

Consider the discrete-time LTV uncertain dynamical system

$$x_k = A_{k-1}x_{k-1} + B_{k-1}u_{k-1} + z_{k-1}^w + g_{k-1}^w, \quad (4.1)$$

$$y_k = C_k x_k + z_k^v + g_k^v, \quad (4.2)$$

where $A_{k-1} \in \mathbb{R}^{n \times n}$, $B_{k-1} \in \mathbb{R}^{n \times p}$, and $C_k \in \mathbb{R}^{m \times n}$ are the known system matrices, $u_{k-1} \in \mathbb{R}^p$ is the deterministic input vector, $y_k \in \mathbb{R}^m$ is the measured output vector, and $x_k \in \mathbb{R}^n$ is the state vector to be estimated. Unlike Chapter 3, the process $w_{k-1} \in \mathbb{R}^n$ and measurement $v_k \in \mathbb{R}^m$ noises, as well as the initial state $x_0 = c_0^x + z_0^x + g_0^x$, are mixed as in (2.45). The bounded terms w_{k-1}^z , v_k^z , and z_0^x are characterized by the zonotopes $\mathcal{W}_{k-1} = \{G_{k-1}^w, 0_{n \times 1}\}$, $\mathcal{V}_k = \{G_k^v, 0_{m \times 1}\}$, and $\mathcal{X}_0 = \{G_0^x, 0_{n \times 1}\}$, respectively, such that $G_{k-1}^w \in \mathbb{R}^{n \times n_g^w}$, $G_k^v \in \mathbb{R}^{m \times n_g^v}$, $G_0^x \in \mathbb{R}^{n \times n_g^x}$, and $G_k^v (G_k^v)^\top > 0_{m \times m}$. The stochastic terms w_{k-1}^g , v_k^g , and g_0^x are characterized by the GRVs $W_{k-1} \sim \mathcal{N}(0_{n \times 1}, Q_{k-1})$, $V_k \sim \mathcal{N}(0_{m \times 1}, R_k)$, and $X_0 \sim \mathcal{N}(0_{n \times 1}, P_0^{xx})$, such that $Q_{k-1} \geq 0_{n \times n}$, $R_k > 0_{m \times m}$, and $P_0^{xx} > 0_{n \times n}$. In this case, P_0^{xx} must be positive definite due to the confidence ellipsoid (2.46). We assume that the GRVs W_{k-1} , V_k , and X_0 are uncorrelated, and that $E[W_k W_j^\top] = 0_{n \times n}$ and

$E[V_k V_j^T] = 0_{m \times m}$ for all $k \neq j$. Since the exact center c_0^x , zonotope $\mathcal{X}_0$, and GRV X_0 are unknown, they are approximated by the corresponding known estimates $\hat{c}_0^x$, $\hat{\mathcal{X}}_0 = \{\hat{G}_0^x, 0_{n \times 1}\}$, and $\hat{X}_0 \sim \mathcal{N}(0_{n \times 1}, \hat{P}_0^{xx})$.

The goal is to determine the state estimates $\hat{c}_k^x$, $\hat{\mathcal{X}}_k$, and $\hat{X}_k$ over $k \in \mathbb{Z}_+$ using an estimator structure that is similar to KF and ZKF. Therefore, we propose the following center estimate for the mixed-uncertainty case:

$$\hat{c}_k^x = (I_n - K_k C_k) \hat{c}_{k|k-1}^x + K_k y_k, \quad (4.3)$$

where

$$\hat{c}_{k|k-1}^x = A_{k-1} \hat{c}_{k-1}^x + B_{k-1} u_{k-1}, \quad (4.4)$$

and $K_k \in \mathbb{R}^{n \times m}$ is the gain matrix to be determined. By defining the estimation error $e_k \triangleq x_k - \hat{c}_k^x$ and using (4.2), we obtain

$$e_k = (I_n - K_k C_k) (A_{k-1} e_{k-1} + w_{k-1}^z + w_{k-1}^g) - K_k (v_k^z + v_k^g).$$

Since $x_k \triangleq c_k^x + z_k^x + g_k^x$, we also have

$$e_k = (c_k^x - \hat{c}_k^x) + z_k^x + g_k^x.$$

Assuming that the center deviation $(c_k^x - \hat{c}_k^x)$ is shifted to both terms z_k^x and g_k^x , yielding new variables $\hat{z}_k^x$ and $\hat{g}_k^x$, respectively, we reach the error terms

$$\hat{z}_k^x = (I_n - K_k C_k) (A_{k-1} \hat{z}_{k-1}^x + w_{k-1}^z) - K_k v_k^z, \quad (4.5)$$

$$\hat{g}_k^x = (I_n - K_k C_k) (A_{k-1} \hat{g}_{k-1}^x + w_{k-1}^g) - K_k v_k^g, \quad (4.6)$$

which are characterized next in Proposition 4.1.1. This change of variables is required because the center deviation is unknown, and thereby, it should be treated as an additional uncertainty in z_k^x and g_k^x .

Proposition 4.1.1. Consider the state-estimation errors $\hat{z}_k^x$ in (4.5) and $\hat{g}_k^x$ in (4.6) related to the dynamical system (4.1)-(4.2). From Properties 2.1.1, 2.1.2, 2.2.2, and 2.2.3, the error terms $\hat{z}_k^x$ and $\hat{g}_k^x$ are characterized by

$$\hat{\mathcal{X}}_{k|k-1} = A_{k-1} \hat{\mathcal{X}}_{k-1} \oplus \mathcal{W}_{k-1} = \{\hat{G}_{k|k-1}^x, 0_{n \times 1}\}, \quad (4.7)$$

$$\hat{\mathcal{X}}_k = (I_n - K_k C_k) \hat{\mathcal{X}}_{k|k-1} \oplus (-K_k \mathcal{V}_k) = \{\hat{G}_k^x, 0_{n \times 1}\}, \quad (4.8)$$

$$\hat{X}_{k|k-1} = A_{k-1} \hat{X}_{k-1} + W_{k-1} \sim \mathcal{N}(0_{n \times 1}, \hat{P}_{k|k-1}^{xx}), \quad (4.9)$$

$$\hat{X}_k = (I_n - K_k C_k) \hat{X}_{k|k-1} - K_k V_k \sim \mathcal{N}(0_{n \times 1}, \hat{P}_k^{xx}), \quad (4.10)$$

where

$$\hat{G}_{k|k-1}^x = \begin{bmatrix} A_{k-1} \hat{G}_{k-1}^x & G_{k-1}^w \end{bmatrix}, \quad (4.11)$$

$$\hat{G}_k^x = \begin{bmatrix} (I_n - K_k C_k) \hat{G}_{k|k-1}^x & -K_k G_k^v \end{bmatrix}, \quad (4.12)$$

$$\hat{P}_{k|k-1}^{xx} = A_{k-1} \hat{P}_{k-1}^{xx} A_{k-1}^\top + Q_{k-1}, \quad (4.13)$$

$$\hat{P}_k^{xx} = (I_n - K_k C_k) \hat{P}_{k|k-1}^{xx} (I_n - K_k C_k)^\top + K_k R_k K_k^\top. \quad (4.14)$$

Proof. Since the elements of (4.5) and the realizations of (4.6) are replaced by their uncertainty representation, the sets (4.7) and (4.8), and the random vectors (4.9) and (4.10) are obtained. Having in mind that the random vectors were assumed to be uncorrelated, the application of Properties 2.1.1, 2.1.2, 2.2.2, and 2.2.3 to the prior representations imply zonotopic sets and GRVs with characterization given by (4.11)-(4.14). ■

Here, the objective function to be minimized is formulated as in (1.9), where $J_k^{zg}(K_k)$ is the convex linear combination of $\text{tr}(\hat{G}_k^x (\hat{G}_k^x)^\top)$ with $\text{tr}(\hat{P}_k^{xx})$ through of a known parameter $\eta \in [0, 1]$. The optimal computation of K is presented next in Theorem 4.1.1. Thereby, the state estimates $\hat{c}_k^x$, $\hat{X}_k$, and $\hat{X}_k$ are optimally generated.

Theorem 4.1.1. Let $\hat{G}_k^x$ in (4.12) and $\hat{P}_k^{xx}$ in (4.14) be the instances of the cost function J_k^{zg} in (1.9), as well as the known weight parameter $\eta \in [0, 1]$. Then, minimizing J_k^{zg} results in the optimal gain $\check{K}_k$ given by

$$\check{K}_k = \hat{S}_{k|k-1}^{xy} \left(\hat{S}_{k|k-1}^{yy} \right)^{-1}, \quad (4.15)$$

where

$$\hat{S}_{k|k-1}^{xy} = \hat{P}_{k|k-1}^{zg} C_k^\top, \quad (4.16)$$

$$\hat{S}_{k|k-1}^{yy} = C_k \hat{P}_{k|k-1}^{zg} C_k^\top + R_k^{zg}, \quad (4.17)$$

$$\hat{P}_{k|k-1}^{zg} = (1 - \eta) \hat{G}_{k|k-1}^x (\hat{G}_{k|k-1}^x)^\top + \eta \hat{P}_{k|k-1}^{xx}, \quad (4.18)$$

$$R_k^{zg} = (1 - \eta) G_k^v (G_k^v)^\top + \eta R_k. \quad (4.19)$$

Proof. According to Proposition 2.5.1, the function J_k^{zg} in (1.9) can be rewritten as

$$J_k^{zg} = \text{tr}(\hat{P}_{k|k-1}^{zg}) - 2\text{tr}(\hat{S}_{k|k-1}^{xy} K_k^\top) + \text{tr}(K_k \hat{S}_{k|k-1}^{yy} K_k^\top),$$

where $\hat{S}_{k|k-1}^{xy}$, $\hat{S}_{k|k-1}^{yy}$, and $\hat{P}_{k|k-1}^{zg}$ are given by (4.16), (4.17), and (4.18), respectively.

By employing Proposition 2.5.1 once again, and by exploiting the symmetry of both $\hat{P}_{k|k-1}^{zg}$ and $\hat{S}_{k|k-1}^{yy}$, the procedure $\frac{\partial J_k^{zg}(K_k)}{\partial K_k} = 0$ yields

$$\hat{S}_{k|k-1}^{xy} = K_k \hat{S}_{k|k-1}^{yy}.$$

Finally, the fact of $R_k^{zg} > 0_{m \times m}$ is enough to guarantee that $\hat{S}_{k|k-1}^{yy} > 0_{m \times m}$, whose inversion

yields the optimal gain $\check{K}_k$ in (4.15). ■

Recall that the generator matrix $\hat{G}_k^x$ in (4.12) is obtained by means of Minkowski sum. Therefore, in comparison with $\hat{G}_{k-1}^x$, its order increases from n_g to $(n_g + n_g^w + n_g^v)$ at the end of one iteration. To mitigate the increasing computational cost, ZR is here applied to $\hat{X}_{k-1}$ to reduce its order from n_g to φ_g . Besides the current state estimates, we also compute the confidence box $\mathcal{I}_k^a$ given by (2.49). This set represents the fusion of the mixed state estimates into the Euclidean space.

Next, the ZGKF is described in Algorithm 4.1 and illustrated in Figure 4.1, from which Problem 1.2.4 is solved without enforcing state constraints. Its complexity order is given by

$$O(\text{ZGKF}) = O(n_g(n + \log(n_g))) + n(p + mn_g^v) + n^2(n + m + \varphi_g + n_g^w) + m^2(m + n + n_g^v).$$

By comparing the complexity order of both ZGKF and ZGKP, we have:

$$O(\text{ZGKF}) - O(\text{ZGKP}) = nn_g^w(n - 1).$$

Then, ZGKP may be computationally cheaper than ZGKF.

Algorithm 4.1: $[\hat{c}_k^x, \hat{X}_k, \hat{X}_k, \mathcal{I}_k^a] = \text{ZGKF}(\hat{c}_{k-1}^x, \hat{X}_{k-1}, \hat{X}_{k-1}, A_{k-1}, B_{k-1}, C_k, \mathcal{W}_{k-1}, \mathcal{V}_k, W_{k-1}, V_k, u_{k-1}, y_k, \varphi_g, \eta, \alpha)$

- 1 Compute $\hat{X}_{k-1}^\downarrow = \text{ZR}(\hat{X}_{k-1}, \varphi_g)$, where ZR is discussed in Subsection B.1.1.1
 - 2 Determine the forecast estimates $\hat{c}_{k|k-1}^x$ in (4.4), $\hat{G}_{k|k-1}^x$ in (4.11), and $\hat{P}_{k|k-1}^{xx}$ in (4.13)
 - 3 Apply Theorem 4.1.1 to obtain the optimal gain $\check{K}_k$
 - 4 Compute the *a posteriori* estimates $\hat{c}_k^x$ in (4.3), $\hat{G}_k^x$ in (4.12), and $\hat{P}_k^{xx}$ in (4.14), yielding $\hat{X}_k$ in (4.8) and $\hat{X}_k$ in (4.10), and thereby, the confidence box $\mathcal{I}_k^a$ given by (2.49)
-

Remark 4.1.1. Besides confidence box, Algorithm 4.1 could return a confidence zonotope as in (2.47).

4.1.1 Numerical Example

Next, we present an example modified from Alamo et al. (2005) to illustrate the advantages of ZGKF over ZGKP. Consider the discrete-time linear system

$$x_k = \begin{bmatrix} 0 & -0.5 \\ 1 & 1 \end{bmatrix} x_{k-1} + 0.02 \begin{bmatrix} -6 \\ 1 \end{bmatrix} w_{k-1}, \quad (4.20)$$

$$y_k = [-2 \quad 1] x_k + 0.2 v_k, \quad (4.21)$$

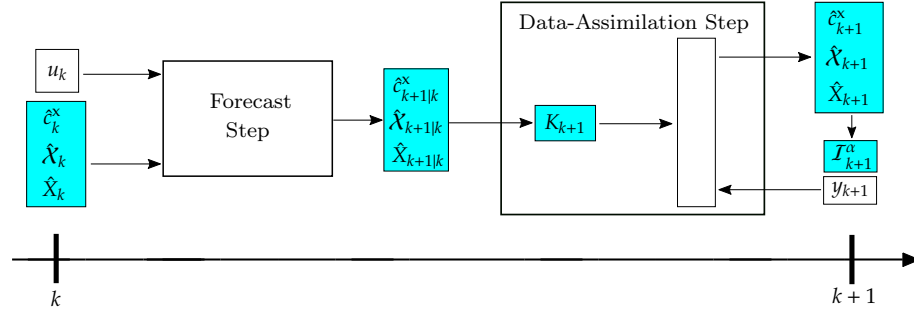


Figure 4.1: Illustrative diagram of an iteration of the zonotopic and Gaussian Kalman filter (ZGKF) given by Algorithm 4.1. The cyan color is used to highlight the main estimates manipulated/obtained by the ZGKF throughout two base steps. The forecast step consists of propagating the mixed state estimates at k over the process model (4.1). The data-assimilation step employs the forecast mixed estimates to compute the Kalman-like gain K_{k+1} given the measurement model (4.2), and uses the current measurements y_{k+1} to determine the mixed state estimates at $k+1$, and thereby, the confidence box in (2.49).

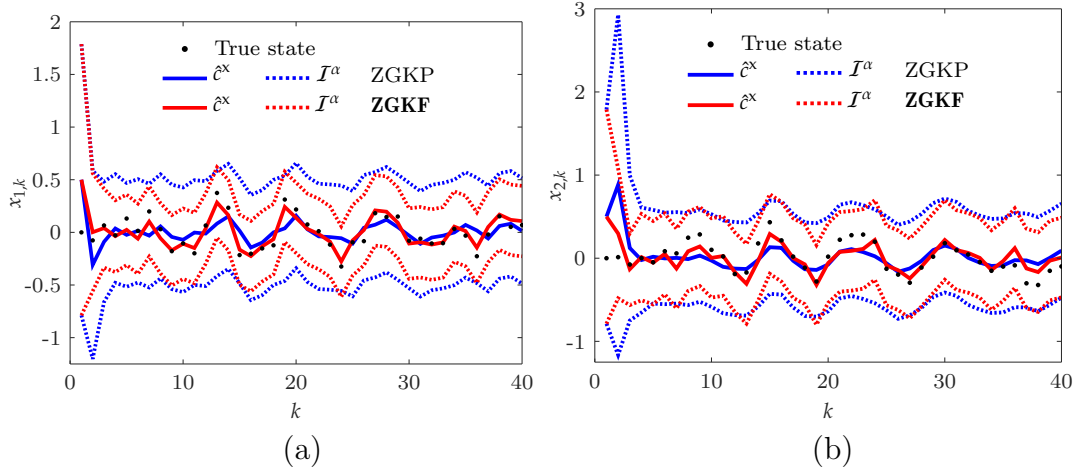


Figure 4.2: State estimation for a two-state linear system. Graphs (a) and (b) illustrate the confidence intervals ($\cdots$) and the centers ($—$) generated by the ZGKF (blue color) and ZGKF (red color) algorithms to represent the true states ($\cdot$).

where the realizations w_{k-1} are taken from the GRV $W_{k-1} \sim \mathcal{N}(0, 1)$, while the realizations v_k are taken from a uniform variable defined in the zonotope $\mathcal{V}_k = \{1, 0\}$. The initial state is set as $x_0 = 0_{2 \times 1}$.

To estimate the states, we set the following parameters: $\hat{c}_0^x = [0.5 \ 0.5]^\top$, $\hat{G}_0 = 0.60I_2$, $\hat{P}_0^{xx} = 0.040I_2$, $\varphi_g = 14$, $\eta = 0.5$, and $\alpha = 0.0027$, such that $x_0 \in \mathcal{I}_0^\alpha$ with 99.73% confidence level. To compare the algorithms, we employ the indexes T^{CPU} in (3.57), r^α in (3.84), and RMSE in (3.58) with $m_s = 100$, $k_0 = 4$, and $k_f = 40$.

Figures 4.2a and 4.2b illustrate the state estimates computed by ZGKF and ZGKF. As expected, ZGKF provides: (i) center estimates closer to the true states; and (ii) confidence boxes smaller than the ZGKF. Beyond corroborating that ZGKF produces better precision and accuracy, Table 4.2 shows that the T^{CPU} of ZGKF is better than or equal to the ZGKF one.

Table 4.2: Numerical results from the state estimation of the system (4.20)-(4.21) using ZGKP and ZGKF. The reduction/growth of value with respect to the ZGKP is shown between parenthesis.

Estimators	$T^{\text{CPU}} \times 10^{-4}(\text{s})$	$r^{\square}$	RMSE ₁	RMSE ₂
ZGKP	4.81	0.627	0.129	0.141
ZGKF	4.90 ($\uparrow 1.9\%$)	0.473 ($\downarrow 25\%$)	0.0627 ($\downarrow 51\%$)	0.100 ($\downarrow 29\%$)

4.2 Mixed-Uncertainty Filter for Descriptor LTV Systems

In this section, we propose the mixed-uncertainty filter for descriptor LTV systems (LTVMF). This algorithm brings up different contributions in comparison with the ZGKF. First, the LTVMF combines the DGF (Algorithm 3.1) with the DTZF (Subsection B.1.2). Second, zonotopes are replaced by CZs to allow for propagation of asymmetric polytopes. Third, state linear equality and inequality constraints are introduced to promote tight solutions. Fourth, quadratic and explicit versions are proposed to execute the LTVMF, with the motivation being a tradeoff between precision and computational burden. Finally, a Pareto analysis is provided to select the optimal solutions over time. The LTVMF is presented in de Paula et al. (2022b).

4.2.1 General Filter Design

4.2.1.1 Original Problem

Consider the discrete-time descriptor LTV system

$$E_k x_k = A_{k-1} x_{k-1} + B_{k-1} u_{k-1} + w_{k-1}, \quad (4.22)$$

$$y_k = C_k x_k + v_k, \quad (4.23)$$

where $E_k \in \mathbb{R}^{n \times n}$, $A_{k-1} \in \mathbb{R}^{n \times n}$, $B_{k-1} \in \mathbb{R}^{n \times p}$, and $C_k \in \mathbb{R}^{m \times n}$ are known matrices, $u_{k-1} \in \mathbb{R}^p$ is the deterministic input vector, $y_k \in \mathbb{R}^m$ is the measured output vector, and $x_k \in \mathbb{R}^n$ is the state vector to be estimated over $k \in \mathbb{Z}_+$. We assume that $\text{rank} \left(\begin{bmatrix} E_k \\ C_k \end{bmatrix} \right) = n$. Additionally, we assume that the states x_k satisfy the state linear inequality constraints in (1.4).

The terms $w_{k-1} = z_{k-1}^w + g_{k-1}^w \in \mathbb{R}^n$ and $v_k = z_k^v + g_k^v \in \mathbb{R}^m$ correspond to the process and measurement noises, respectively. The set-bounded parcels z_{k-1}^w and z_k^v are characterized by the zero-center CZs $\mathcal{W}_{k-1} = \{G_{k-1}^w, 0_{n \times 1}, A_{k-1}^w, b_{k-1}^w\}$ and $\mathcal{V}_k = \{G_k^v, 0_{m \times 1}, A_k^v, b_k^v\}$, respectively, whose generator matrices $G_{k-1}^w \in \mathbb{R}^{n \times n_g^w}$ and $G_k^v \in \mathbb{R}^{m \times n_g^v}$ have full rank. The stochastic parcels g_{k-1}^w and g_k^v are realizations of independent stochastic processes, and characterized by the zero-mean GRVs $W_{k-1} \sim \mathcal{N}(0_{n \times 1}, Q_{k-1})$ and $V_k \sim \mathcal{N}(0_{m \times 1}, R_k)$, respectively, with positive-definite covariance matrices $Q_{k-1} > 0_{n \times n}$ and $R_k > 0_{m \times m}$.

In turn, the initial state vector $x_0 = c_0^x + z_0^x + g_0^x$ is considered *mixed*, where c_0^x is its

center, z_0^x is the set-bounded term represented by the zero-center CZ $X_0 = \{G_0^x, 0_{n \times 1}, A_0^x, b_0^x\}$, and g_0^x is the stochastic term represented by the zero-mean GRV $X_0 \sim \mathcal{N}(0_{n \times 1}, P_0^{xx})$ such that $P_0^{xx} \succ 0_{n \times n}$. Moreover, we assume that g_0^x is uncorrelated to g_{k-1}^w and g_k^v . Therefore, the state vector x_k is given by the combination of c_k^x , $X_k = \{G_k^x, 0_{n \times 1}, A_k^x, b_k^x\}$, and $X_k \sim \mathcal{N}(0_{n \times 1}, P_k^{xx})$.

As $\text{rank}\left(\begin{bmatrix} E_k \\ C_k \end{bmatrix}\right) = n$, there exists a full-rank matrix $\begin{bmatrix} T_k & N_k \end{bmatrix}$ that satisfies (3.3), where $T_k \in \mathbb{R}^{n \times n}$ and $N_k \in \mathbb{R}^{n \times m}$ are design matrices. Post-multiplying (3.3) by x_k and using (4.22)-(4.23), we obtain

$$x_k = T_k A_{k-1} x_{k-1} + T_k B_{k-1} u_{k-1} + N_k y_k + T_k z_{k-1}^w + T_k g_{k-1}^w - N_k z_k^v - N_k g_k^v. \quad (4.24)$$

Besides adding degrees of freedom by means of T_k and N_k , the equality (3.3) is useful to obtain a non-descriptor representation for (4.22)-(4.23). According to (4.23) and (4.24), we formulate the following center estimate:

$$\hat{c}_k^x = T_k A_{k-1} \hat{c}_{k-1}^x + T_k B_{k-1} u_{k-1} + K_{k-1} (y_{k-1} - C_{k-1} \hat{c}_{k-1}^x) + N_k y_k. \quad (4.25)$$

Next, we present two lemmas. The first addresses the state estimation error, while the second characterizes the mixed state vector x_k .

Lemma 4.2.1. Let $\hat{c}_k^x$ given by (4.25) be the state estimate for the system (4.22)-(4.23). Given Definition 2.4.1 and the estimation error $e_k \triangleq x_k - \hat{c}_k^x$, we obtain the following constrained zonotopic $\hat{z}_k^x$ and Gaussian $\hat{g}_k^x$ terms:

$$\hat{z}_k^x = (T_k A_{k-1} - K_{k-1} C_{k-1}) \hat{z}_{k-1}^x + T_k z_{k-1}^w - K_{k-1} z_{k-1}^v - N_k z_k^v, \quad (4.26)$$

$$\hat{g}_k^x = (T_k A_{k-1} - K_{k-1} C_{k-1}) \hat{g}_{k-1}^x + T_k g_{k-1}^w - K_{k-1} g_{k-1}^v - N_k g_k^v. \quad (4.27)$$

Proof. According to $\hat{c}_k^x$ in (4.25), the estimation error is given by

$$\begin{aligned} e_k &= x_k - \hat{c}_k^x \\ &= T_k A_{k-1} (x_{k-1} - \hat{c}_{k-1}^x) + T_k z_{k-1}^w + T_k g_{k-1}^w - N_k z_k^v - N_k g_k^v - K_{k-1} y_{k-1} + K_{k-1} C_{k-1} \hat{c}_{k-1}^x, \end{aligned}$$

where x_k is given by (4.24). After making explicit $y_{k-1} = C_{k-1} x_{k-1} + z_{k-1}^v + g_{k-1}^v$, we obtain

$$e_k = (T_k A_{k-1} - K_{k-1} C_{k-1}) e_{k-1} + T_k z_{k-1}^w + T_k g_{k-1}^w - K_{k-1} z_{k-1}^v - K_{k-1} g_{k-1}^v - N_k z_k^v - N_k g_k^v.$$

From Definition 2.4.1,

$$\begin{aligned} e_k &= x_k - \hat{c}_k^x \\ &= c_k^x + z_k^x + g_k^x - \hat{c}_k^x. \end{aligned}$$

By assuming that the center deviation $c_k^x - \hat{c}_k^x$ is shifted to both z_k^x and g_k^x , we obtain new $\hat{z}_k^x$ and $\hat{g}_k^x$ (respectively) such that $e_k = \hat{z}_k^x + \hat{g}_k^x$. Thereby, we summarize the constrained zonotopic terms $\hat{z}_k^x$ in (4.26), and the Gaussian terms $\hat{g}_k^x$ in (4.27), finalizing the proof. ■

Lemma 4.2.2. Let the elements $\hat{z}_k^x$ and realizations $\hat{g}_k^x$ be expressed as in Lemma 4.2.1. Then, $\hat{z}_k^x$ and $\hat{g}_k^x$ are represented, respectively, by the CZ

$$\begin{aligned}\hat{\mathcal{X}}_k &= (T_k A_{k-1} - K_{k-1} C_{k-1}) \hat{\mathcal{X}}_{k-1} \oplus T_k \mathcal{W}_{k-1} \oplus (-K_{k-1}) \mathcal{V}_{k-1} \oplus (-N_k) \mathcal{V}_k \\ &= \left\{ \hat{G}_k^x, 0_{n \times 1}, \hat{A}_k^x, \hat{b}_k^x \right\}\end{aligned}\quad (4.28)$$

and by the GRV

$$\hat{X}_k = (T_k A_{k-1} - K_{k-1} C_{k-1}) \hat{X}_{k-1} + T_k W_{k-1} - K_{k-1} V_{k-1} - N_k V_k \sim \mathcal{N}(0_{n \times 1}, \hat{P}_k^{xx}), \quad (4.29)$$

with

$$\hat{G}_k^x = \begin{bmatrix} (T_k A_{k-1} - K_{k-1} C_{k-1}) \hat{G}_{k-1}^x & T_k G_{k-1}^w & -K_{k-1} G_{k-1}^v & -N_k G_k^v \end{bmatrix}, \quad (4.30)$$

$$\hat{A}_k^x = \begin{bmatrix} \hat{A}_{k-1}^x & 0_{n_h \times n_g^w} & 0_{n_h \times n_g^v} & 0_{n_h \times n_g^v} \\ 0_{n_h^w \times n_g} & A_{k-1}^w & 0_{n_h^w \times n_g^v} & 0_{n_h^w \times n_g^v} \\ 0_{n_h^v \times n_g} & 0_{n_h^v \times n_g^w} & A_{k-1}^v & 0_{n_h^v \times n_g^v} \\ 0_{n_h^v \times n_g} & 0_{n_h^v \times n_g^w} & 0_{n_h^v \times n_g^v} & A_k^v \end{bmatrix}, \quad \hat{b}_k^x = \begin{bmatrix} \hat{b}_{k-1}^x \\ b_{k-1}^w \\ b_{k-1}^v \\ b_k^v \end{bmatrix}, \quad (4.31)$$

$$\hat{P}_k^{xx} = (T_k A_{k-1} - K_{k-1} C_{k-1}) \hat{P}_{k-1}^{xx} (T_k A_{k-1} - K_{k-1} C_{k-1})^\top + T_k Q_{k-1} T_k^\top + K_{k-1} R_{k-1} K_{k-1}^\top + N_k R_k N_k^\top. \quad (4.32)$$

Proof. Replacing (4.26) and (4.27) by their uncertainty representation yields the descriptions $\hat{\mathcal{X}}_k$ in (4.28) and $\hat{X}_k$ in (4.29). According to sum and affine transformation operations of CZs and GRVs, the CZ $\hat{\mathcal{X}}_k$ and the GRV $\hat{X}_k$ are specified by (4.30)-(4.32). ■

Remark 4.2.1. In Lemma 4.2.2, zero center and zero mean were defined to $\hat{\mathcal{X}}_k$ and $\hat{X}_k$, respectively. In this case, we assume that the center deviation $c_k^x - \hat{c}_k^x$ is treated as an additional uncertainty such that $\hat{\mathcal{X}}_k \supset \mathcal{X}_k$ and $\hat{P}_k^{xx} \succ P_k^{xx}$, where $\mathcal{X}_k$ and P_k^{xx} correspond to the exact CZ and covariance matrix, respectively.

In order to determine design matrices T_k , N_k , and K_{k-1} , we here pursue the objective function J_k^{zg} defined in (1.9). By considering the trace criterion, a new challenge arises about approximating CZs by zonotopes. This approximation is required to keep algebraic connections between covariance and covariation matrices, and thereby, to formulate quadratic programs. According to Scott et al. (2016), we identify three possible strategies: (i) either to disregard the equality constraints given by $\hat{A}_k^x$ and $\hat{b}_k^x$, (ii) or to apply CZR to completely remove equality constraints, or (iii) to replace $\mathcal{B}(\hat{A}_k^x, \hat{b}_k^x)$ by its interval hull. In order to verify implications of each choice, we present the following illustrative example.

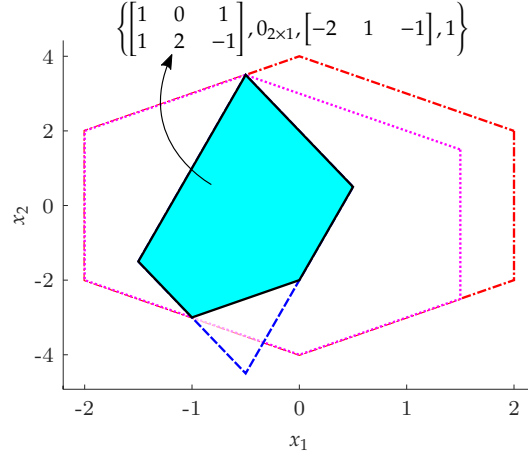


Figure 4.3: Three different zonotopes are obtained from a CZ (—) by: (i) disregarding the equality constraints on the unitary box (— · —), (ii) applying CZR to completely remove equality constraints (— —), and (iii) approximating the constrained unitary box by its interval hull (···).

Example 4.2.1. Consider the CZ shown in Figure 4.3. This set is overapproximated by zonotopes through three methods, from which: (i) returns the most conservative set; (ii) returns the most precise set; and (iii) returns the intermediate precision set.

According to Example 4.2.1, we realize that (ii) would be the most appealing strategy to approximate the CZ $\hat{\mathcal{X}}_k$, given by (4.28), by a zonotope $\tilde{\mathcal{X}}_k$, since more precise sets could be obtained. However, (ii) would require the knowledge of the design matrices T_k , N_k , and K_{k-1} (so far unknown because the zonotopic approximation is here a requirement to address optimal solutions to such matrices) to execute the order reduction, then it is disregarded. Among the remaining options, (iii) is indeed the most recommended methodology, because affine operations do not alter the original equality constraints of CZs (known).

Then, the following proposition is presented to make minimum-trace formulations involving CZs.

Proposition 4.2.1. Let $[\zeta^L, \zeta^U] \triangleq \square \mathcal{B}(\hat{A}_k^x, \hat{b}_k^x)$ be the interval hull of the constrained unitary box $\mathcal{B}(\hat{A}_k^x, \hat{b}_k^x)$ of the CZ $\hat{\mathcal{X}}_k$ given by (4.28), where

$$[\zeta^L, \zeta^U] = \begin{bmatrix} [\zeta_{k-1}^{Lx}, \zeta_{k-1}^{Ux}] \\ [\zeta_{k-1}^{Lw}, \zeta_{k-1}^{Uw}] \\ [\zeta_{k-1}^{Lv}, \zeta_{k-1}^{Uv}] \\ [\zeta_k^{Lv}, \zeta_k^{Uv}] \end{bmatrix} = \begin{bmatrix} \square \mathcal{B}(\hat{A}_{k-1}^x, \hat{b}_{k-1}^x) \\ \square \mathcal{B}(A_{k-1}^w, b_{k-1}^w) \\ \square \mathcal{B}(A_{k-1}^v, b_{k-1}^v) \\ \square \mathcal{B}(A_k^v, b_k^v) \end{bmatrix}. \quad (4.33)$$

By employing the enclosure (4.33), the zonotope $\tilde{\mathcal{X}}_k \supset \hat{\mathcal{X}}_k$ is obtained as follows:

$$\tilde{\mathcal{X}}_k = \{\tilde{\mathcal{G}}_k^x, \tilde{\mathcal{C}}_k^x\} = \hat{\mathcal{G}}_k^x [\zeta^L, \zeta^U], \quad (4.34)$$

where

$$\begin{aligned}\tilde{G}_k^x &= \frac{1}{2}\hat{G}_k^x \text{diag}(\zeta^U - \zeta^L) \\ &= \begin{bmatrix} (T_k A_{k-1} - K_{k-1} C_{k-1}) \tilde{G}_{k-1}^x & T_k \tilde{G}_{k-1}^w & -K_{k-1} \tilde{G}_{k-1}^v & -N_k \tilde{G}_k^v \end{bmatrix},\end{aligned}\quad (4.35)$$

$$\tilde{c}_k^x = \frac{1}{2}\hat{G}_k^x (\zeta^L + \zeta^U), \quad (4.36)$$

with $\tilde{G}_{k-1}^x = \frac{1}{2}\hat{G}_{k-1}^x \text{diag}(\zeta_{k-1}^{Ux} - \zeta_{k-1}^{Lx})$, $\tilde{G}_{k-1}^w = \frac{1}{2}\hat{G}_{k-1}^w \text{diag}(\zeta_{k-1}^{Uw} - \zeta_{k-1}^{Lw})$, $\tilde{G}_{k-1}^v = \frac{1}{2}\hat{G}_{k-1}^v \text{diag}(\zeta_{k-1}^{Uv} - \zeta_{k-1}^{Lv})$, and $\tilde{G}_k^v = \frac{1}{2}\hat{G}_k^v \text{diag}(\zeta_k^{Uv} - \zeta_k^{Lv})$.

Proof. Due to the block-diagonal characteristic of $\hat{A}_k^x$, given by (4.31), each inner constraint block can be individually evaluated to yield the box $[\zeta^L, \zeta^U]$ in (4.33) (Scott et al., 2016, Algorithm 1). To obtain the zonotope $\tilde{\mathcal{X}}_k$ in (4.34), we rewrite the box $[\zeta^L, \zeta^U]$ in terms of affine arithmetic, where the generator matrix $\tilde{G}_k^x$ in (4.35) and the center $\tilde{c}_k^x$ in (4.36) are made explicit in the standard format of (2.8) with null constraints. Since $\mathcal{B}(\hat{A}_k^x, \hat{b}_k^x) \subset \square \mathcal{B}(\hat{A}_k^x, \hat{b}_k^x)$, we ensure that $\hat{\mathcal{X}}_k \subset \tilde{\mathcal{X}}_k$. ■

Remark 4.2.2. Let $\tilde{\mathcal{X}}_{k-1}$, $\mathcal{W}_{k-1}$, $\mathcal{V}_{k-1}$, and $\mathcal{V}_k$ be zero-center CZs with $(n_h, n_h^w, n_{h,k-1}^v, n_{h,k}^v)$ constraints and $(n_g, n_g^w, n_{g,k-1}^v, n_{g,k}^v)$ generators. If solvers of LPs with the same complexity order of Kelner & Spielman (2006) are used in Proposition 4.2.1, we have

$$\begin{aligned}O([\zeta^L, \zeta^U]) &= O([\zeta_{k-1}^{Lx}, \zeta_{k-1}^{Ux}]) + O([\zeta_{k-1}^{Lw}, \zeta_{k-1}^{Uw}]) + O([\zeta_{k-1}^{Lv}, \zeta_{k-1}^{Uv}]) + O([\zeta_k^{Lv}, \zeta_k^{Uv}]) \\ &= O\left(n_g^2 (n_h + n_g)^3 + (n_g^w)^2 (n_h^w + n_g^w)^3 + (n_{g,k-1}^v)^2 (n_{h,k-1}^v + n_{g,k-1}^v)^3 + (n_{g,k}^v)^2 (n_{h,k}^v + n_{g,k}^v)^3\right).\end{aligned}$$

If the iterative method proposed in Scott et al. (2016, Algorithm 1) is used otherwise, we reach a smaller computational cost:

$$\begin{aligned}O([\zeta^L, \zeta^U]) &= O\left(n_h^2 n_g + n_h n_g^2 + (n_h^w)^2 n_g^w + n_h^w (n_g^w)^2\right. \\ &\quad \left.+ (n_{h,k-1}^v)^2 n_{g,k-1}^v + n_{h,k-1}^v (n_{g,k-1}^v)^2 + (n_{h,k}^v)^2 n_{g,k}^v + n_{h,k}^v (n_{g,k}^v)^2\right).\end{aligned}$$

For both cases, note that the partition of $\square \mathcal{B}(A_k^x, b_k^x)$ in four smaller boxes implies reduction of computational cost.

Next, we present how to compute the design matrices T_k , N_k , and K_{k-1} given a quadratic optimization problem involving the cost J_k^g given by (1.9), in which state equality and inequality constraints are transmitted to the desired matrices.

Theorem 4.2.1. Consider the cost function (1.9) whose zonotope $\tilde{\mathcal{X}}_k$ is given by Proposition 4.2.1, the known weight parameter $\eta \in [0, 1]$, and the corresponding inequality (1.4) and equality (3.3) constraints. Let $z = \text{vec}\left(\begin{bmatrix} T_k & N_k & K_{k-1} \end{bmatrix}\right) \in \mathbb{R}^{(n^2+2nm)}$ be the variable vector, where $\text{vec}(\cdot)$ is the vectorization operator that stacks up columns of a matrix starting from

the first one to the last one, and $M \in \mathbb{R}^{(n^2+2nm) \times (n^2+2nm)}$ be the vectorized transpose matrix such that $\begin{bmatrix} \text{vec}(T_k^\top)^\top & \text{vec}(N_k^\top)^\top & \text{vec}(K_{k-1}^\top)^\top \end{bmatrix}^\top = Mz$. Let $\otimes$ be the Kronecker product, and $\text{reshape}(z, n, m)$ be the operator that takes nm specific elements of z to build a matrix with dimension $n \times m$. Then, minimizing (1.9) subject to (1.4) and (3.3) is equivalent to solving the quadratic optimization problem

$$\min \frac{1}{2} z^\top H z \quad \text{s.t.} \quad \check{A}z = \check{b}, \quad \check{D}z \leq d_k^i, \quad (4.37)$$

where

$$H = 2M^\top \check{P} M, \quad (4.38)$$

$$\check{P} = \begin{bmatrix} I_n \otimes \hat{P}_{k|k-1}^{zg} & 0_{n^2 \times nm} & -I_n \otimes \hat{S}_{k-1}^{xy} \\ 0_{nm \times n^2} & I_n \otimes R_k^{zg} & 0_{nm \times nm} \\ -I_n \otimes (\hat{S}_{k-1}^{xy})^\top & 0_{nm \times nm} & I_n \otimes \hat{S}_{k-1}^{yy} \end{bmatrix}, \quad (4.39)$$

$$\check{A} = \begin{bmatrix} E_k^\top & C_k^\top & 0_{n \times m} \end{bmatrix} \otimes I_n, \quad (4.40)$$

$$\check{b} = \text{vec}(I_n), \quad (4.41)$$

$$\check{D} = \check{x}^\top \otimes D_k^i, \quad (4.42)$$

$$\check{x} = \begin{bmatrix} (A_{k-1} \hat{c}_{k-1}^x + B_{k-1} u_{k-1})^\top & y_k^\top & (y_{k-1} - C_{k-1} \hat{c}_{k-1}^x)^\top \end{bmatrix}^\top, \quad (4.43)$$

$$\hat{P}_{k|k-1}^{zg} = A_{k-1} \hat{P}_{k-1}^{zg} A_{k-1}^\top + Q_{k-1}^{zg}, \quad (4.44)$$

$$\hat{P}_{k-1}^{zg} = (1 - \eta) \tilde{G}_{k-1}^x (\tilde{G}_{k-1}^x)^\top + \eta \hat{P}_{k-1}^{xx}, \quad (4.45)$$

$$Q_{k-1}^{zg} = (1 - \eta) \tilde{G}_{k-1}^w (\tilde{G}_{k-1}^w)^\top + \eta Q_{k-1}, \quad (4.46)$$

$$\hat{S}_{k-1}^{xy} = A_{k-1} \hat{P}_{k-1}^{zg} C_{k-1}^\top, \quad (4.47)$$

$$\hat{S}_{k-1}^{yy} = C_{k-1} \hat{P}_{k-1}^{zg} C_{k-1}^\top + R_{k-1}^{zg}, \quad (4.48)$$

$$R_{k-1}^{zg} = (1 - \eta) \tilde{G}_{k-1}^v (\tilde{G}_{k-1}^v)^\top + \eta R_{k-1}, \quad (4.49)$$

$$R_k^{zg} = (1 - \eta) \tilde{G}_k^v (\tilde{G}_k^v)^\top + \eta R_k, \quad (4.50)$$

with optimal solution $\check{z}$, from which we obtain the optimal matrices

$$\check{T}_k = \text{reshape}(\check{z}_{1:n^2}, n, n), \quad (4.51)$$

$$\check{N}_k = \text{reshape}(\check{z}_{(n^2+1):(n^2+nm)}, n, m), \quad (4.52)$$

$$\check{K}_{k-1} = \text{reshape}(\check{z}_{(n^2+nm+1):(n^2+2nm)}, n, m). \quad (4.53)$$

Proof. The original problem to be solved is to minimize the cost function J_k^{zg} , given by (1.9), subject to equality (3.3) and inequality (1.4) constraints. To achieve that, we aim at a standard quadratic formulation with variables $z = \text{vec}(\begin{bmatrix} T_k & N_k & K_{k-1} \end{bmatrix})$.

First, $\begin{bmatrix} T_k & N_k & K_{k-1} \end{bmatrix} \begin{bmatrix} E_k^\top & C_k^\top & 0_{n \times m} \end{bmatrix}^\top = I_n$ is rewritten as $\check{A}z = \check{b}$ using the equivalence $\text{vec}(ABC) = (C^\top \otimes A) \text{vec}(B)$ (Proposition 2.5.1), which results in (4.40) and (4.41). Second,

the inequality constraints (1.4) are applied to the center estimate $\hat{c}_k^x$ given by (4.25), yielding $D_k^i \begin{bmatrix} T_k & N_k & K_{k-1} \end{bmatrix} \check{x} \leq d_k^i$, where $\check{x}$ is given by (4.43). Applying the relation $\text{vec}(ABC) = (C^\top \otimes A)\text{vec}(B)$ once again, we reach (4.42) such that $\check{D}z \leq d_k^i$.

Third, by expanding J_k^{zg} using the relation $\text{tr}(A) + \text{tr}(B) = \text{tr}(A + B)$, we obtain

$$\begin{aligned} \text{tr}(P_k^{zg}) &= \text{tr}\left((T_k A_{k-1} - K_{k-1} C_{k-1}) \hat{P}_{k-1}^{zg} (T_k A_{k-1} - K_{k-1} C_{k-1})^\top\right. \\ &\quad \left.+ T_k Q_{k-1}^{zg} T_k^\top + K_{k-1} R_{k-1}^{zg} K_{k-1}^\top + N_k R_k^{zg} N_k^\top\right) \\ &= \text{tr}\left(T_k \hat{P}_{k|k-1}^{zg} T_k^\top\right) + \text{tr}\left(N_k R_k^{zg} N_k^\top\right) + \text{tr}\left(K_{k-1} \hat{S}_{k-1}^{yy} K_{k-1}^\top\right) \\ &\quad + \text{tr}\left(-K_{k-1} \left(\hat{S}_{k-1}^{xy}\right)^\top T_k^\top\right) + \text{tr}\left(-T_k \hat{S}_{k-1}^{xy} K_{k-1}^\top\right), \end{aligned}$$

whose matrices $\hat{P}_{k|k-1}^{zg}$, $\hat{P}_{k-1}^{zg}$, Q_{k-1}^{zg} , $\hat{S}_{k-1}^{xy}$, $\hat{S}_{k-1}^{yy}$, R_{k-1}^{zg} , and R_k^{zg} are given by (4.44)-(4.50). According to the equivalence $\text{tr}(ABC) = \text{vec}(A^\top)^\top (\text{I} \otimes B) \text{vec}(C)$ (Proposition 2.5.1), each prior trace can be rewritten as

$$\begin{aligned} \text{tr}\left(T_k \hat{P}_{k|k-1}^{zg} T_k^\top\right) &= \text{vec}(T_k^\top)^\top (\text{I}_n \otimes \hat{P}_{k|k-1}^{zg}) \text{vec}(T_k^\top), \\ \text{tr}\left(N_k R_k^{zg} N_k^\top\right) &= \text{vec}(N_k^\top)^\top (\text{I}_n \otimes R_k^{zg}) \text{vec}(N_k^\top), \\ \text{tr}\left(K_{k-1} \hat{S}_{k-1}^{yy} K_{k-1}^\top\right) &= \text{vec}(K_{k-1}^\top)^\top (\text{I}_n \otimes \hat{S}_{k-1}^{yy}) \text{vec}(K_{k-1}^\top), \\ \text{tr}\left(-K_{k-1} \left(\hat{S}_{k-1}^{xy}\right)^\top T_k^\top\right) &= \text{vec}(K_{k-1}^\top)^\top \left(-\text{I}_n \otimes \left(\hat{S}_{k-1}^{xy}\right)^\top\right) \text{vec}(T_k^\top), \\ \text{tr}\left(-T_k \hat{S}_{k-1}^{xy} K_{k-1}^\top\right) &= \text{vec}(T_k^\top)^\top \left(-\text{I}_n \otimes \hat{S}_{k-1}^{xy}\right) \text{vec}(K_{k-1}^\top). \end{aligned}$$

Therefore, J_k^{zg} is equivalent to

$$\text{tr}(P_k^{zg}) = \begin{bmatrix} \text{vec}(T_k^\top) \\ \text{vec}(N_k^\top) \\ \text{vec}(K_{k-1}^\top) \end{bmatrix}^\top \check{P} \begin{bmatrix} \text{vec}(T_k^\top) \\ \text{vec}(N_k^\top) \\ \text{vec}(K_{k-1}^\top) \end{bmatrix},$$

where $\check{P}$ is a symmetric and square matrix given by (4.39). In order to make explicit z in J_k^{zg} , we take a block-diagonal matrix $M = \text{blkdiag}(M_1, M_2, M_3)$ such that $\text{vec}(T_k^\top) = M_1 \text{vec}(T_k)$, $\text{vec}(N_k^\top) = M_2 \text{vec}(N_k)$, and $\text{vec}(K_{k-1}^\top) = M_3 \text{vec}(K_{k-1})$. Finally, the format of (4.37) is achieved after embedding the variable transformation in the matrix H , given by (4.38). Once the quadratic optimization problem (4.37) is solved, the design matrices (4.51)-(4.53) are obtained by applying a reshape in elements from $\check{z}$, which is an inverse procedure to vectorization. ■

Remark 4.2.3. Given H in (4.38), the convexity of the quadratic problem presented in Theorem 4.2.1 relies on $\check{P}$ in (4.39) being positive definite. To achieve that, a sufficient condition is that generator and covariance matrices of noises w_{k-1} , v_{k-1} , and v_k have full rank. However, if H is positive semidefinite, then numerical problems may occur. In this case, it is possible to add a matrix $\delta \text{I}_{(n^2+2nm)}$ to (4.38), with $\delta > 0$ being the smallest possible

value, only to mitigate the singularity of H .

4.2.1.2 Normalized Problem

In Theorem 4.2.1, we show how to compute matrices T_k , N_k , and K_{k-1} from J_k^{zg} in (1.9) for a previously chosen value of the weight parameter η . In this subsection, we show how to produce normalized solutions $\check{T}_k$, $\check{N}_k$, and $\check{K}_{k-1}$ with multiobjective meaning.

Next, two definitions are presented. The first one uses the weighted-sum strategy to normalize costs. The second definition states a cost function that evaluates the distance between two points in the objective space $J_k^z \times J_k^{\text{g}}$.

Definition 4.2.1. Consider the monobjective function $J_k^{\text{zg}}(K_k) = (1 - \eta)J_k^z + \eta J_k^{\text{g}}$, for a chosen weighting parameter $\eta \in [0, 1]$, with K_k being the gain that minimizes J_k^{zg} . The application of $\eta = 0$ yields the limits J_k^{Lz} and J_k^{Ug} , whereas the application of $\eta = 1$ yields the limits J_k^{Uz} and J_k^{Lg} . By assuming these limits are bounded, the costs J_k^z and J_k^{g} may be normalized as

$$\tilde{J}_k^z(\tilde{K}_k) \triangleq \frac{J_k^z - J_k^{\text{Lz}}}{J_k^{\text{Uz}} - J_k^{\text{Lz}}}, \quad (4.54)$$

$$\tilde{J}_k^{\text{g}}(\tilde{K}_k) \triangleq \frac{J_k^{\text{g}} - J_k^{\text{Lg}}}{J_k^{\text{Ug}} - J_k^{\text{Lg}}}, \quad (4.55)$$

where $\tilde{K}_k$ is the corresponding matrix of normalized costs in $[0, 1]$.

Definition 4.2.2. Consider two functions J_k^z and J_k^{g} . Let $\mu_{1,k}, \mu_{2,k} \in \mathbb{R}$ be a reference objective to be reached. Then, the minimum-distance objective function is defined as

$$J_k^{\text{md}}(K_k) \triangleq \left[|\mu_{1,k} - J_k^z|^r + |\mu_{2,k} - J_k^{\text{g}}|^r \right]^{\frac{1}{r}}, \quad (4.56)$$

where K_k is the gain that minimizes J_k^{md} , and $r \in \mathbb{Z}_+$.

Equation (4.56) depicts an appealing case in multiobjective optimization problems, in which we pursuit a point on the Pareto-optimal front as illustrated in Figure 1.2, which denotes global solutions.

Many works involving mixed uncertainties apply $\eta = 0.5$ to unconstrained problems with non-normalized costs to obtain a solution from the Pareto-optimal front (Pan et al., 2022; Noack et al., 2012; Combastel, 2015; Sun et al., 2019). In doing so, the authors are assuming that the uncertainties are well balanced. Therefore, we intuitively expect that the obtained solution is around the middle point of the Pareto curve. Actually, this expected result is justified by Definition 4.2.2, which provides a methodology to compare global solutions. The following fact states when the weighted-sum solution achieves the minimum distance.

Fact 4.2.1. Consider $r = 1$, $\tilde{\eta} = 0.5$, $\tilde{\mu}_{1,k} = \tilde{\mu}_{2,k} = 0$, and the normalized costs $\tilde{J}_k^z$ and $\tilde{J}_k^g$ given by (4.54) and (4.55), respectively, with $\tilde{K}_k$ being the variable matrix. Assume that there are no inequality constraints given by (1.4). Then, minimizing

$$\tilde{J}_k^{zg}(\tilde{K}_k) = (1 - \tilde{\eta})\tilde{J}_k^z + \tilde{\eta}\tilde{J}_k^g \quad (4.57)$$

is equivalent to minimizing

$$\tilde{J}_k^{\text{md}}(\tilde{K}_k) = \left[|\tilde{\mu}_{1,k} - \tilde{J}_k^z|^r + |\tilde{\mu}_{2,k} - \tilde{J}_k^g|^r \right]^{\frac{1}{r}}. \quad (4.58)$$

In this case, $\tilde{\eta} = 0.5$ implies the best tradeoff between the partial costs $\tilde{J}_k^z$ and $\tilde{J}_k^g$ in 1-norm sense.

Proof. Given two costs $J_k^z \in [J_k^{Lz}, J_k^{Uz}]$ and $J_k^g \in [J_k^{Lg}, J_k^{Ug}]$, their normalization as in Definition 4.2.1 yields $\tilde{J}_k^z, \tilde{J}_k^g \in [0, 1]$. Therefore, we can define the normalized monobjective functions $\tilde{J}_k^{zg}$ and $\tilde{J}_k^{\text{md}}$ given by (4.57) and (4.58), respectively. In order to relate these functions, we choose the norm $r = 1$, the scalar parameter $\tilde{\eta} = 0.5$, and the reference objective $\tilde{\mu}_{1,k} = 0 = \tilde{\mu}_{2,k}$. Since the normalized partial costs are nonnegative, we have $\tilde{J}_k^{zg} = 0.5\tilde{J}_k^{\text{md}}$. Then, the functions $\tilde{J}_k^{zg}$ and $\tilde{J}_k^{\text{md}}$ assume different values under the same variable matrix $\tilde{K}_k$.

However, there exist cases in which $\tilde{K}_k$ is the same global solution for both functions, such as those involving unconstrained or equality-constrained problems. The unconstrained case can be verified doing $\frac{\partial \tilde{J}_k^{zg}}{\partial \tilde{K}_k} = \frac{\partial (0.5\tilde{J}_k^{\text{md}})}{\partial \tilde{K}_k} = \frac{\partial \tilde{J}_k^{\text{md}}}{\partial \tilde{K}_k} = 0$. The equality-constrained case can be verified as follows. By employing a Lagrange function $L(\cdot)$, the equality constraints $h(\tilde{K}_k) = 0$ are transmitted to the normalized functions as a common additive parcel, that is,

$$\begin{aligned} \tilde{J}_k^{zg+} &= \tilde{J}_k^{zg} + L(h(\tilde{K}_k), \Lambda_k), \\ \tilde{J}_k^{\text{md}+} &= \tilde{J}_k^{\text{md}} + L(h(\tilde{K}_k), \Lambda_k), \end{aligned}$$

where Λ_k denotes the Lagrange multipliers. By computing the first derivatives and equalling them to zero, we have

$$\begin{aligned} \frac{\partial \tilde{J}_k^{zg+}}{\partial \tilde{K}_k} &= \frac{\partial (\tilde{J}_k^{zg} + L(h(\tilde{K}_k), \Lambda_k))}{\partial \tilde{K}_k} = \frac{\partial (0.5\tilde{J}_k^{\text{md}})}{\partial \tilde{K}_k} + \frac{\partial L(h(\tilde{K}_k), \Lambda_k)}{\partial \tilde{K}_k} = 0, \\ \frac{\partial \tilde{J}_k^{\text{md}+}}{\partial \tilde{K}_k} &= \frac{\partial \tilde{J}_k^{\text{md}}}{\partial \tilde{K}_k} + \frac{\partial L(h(\tilde{K}_k), \Lambda_k)}{\partial \tilde{K}_k} = 0. \end{aligned}$$

Subtracting $\frac{\partial \tilde{J}_k^{zg+}}{\partial \tilde{K}_k} = 0$ from $\frac{\partial \tilde{J}_k^{\text{md}+}}{\partial \tilde{K}_k} = 0$, we reach the same result than the unconstrained case. Then, the two depicted cases show that $\tilde{\eta} = 0.5$ provides the best tradeoff between the partial costs $\tilde{J}_k^z$ and $\tilde{J}_k^g$ in 1-norm sense.

Finally, whenever there exist inequality constraints on $\tilde{K}_k$, we cannot equal the deriva-

tives to zero. Then, weighted sum does not imply minimum distance anymore. ■

Remark 4.2.4. Fact 4.2.1 could be stated with non-normalized but nonnegative functions. However, Definition 4.2.1 is useful to establish the reference objective as the Utopia point, which represents where the costs would be minimum at the same time. Then, Definition 4.2.1 may improve the tradeoff between the partial costs.

Thanks to the convexity of the quadratic problem (4.37) in Theorem 4.2.1, we guarantee that the obtained solution $\check{z}$ yields the partial costs J_k^z and J_k^g , given by (1.8) and (1.5) respectively, on the Pareto-optimal front. Conversely, the optimization problem contains inequality constraints and, according to Fact 4.2.1, Theorem 4.2.1 must be reformulated in terms of Definition 4.2.2 to yield the final solutions with a multiobjective perspective. This result is presented in the following theorem for the normalized cost function

$$\tilde{J}_k^{\text{md}} = \tilde{J}_k^z + \tilde{J}_k^g, \quad (4.59)$$

where $\tilde{J}_k^z$ and $\tilde{J}_k^g$ are the partial costs J_k^z (1.8) and J_k^g (1.5) normalized as in Definition 4.2.1.

Theorem 4.2.2. Consider the cost function (4.59) whose zonotope $\tilde{X}_k$ is given by Proposition 4.2.1, and the corresponding inequality (1.4) and equality (3.3) constraints. Let $z = \text{vec}\left(\begin{bmatrix} \tilde{T}_k & \tilde{N}_k & \tilde{K}_{k-1} \end{bmatrix}\right) \in \mathbb{R}^{(n^2+2nm)}$ be the variable vector, and let $M \in \mathbb{R}^{(n^2+2nm) \times (n^2+2nm)}$ be the vectorized transpose matrix such that $\begin{bmatrix} \text{vec}(\tilde{T}_k^\top)^\top & \text{vec}(\tilde{N}_k^\top)^\top & \text{vec}(\tilde{K}_{k-1}^\top)^\top \end{bmatrix}^\top = Mz$. Then, minimizing (4.59) subject to (1.4) and (3.3) is equivalent to solving the quadratic optimization problem

$$\min \frac{1}{2} z^\top H z \quad \text{s.t.} \quad \tilde{A}z = \tilde{b}, \quad \tilde{D}z \leq \tilde{d}_k,$$

where H , $\tilde{A}$, $\tilde{b}$, and $\tilde{D}$ are given by (4.38), (4.40), (4.41), and (4.42), respectively, assuming that

$$\hat{P}_{k-1}^{zg} = \frac{1}{J_k^{\text{Uz}} - J_k^{\text{Lz}}} \tilde{G}_{k-1}^x (\tilde{G}_{k-1}^x)^\top + \frac{1}{J_k^{\text{Ug}} - J_k^{\text{Lg}}} \hat{P}_{k-1}^{\text{xx}}, \quad (4.60)$$

$$Q_{k-1}^{zg} = \frac{1}{J_k^{\text{Uz}} - J_k^{\text{Lz}}} \tilde{G}_{k-1}^w (\tilde{G}_{k-1}^w)^\top + \frac{1}{J_k^{\text{Ug}} - J_k^{\text{Lg}}} Q_{k-1}, \quad (4.61)$$

$$R_{k-1}^{zg} = \frac{1}{J_k^{\text{Uz}} - J_k^{\text{Lz}}} \tilde{G}_{k-1}^v (\tilde{G}_{k-1}^v)^\top + \frac{1}{J_k^{\text{Ug}} - J_k^{\text{Lg}}} R_{k-1}, \quad (4.62)$$

$$R_k^{zg} = \frac{1}{J_k^{\text{Uz}} - J_k^{\text{Lz}}} \tilde{G}_k^v (\tilde{G}_k^v)^\top + \frac{1}{J_k^{\text{Ug}} - J_k^{\text{Lg}}} R_k. \quad (4.63)$$

Thereby, we obtain the optimal vector $\check{z}$, and thereby, the optimal matrices

$$\check{\tilde{T}}_k = \text{reshape}(\check{z}_{1:n^2}, n, n), \quad (4.64)$$

$$\check{\tilde{N}}_k = \text{reshape}(\check{z}_{(n^2+1):(n^2+nm)}, n, m), \quad (4.65)$$

$$\check{\tilde{K}}_{k-1} = \text{reshape}(\check{z}_{(n^2+nm+1):(n^2+2nm)}, n, m). \quad (4.66)$$

Proof. By making explicit the costs J_k^z and J_k^g , given by (1.8) and (1.5) respectively, in (4.59), we have

$$\tilde{J}_k^{\text{md}} = \frac{1}{J_k^{\text{Uz}} - J_k^{\text{Lz}}} J_k^z + \frac{1}{J_k^{\text{Ug}} - J_k^{\text{Lg}}} J_k^g - \frac{1}{J_k^{\text{Uz}} - J_k^{\text{Lz}}} J_k^{\text{Lz}} - \frac{1}{J_k^{\text{Ug}} - J_k^{\text{Lg}}} J_k^{\text{Lg}}.$$

As the last two terms of $\tilde{J}_k^{\text{md}}$ are constant, they can be removed during the optimization process. In so doing, we obtain a cost similar to J_k^{zg} in Theorem 4.2.1, whose difference is the replacement of $(1 - \eta)$ by $\frac{1}{J_k^{\text{Uz}} - J_k^{\text{Lz}}}$, and of η by $\frac{1}{J_k^{\text{Ug}} - J_k^{\text{Lg}}}$. The development and the solution of the new optimization problem follow the prior modified theorem. ■

4.2.1.3 Algorithm

The LTVMF algorithm solves Problem 1.2.4 by means of Theorems 4.2.1 and 4.2.2. In short, the first theorem is used to normalize costs whereas the second theorem provides normalized solutions with multiobjective meaning.

Regarding the CZ $\hat{\mathcal{X}}_k$ in (4.28), note that its number of constraints and generators is larger than the initial values n_h and n_g , respectively. In order to mitigate such growth over iterations, we here apply CZR to $\hat{\mathcal{X}}_{k-1}$, yielding

$$\mathcal{X}_{k-1}^\downarrow = \text{CZR}(\hat{\mathcal{X}}_{k-1}, \varphi_c, \varphi_g) = \{G_{k-1}^{\downarrow x}, x_{k-1}^\downarrow, A_{k-1}^{\downarrow x}, b_{k-1}^{\downarrow x}\} \quad (4.67)$$

with φ_c constraints and φ_g generators, and employ the resulting CZ $\mathcal{X}_{k-1}^\downarrow$ instead of $\hat{\mathcal{X}}_{k-1}$. As $x_{k-1}^\downarrow$ may be nonzero, we shift it to the center estimate $\hat{c}_{k-1}^x$ to obtain a conventional zero-center CZ. Next, LTVMF is described in Algorithm 4.2 and illustrated in Figure 4.4.

Algorithm 4.2: $[\hat{c}_k^x, \hat{\mathcal{X}}_k, \hat{\mathcal{X}}_k, \mathcal{X}_k^\alpha, \mathcal{I}_k^\alpha] = \text{LTVMF}(\hat{c}_{k-1}^x, \hat{\mathcal{X}}_{k-1}, \hat{\mathcal{X}}_{k-1}, D_k^i, d_k^i, M, E_k, A_{k-1}, B_{k-1}, u_{k-1}, \mathcal{W}_{k-1}, W_{k-1}, y_{k-1}, y_k, C_{k-1}, C_k, \mathcal{V}_{k-1}, V_{k-1}, \mathcal{V}_k, V_k, \varphi_c, \varphi_g, \alpha)$

- 1 Compute $\mathcal{X}_{k-1}^\downarrow$ given by (4.67)
 - 2 Apply Proposition 4.2.1 to obtain the generator matrices $\tilde{G}_{k-1}^x$, $\tilde{G}_{k-1}^w$, $\tilde{G}_{k-1}^v$, and $\tilde{G}_k^v$
 - 3 Given $\eta = 0$, compute the covariance matrices given by (4.44)–(4.50)
 - 4 Apply Theorem 4.2.1 to compute the matrices $\tilde{T}_k$, $\tilde{N}_k$, and $\tilde{K}_{k-1}$
 - 5 Compute $\tilde{G}_k^x$ and $\hat{P}_k^{\text{xx}}$ using (4.35) and (4.32), respectively, to determine J_k^{Lz} in (1.8) and J_k^{Ug} in (1.5)
 - 6 Execute the steps (3)–(5) with $\eta = 1$ to compute J_k^{Uz} in (1.8) and J_k^{Lg} in (1.5)
 - 7 Apply Theorem 4.2.2 to compute the final matrices $\tilde{T}_k$, $\tilde{N}_k$, and $\tilde{K}_{k-1}$
 - 8 Determine the mixed state estimates $\hat{c}_k^x$ (4.25), $\hat{\mathcal{X}}_k$, and $\hat{\mathcal{X}}_k$, where $\hat{\mathcal{X}}_k$ is given by (4.28), (4.30)–(4.31) and $\hat{\mathcal{X}}_k$ is given by (4.29), (4.32)
 - 9 Compute the confidence CZ $\mathcal{X}_k^\alpha$ and the confidence box $\mathcal{I}_k^\alpha$ using (2.47) and (2.49), respectively
-

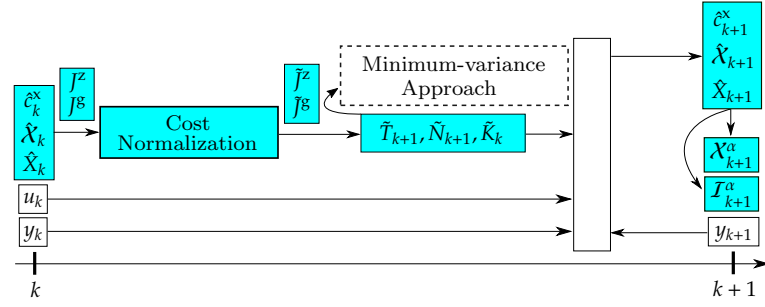


Figure 4.4: The LTVMF algorithm is illustrated by a diagram, whose cyan boxes highlight its main items. Given mixed uncertainties at k , LTVMF solves an online constrained minimum-variance problem with normalized costs. This problem is formulated in terms of design matrices $\tilde{T}_{k+1}$, $\tilde{N}_{k+1}$, and $\tilde{K}_k$, which yield mixed state estimates at $k+1$ and, thereby, a confidence CZ $\mathcal{X}_{k+1}^a$ and a confidence box $\mathcal{I}_{k+1}^a$ that merge the mixed estimates in compact sets.

4.2.2 Particular Case with Polytopes

In Subsection 4.2.1, we present the general algorithm to solve Problem 1.2.4. Now, we present an optional strategy to estimate states when the inequality constraints (1.4) imply polytopes $\mathcal{X}_k^F$ (or feasibility sets); see Figure 2.5. The idea is to first minimize J_k^{zg} , given by (1.9), subject to equality constraints (3.3) and, thereby, to compute state estimates; after, a step is appended to correct such estimates using inequality constraints (1.4) for the polytopic case. This procedure aims at reducing the computational burden, since quadratic programming solvers can be replaced by *explicit* solutions as in Tang et al. (2020). For the unconstrained cases, we just do not enforce state constraints.

The following corollaries address the main results, where the design matrices are obtained explicitly. They are special cases of Theorems 4.2.1 and 4.2.2 by disregarding inequality constraints. To achieve that, the equality constraints (3.3) are transmitted to the cost functions J_k^{zg} and $\tilde{J}_k^{md}$, given by (1.9) and (4.59) respectively, through the Lagrange multipliers $\Lambda_k \in \mathbb{R}^{n \times n}$, yielding the Lagrange functions

$$J_k^{\text{eqzg}}(T_k, N_k, K_{k-1}, \Lambda_k) \triangleq J_k^{zg} + \text{tr}(\Lambda_k (E_k^\top T_k^\top + C_k^\top N_k^\top - I_n)), \quad (4.68)$$

$$\tilde{J}_k^{\text{eqzg}}(\tilde{T}_k, \tilde{N}_k, \tilde{K}_{k-1}, \tilde{\Lambda}_k) \triangleq \tilde{J}_k^{md} + \text{tr}(\tilde{\Lambda}_k (E_k^\top \tilde{T}_k^\top + C_k^\top \tilde{N}_k^\top - I_n)). \quad (4.69)$$

Corollary 4.2.1. Let $\Lambda_k \in \mathbb{R}^{n \times n}$ be the Lagrange multipliers and $\Psi = [T_k \ N_k \ K_{k-1}] \in \mathbb{R}^{n \times (n+2m)}$ the variable matrix. Then, minimizing (1.9) subject to (3.3) is equivalent to minimizing J_k^{eqzg} in (4.68), from which we obtain the optimal matrices

$$\check{\Lambda}_k = -(\Pi_k^\top \Sigma_k^{-1} \Pi_k)^{-1}, \quad (4.70)$$

$$\check{\Psi} = -\check{\Lambda}_k \Pi_k^\top \Sigma_k^{-1}, \quad (4.71)$$

where Π_k is given by (3.14),

$$\Sigma_k = 2 \begin{bmatrix} \hat{P}_{k|k-1}^{zg} & 0_{n \times m} & -\hat{S}_{k-1}^{xy} \\ 0_{m \times n} & R_k^{zg} & 0_{m \times m} \\ -(\hat{S}_{k-1}^{xy})^\top & 0_{m \times m} & \hat{S}_{k-1}^{yy} \end{bmatrix}, \quad (4.72)$$

and the covariance matrices $\hat{P}_{k|k-1}^{zg}$, $\hat{S}_{k-1}^{xy}$, $\hat{S}_{k-1}^{yy}$, and R_k^{zg} are given by (4.44)-(4.50). Thereby, the design matrices are made explicit as follows:

$$\check{T}_k = \check{\Psi}_{1:n, 1:n}, \quad (4.73)$$

$$\check{N}_k = \check{\Psi}_{1:n, (n+1):(n+m)}, \quad (4.74)$$

$$\check{K}_{k-1} = \check{\Psi}_{1:n, (n+m+1):(n+2m)}. \quad (4.75)$$

Proof. The unconstrained optimization problem to be solved is to minimize J_k^{eqzg} in (4.68). According to Proposition 2.5.1, taking the partial derivative of J_k^{eqzg} with respect to T_k , N_k , K_{k-1} , and $\Lambda_k \in \mathbb{R}^{n \times n}$ yields

$$\frac{\partial J_k^{\text{eqzg}}}{\partial T_k} = 2T_k \hat{P}_{k|k-1}^{zg} - 2K_{k-1} (\hat{S}_{k-1}^{xy})^\top + \Lambda_k, \quad (4.76)$$

$$\frac{\partial J_k^{\text{eqzg}}}{\partial N_k} = 2N_k R_k^{zg} + \Lambda_k C_k^\top, \quad (4.77)$$

$$\frac{\partial J_k^{\text{eqzg}}}{\partial K_{k-1}} = 2K_{k-1} \hat{S}_{k-1}^{yy} - 2T_k \hat{S}_{k-1}^{xy}, \quad (4.78)$$

$$\frac{\partial J_k^{\text{eqzg}}}{\partial \Lambda_k} = T_k E_k + N_k C_k - I_n, \quad (4.79)$$

where $\hat{P}_{k|k-1}^{zg}$, $\hat{S}_{k-1}^{xy}$, $\hat{S}_{k-1}^{yy}$, and R_k^{zg} are given by (4.44)-(4.50). Henceforth, the remaining proof is similar to the proof of Proposition 3.1.1, in which a linear system is derived to determine the optimal solutions (4.73)-(4.75). ■

Corollary 4.2.2. Let $\tilde{\Lambda}_k \in \mathbb{R}^{n \times n}$ be the Lagrange multipliers and $\tilde{\Psi} = [\tilde{T}_k \quad \tilde{N}_k \quad \tilde{K}_{k-1}] \in \mathbb{R}^{n \times (n+2m)}$ the variable matrix. Then, minimizing (4.59) subject to (3.3) is equivalent to minimizing $\tilde{J}_k^{\text{eqzg}}$ in (4.69), from which we obtain the optimal matrices

$$\begin{aligned} \check{\tilde{\Lambda}}_k &= -(\Pi_k^\top \Sigma_k^{-1} \Pi_k)^{-1}, \\ \check{\tilde{\Psi}} &= -\check{\tilde{\Lambda}}_k \Pi_k^\top \Sigma_k^{-1}, \end{aligned}$$

where

$$\Pi_k = \begin{bmatrix} E_k^\top & C_k^\top & 0_{n \times m} \end{bmatrix}^\top,$$

$$\Sigma_k = 2 \begin{bmatrix} \hat{P}_{k|k-1}^{zg} & 0_{n \times m} & -\hat{S}_{k-1}^{xy} \\ 0_{m \times n} & R_k^{zg} & 0_{m \times m} \\ -(\hat{S}_{k-1}^{xy})^\top & 0_{m \times m} & \hat{S}_{k-1}^{yy} \end{bmatrix},$$

and the covariance matrices $\hat{P}_{k|k-1}^{zg}$, $\hat{S}_{k-1}^{xy}$, $\hat{S}_{k-1}^{yy}$, and R_k^{zg} are given by (4.60)-(4.63). Thereby, the design matrices are made explicit as follows:

$$\check{T}_k = \check{\Psi}_{1:n,1:n}, \quad (4.80)$$

$$\check{N}_k = \check{\Psi}_{1:n,(n+1):(n+m)}, \quad (4.81)$$

$$\check{K}_{k-1} = \check{\Psi}_{1:n,(n+m+1):(n+2m)}. \quad (4.82)$$

Proof. By making explicit the costs J_k^z (1.8) and J_k^g (1.5) in (4.69), we have

$$\tilde{J}_k^{\text{eqzg}} = \frac{1}{J_k^{\text{Uz}} - J_k^{\text{Lz}}} J_k^z + \frac{1}{J_k^{\text{Ug}} - J_k^{\text{Lg}}} J_k^g + \text{tr} \left(\tilde{\Lambda}_k \left(E_k^\top \tilde{T}_k^\top + C_k^\top \tilde{N}_k^\top - I_n \right) \right) - \frac{1}{J_k^{\text{Uz}} - J_k^{\text{Lz}}} J_k^{\text{Lz}} - \frac{1}{J_k^{\text{Ug}} - J_k^{\text{Lg}}} J_k^{\text{Lg}}.$$

As the last two terms of $\tilde{J}_k^{\text{eqzg}}$ are constant, they can be removed during the optimization process. In so doing, we obtain a cost similar to J_k^{eqzg} in Corollary 4.2.1, whose difference is the replacement of $(1 - \eta)$ by $\frac{1}{J_k^{\text{Uz}} - J_k^{\text{Lz}}}$, and of η by $\frac{1}{J_k^{\text{Ug}} - J_k^{\text{Lg}}}$. The development and the solution of the new optimization problem follow the prior modified corollary. ■

Remark 4.2.5. According to Fact 4.2.1, Corollary 4.2.2 could be equivalently presented with $\tilde{\eta} = 0.5$.

So far, we show how to compute the design matrices $\tilde{T}_k$, $\tilde{N}_k$, and $\tilde{K}_{k-1}$, which are employed to compute the first state estimates $\hat{c}_k^x$ in (4.25), $\hat{X}_k$ in (4.28), and $\hat{X}_k$ in (4.29). Now, we discuss how to enforce inequality constraints (1.4) by means of an additional step. To achieve that, $D_k^i x_k \leq d_k^i$ in (1.4) is rewritten as the CZ $\mathcal{X}_k^F = \{G_k^{xF}, c_k^{xF}, A_k^{xF}, b_k^{xF}\}$.

First, we replace the zero center of the CZ $\hat{X}_k$ by the center c_k^{xF} of the CZ $\mathcal{X}_k^F$, obtaining

$$\hat{X}_k^+ = \{\hat{G}_k^x, c_k^{xF}, \hat{A}_k^x, \hat{b}_k^x\}. \quad (4.83)$$

After, using (2.29) yields

$$\hat{X}_k^{t+} = \hat{X}_k^+ \cap_{\text{In}} \mathcal{X}_k^F. \quad (4.84)$$

Soon after, we cancel the shift c_k^{xF} of $\hat{X}_k^{t+}$, yielding the desired zero-center CZ $\hat{X}_k^t$. Second, we correct the center $\hat{c}_k^x$ and covariance $\hat{P}_k^{\text{xx}}$ estimates through some stochastic approach, such as ICUT and PT discussed in Appendix A. Here, we employ ICUT with the spreading parameter $\kappa = 0$ to obtain

$$[\hat{c}_k^{\text{xt}}, \hat{P}_k^{\text{xtt}}] = \text{ICUT}(\hat{c}_k^x, \hat{P}_k^{\text{xx}}, x_k^{\text{min}}, x_k^{\text{max}}, \kappa), \quad (4.85)$$

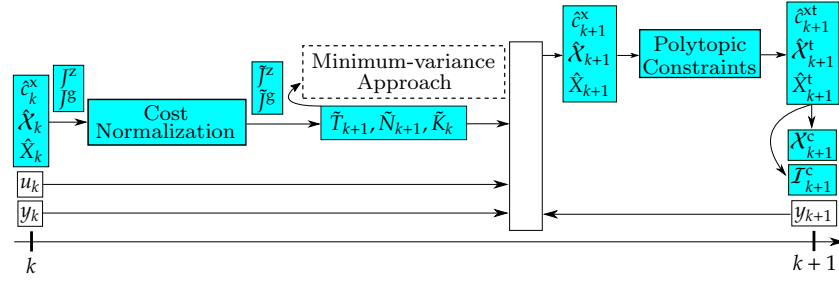


Figure 4.5: The explicit LTVMF algorithm is illustrated by a diagram, whose cyan boxes highlight its main items. Given mixed uncertainties at k , LTVMF explicitly solves a minimum-variance problem containing equality constraints and normalized costs, whose solutions are the design matrices $\tilde{T}_{k+1}$, $\tilde{N}_{k+1}$, and $\tilde{K}_k$. These matrices are used to yield the first mixed state estimates at $k+1$. In the presence of polytopic constraints, intersection (for CZ) and some stochastic technique (for both center and GRV) are applied to correct the prior state estimates. Finally, the new state estimates are used to compute a constrained confidence CZ $\mathcal{X}_{k+1}^c$ and a constrained confidence box $\mathcal{I}_{k+1}^c$ that merge the mixed estimates in compact sets.

where

$$[\mathcal{X}_k^{\min}, \mathcal{X}_k^{\max}] = \square \mathcal{X}_k^F. \quad (4.86)$$

Finally, the estimates $\hat{c}_k^{\text{xt}}$, $\hat{X}_k^{\text{t}}$, and $\hat{P}_k^{\text{xx}}$ are used to find the confidence CZ $\mathcal{X}_k^\alpha$ in (2.47) and the confidence box $\mathcal{I}_k^\alpha$ in (2.49). The precision of these sets can still be improved by means of intersection with $\mathcal{X}_k^F$, yielding the corresponding constrained sets

$$\mathcal{X}_k^c \triangleq \mathcal{X}_k^\alpha \cap_{\text{In}} \mathcal{X}_k^F, \quad (4.87)$$

$$\mathcal{I}_k^c \triangleq \square \{\mathcal{I}_k^\alpha \cap_{\text{In}} \mathcal{X}_k^F\}. \quad (4.88)$$

Next, the explicit LTVMF algorithm (LTVMF-E) is described in Algorithm 4.3 and illustrated in Figure 4.5.

Algorithm 4.3: $[\hat{c}_k^x, \hat{X}_k, \hat{X}_k^c, \mathcal{X}_k^c, \mathcal{I}_k^c] = \text{LTVMF-E}(\hat{c}_{k-1}^x, \hat{X}_{k-1}, \hat{X}_{k-1}^c, \mathcal{X}_k^F, E_k, A_{k-1}, B_{k-1}, u_{k-1}, \mathcal{W}_{k-1}, W_{k-1}, y_{k-1}, y_k, C_{k-1}, C_k, \mathcal{V}_{k-1}, V_{k-1}, \mathcal{V}_k, V_k, \varphi_c, \varphi_g, \alpha, \kappa)$

- 1 Apply the steps 1–8 of Algorithm 4.2 with Corollaries 4.2.1 and 4.2.2 (instead of Theorems 4.2.1 and 4.2.2) to compute the mixed state estimates $\hat{c}_k^x$ in (4.25), $\hat{X}_k$ in (4.28), and $\hat{X}_k^c$ in (4.29)
 - 2 Compute the CZ $\hat{X}_k^{\text{t+}}$ given by (4.84)
 - 3 Annul the center of $\hat{X}_k^{\text{t+}}$ to obtain the zero-center CZ $\hat{X}_k^{\text{t}}$
 - 4 Compute $\hat{c}_k^{\text{xt}}$ and $\hat{P}_k^{\text{xx}}$ using (4.85)
 - 5 Let the mixed state estimates be given by $\hat{c}_k^x = \hat{c}_k^{\text{xt}}$, $\hat{X}_k = \hat{X}_k^{\text{t}}$, and $\hat{P}_k^{\text{xx}} = \hat{P}_k^{\text{xx}}$
 - 6 Compute the constrained confidence CZ $\mathcal{X}_k^c$ and the constrained confidence box $\mathcal{I}_k^c$ given by (4.87) and (4.88), respectively
-

Remark 4.2.6. The correction of $\hat{X}_k$ (4.28) via intersection with $\mathcal{X}_k^F$ requires a translation

over Euclidean space since $\hat{\mathcal{X}}_k$ does not contain the whole state information. Here, we opt by (4.84) because the center reference does not depend on state estimates. Other possible choices of translation are: (i) to replace the center $\hat{c}_k^{\mathbf{x}^{\mathbf{F}}}$ of $\mathcal{X}_k^{\mathbf{F}}$ by the zero center of $\hat{\mathcal{X}}_k$; or (ii) to shift $\hat{\mathcal{X}}_k$ to the center estimate $\hat{c}_k^{\mathbf{x}}$. Besides altering the center of specific sets, each choice can imply different solutions.

4.2.3 Numerical Example

4.2.3.1 Setup

Consider a three-state LTV descriptor system (Tang et al., 2020) of the form (4.22)-(4.23), where

$$E_k = \begin{bmatrix} 1 + 0.2 \sin(0.1k) & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad A_k = \begin{bmatrix} 0.6 - 0.1 \sin(0.1k) & 0 & 0.3 \\ -0.2 \sin(0.2k) & 0.4 & 0 \\ 0 & 1 & 1 \end{bmatrix}, \quad B_k = \begin{bmatrix} 1 \\ \sin(0.1k) \\ 0 \end{bmatrix},$$

$C_k = [0 \quad 1 + 0.2 \sin(0.2k) \quad 0.5]$, $Q = 0.2826 \times 10^{-4} \mathbf{I}_3$, $G^w = 0.02 \mathbf{I}_3$, $R = 2.7778 \times 10^{-6}$, $G^v = 0.005$, and $u_k = 0.5 \sin(0.15k)$. The realizations z_{k-1}^w and z_k^v are taken from uniform distributions, while the realizations w_{k-1}^g and g_k^v are taken from GRVs. This example illustrates the application of LTVMF in LTV descriptor systems.

To simulate the states, we set $x_0 = [0.1 \quad 0 \quad 0]^T$, $k_f = 80$, $k_0 = 4$, and $m_s = 1$. To estimate the states, we set $\hat{c}_0^{\mathbf{x}} = \mathbf{0}_{3 \times 1}$, $\hat{\mathcal{X}}_0 = \{0.1 \mathbf{I}_3, \mathbf{0}_{3 \times 1}, \mathbf{1}_{1 \times 3}, \mathbf{1}\}$, $\hat{P}_0^{\mathbf{xx}} = 0.0007 \mathbf{I}_3$, $\varphi_c = 3$, $\varphi_g = 8$, and $\alpha = 0.0027$, such that $x_0 \in \mathcal{X}_0^{\alpha}$.

Since there are no inequality constraints (1.4), only the explicit version of LTVMF (Algorithm 4.3) is executed. In this case, we are going to obtain the same state estimates than the quadratic version (Algorithm 4.2) with smaller T^{CPU} . To yield punctual estimates $\bar{x}_k$, we do:

$$\bar{x}_k = \arg \min_{x_k} \|x_k - \hat{c}_k^{\mathbf{x}}\|_1, \quad \text{s.t.} \quad x_k \in \mathcal{X}_k^{\alpha}. \quad (4.89)$$

4.2.3.2 Discussion

In Figure 4.6a, the punctual estimate $\bar{x}_k \in \mathcal{X}_k^{\alpha}$ and the true state x_k evolve over a specific region in the Euclidean space, which is related to linear invariant embedded in the descriptor representation. In order to illustrate the mixing of the state estimates, we also sketch some confidence boxes $\mathcal{I}_k^{\alpha}$. Regarding the estimate $\bar{x}_k$, we now rise a brief discussion. Due to the guaranteed characteristic of CZs (to include the true states x_k) and to the *a posteriori* center estimate $\hat{c}_k^{\mathbf{x}}$ satisfies all state constraints, we expect in most cases that $\bar{x}_k = \hat{c}_k^{\mathbf{x}}$; this occurs in Figure 4.6a. It is noteworthy that by assuming that $\hat{c}_k^{\mathbf{x}}$ is close enough to x_k , the confidence CZ $\mathcal{X}_k^{\alpha}$ may include its center $\hat{c}_k^{\mathbf{x}}$, resulting in $\bar{x}_k = \hat{c}_k^{\mathbf{x}}$. However, since $\hat{c}_k^{\mathbf{x}} \in \mathcal{X}_k^{\alpha}$ cannot be enforced over the optimization problems, the use of

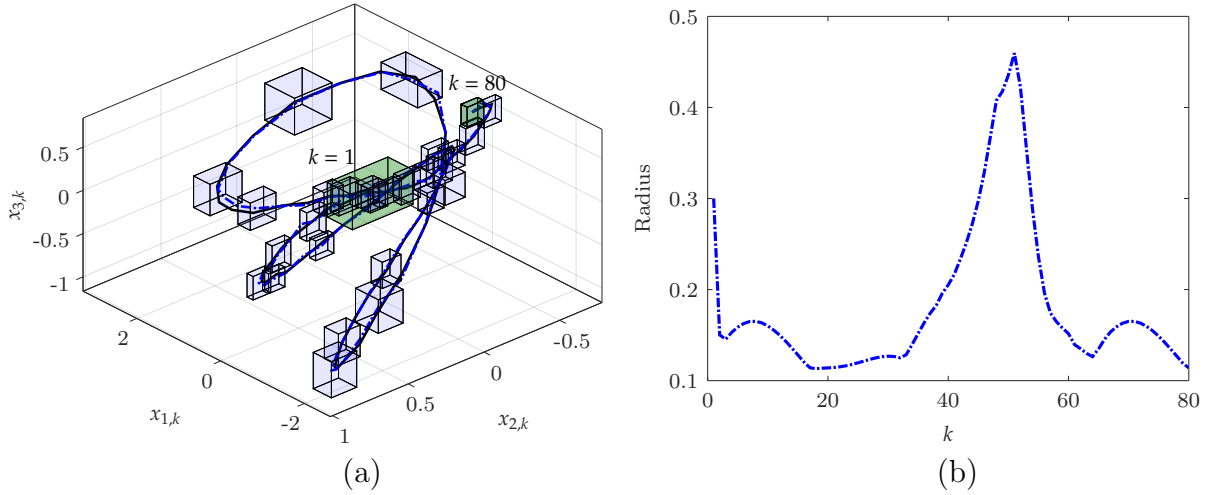


Figure 4.6: State estimation for the three-state LTV system. In (a), the true states (black solid line) and the punctual estimates computed by LTVMF (blue dashed-dotted line) are sketched in 3D along with some confidence boxes. The green boxes make explicit the initial and final sets. In (b), the maximum radius of confidence boxes computed by LTVMF is presented over time.

(4.89) is required to yield punctual estimates $\bar{x}_k \in X_k^\alpha$. Therefore, $\bar{x}_k$ cannot have the same properties than $\hat{c}_k^x$, although it is the best approximation of $\hat{c}_k^x$ (in 1-norm sense) that tries to comply with the state constraints. This discussion includes both quadratic and explicit versions of LTVMF.

Figure 4.6b illustrates that the maximum radius of $\mathcal{I}_k^\alpha$ varies over time. This behavior is expected because the system is LTV. The peak value around $k = 50$ is justified by the growth of volume verified in $\hat{X}_k$ given by (4.28), which is due to the small values of φ_c and φ_g . Besides, the trace minimization does not necessarily imply volume minimization.

4.3 Mixed-Uncertainty Filter for Descriptor LTI Systems

4.3.1 Estimator Design

Consider the LTI version of (4.22)-(4.23) given by:

$$Ex_k = Ax_{k-1} + Bu_{k-1} + z_{k-1}^w + g_{k-1}^w, \quad (4.90)$$

$$y_k = Cx_k + z_k^v + g_k^v, \quad (4.91)$$

where $E \in \mathbb{R}^{n \times n}$, $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times p}$, and $C \in \mathbb{R}^{m \times n}$ are known constant matrices. Without loss of generality, the process and measurement noises are expressed in terms of their linear mapping, that is, $z_{k-1}^w = G^w \xi_{k-1}^w$, $z_k^v = G^v \xi_k^v$, $g_{k-1}^w = F^w v_{k-1}^w$, and $g_k^v = F^v v_k^v$, where $G^w \in \mathbb{R}^{n \times n_g^w}$ and $G^v \in \mathbb{R}^{m \times n_g^v}$ correspond to generator matrices of known constant CZs $\mathcal{W} = \{G^w, 0_{n \times 1}, A^w, b^w\}$ and $\mathcal{V} = \{G^v, 0_{m \times 1}, A^v, b^v\}$, respectively, with $\xi_{k-1}^w \in \mathcal{B}(A^w, b^w)$ and $\xi_k^v \in \mathcal{B}(A^v, b^v)$, while $F^w \in$

$\mathbb{R}^{n \times n}$ and $F^v \in \mathbb{R}^{m \times m}$ define covariance matrices $Q = F^w (F^w)^\top$ and $R = F^v (F^v)^\top$ of constant GRVs $W \sim \mathcal{N}(0_{n \times 1}, Q)$ and $V \sim \mathcal{N}(0_{m \times 1}, R)$, respectively, and $v_{k-1}^w \in \mathbb{R}^n$ and $v_k^v \in \mathbb{R}^m$ are realizations from normalized GRVs with zero mean and identity covariance.

For the system (4.90)-(4.91), (3.3) is reduced to (3.30). Following similar steps to (4.25)-(4.27), we propose the center estimate

$$\hat{c}_k^x \triangleq \tilde{A}\hat{c}_{k-1}^x + TBu_{k-1} + Ky_{k-1} + Ny_k \quad (4.92)$$

with error terms

$$\hat{z}_k^x = \tilde{A}\hat{z}_{k-1}^x + TG^w \xi_{k-1}^w - KG^v \xi_{k-1}^v - NG^v \xi_k^v, \quad (4.93)$$

$$\hat{g}_k^x = \tilde{A}\hat{g}_{k-1}^x + TF^w v_{k-1}^w - KF^v v_{k-1}^v - NF^v v_k^v, \quad (4.94)$$

where $\tilde{A} = TA - KC$.

For convenience, the error terms (4.93) and (4.94) are rewritten as the error system

$$e_k^s = \tilde{A}e_{k-1}^s + \tilde{B}d_{k-1}^s, \quad (4.95)$$

$$z_k^s = e_k^s, \quad (4.96)$$

where $s = \{z, g\}$, $e_k^s = x_k^s$, $\tilde{B}^z = [TG^w \quad -KG^v \quad -NG^v] \in \mathbb{R}^{n \times (n_g^w + 2n_g^v)}$, $d_{k-1}^z = [(\xi_{k-1}^w)^\top \quad (\xi_{k-1}^v)^\top \quad (\xi_k^v)^\top]^\top \in \mathbb{R}^{(n_g^w + 2n_g^v)}$, $\tilde{B}^g = [TF^w \quad -KF^v \quad -NF^v] \in \mathbb{R}^{n \times (n + 2m)}$, and $d_{k-1}^g = [(v_{k-1}^w)^\top \quad (v_{k-1}^v)^\top \quad (v_k^v)^\top]^\top \in \mathbb{R}^{(n + 2m)}$.

We formulate a mixed H_2/H_∞ problem to obtain the constant matrices T , N , and K . Now, we justify why normalized realizations ξ and elements $v \in \mathcal{B}(A, b)$ are considered. Both H_2 and H_∞ norms in the time domain work on l_2 norm, thus all signals evaluated must be bounded in magnitude. On one hand, the constrained zonotopic terms z_{k-1}^w , z_{k-1}^v , and z_k^v are bounded by CZs, therefore H_∞ norm is here employed to ensure *robustness*. H_∞ norm is not modified by realizations of bounded inputs, but it is influenced by linear mapping of variables; therefore, we decide to make explicit all linear transformations $\tilde{B}^z$ on d_{k-1}^z . On the other hand, the Gaussian terms g_{k-1}^w , g_{k-1}^v , and g_k^v can assume any value in the Euclidean space, then system norms cannot be directly used. According to Khargonekar et al. (1996), H_2 norm can be applied to Gaussian error systems to minimize the steady-state variance, which is bounded. Specifically, when Gaussian inputs are *white* and their power spectral density is equal to *identity matrix*, the steady-state variance is equivalent to computing the H_2 norm of error system. In Khargonekar et al. (1996), this case is discussed for continuous-time systems, while discrete-time systems are approached in Subsection 3.1.2. In order to obtain identity power spectral density for g^w and g^v , we can use Cholesky decomposition, yielding $\tilde{B}^g$ and d_{k-1}^g . Then, we employ H_∞ norm to minimize the worst-case elements of $\hat{z}_k^x$ (4.93), and H_2 norm to minimize the steady-state variance of $\hat{g}_k^x$ (4.94).

Solving an optimization problem containing both H_2 and H_∞ norms is not a simple task. One of the difficulties is to obtain the extreme points of the Pareto-optimal front, since the non-minimized norm may reach a too large value. It means that the weighted-sum approach is not suitable to scale the biobjective optimization problem. Commonly, this type of problem is solved using the epsilon-constraint or minimum-distance approach (Khargonekar et al., 1996; Ozbay & Bachmann, 1994; Rego & Raffo, 2019). Furthermore, an iterative process on the upper bound of the square H_∞ norm can be performed to adjust the final solution. Independently of which strategy is employed, we solve convex optimization problems and, thereby, any solution is on the Pareto-optimal front. Here, we are interested in finding a good tradeoff between the objectives as in LTVMF. Therefore, we formulate a constrained optimization problem in which the summation of the square H_2 ($\psi > 0$) and H_∞ ($\gamma > 0$) norms is minimized. According to Definition 4.2.2, this procedure is equivalent to setting $r = 1$ and $\mu_1 = 0 = \mu_2$, and assuming nonnegative costs (to remove the absolute value). Next, the constrained optimization problem is presented in terms of linear matrix inequalities.

Theorem 4.3.1. Consider variables $P_1 = P_1^\top > 0_{n \times n}$, $P_2 = P_2^\top > 0_{n \times n}$, $M \in \mathbb{R}^{n \times n}$ an invertible matrix, $H = H^\top > 0_{(n+2m) \times (n+2m)}$, $Y_1 \in \mathbb{R}^{n \times (n+m)}$, $Y_2 \in \mathbb{R}^{n \times m}$, $\psi > 0$, and $\gamma > 0$, where ψ is the square H_2 norm of the Gaussian error system (4.95)-(4.96), for $s = g$, and γ is the square H_∞ norm of the constrained zonotopic error system (4.95)-(4.96), for $s = z$. Solving the constrained optimization problem

$$\min \quad (\psi + \gamma) \quad (4.97)$$

$$\text{s.t.} \quad \begin{bmatrix} I_n - P_1 & \Omega_1^\top \\ \Omega_1 & P_1 - M - M^\top \end{bmatrix} < 0_{2n \times 2n}, \quad (4.98)$$

$$\begin{bmatrix} -H & \Omega_2^\top \\ \Omega_2 & P_1 - M - M^\top \end{bmatrix} < 0_{2(n+m) \times 2(n+m)}, \quad \text{tr}(H) < \psi, \quad (4.99)$$

$$\begin{bmatrix} I_n - P_2 & 0_{n \times (n_g^w + 2n_g^v)} & \Omega_1^\top \\ 0_{(n_g^w + 2n_g^v) \times n} & -\gamma I_{(n_g^w + 2n_g^v)} & \Omega_3^\top \\ \Omega_1 & \Omega_3 & P_2 - M - M^\top \end{bmatrix} < 0_{\vartheta \times \vartheta}, \quad (4.100)$$

where $\vartheta = 2n + n_g^w + 2n_g^v$,

$$\Omega_1 = M^\top \Theta^\dagger \alpha_1 A + Y_1 \Psi \alpha_1 A - Y_2 C, \quad (4.101)$$

$$\Omega_2 = \begin{bmatrix} (M^\top \Theta^\dagger \alpha_1 + Y_1 \Psi \alpha_1) F^w & -Y_2 F^v & -(M^\top \Theta^\dagger \alpha_2 + Y_1 \Psi \alpha_2) F^v \end{bmatrix}, \quad (4.102)$$

$$\Omega_3 = \begin{bmatrix} (M^\top \Theta^\dagger \alpha_1 + Y_1 \Psi \alpha_1) G^w & -Y_2 G^v & -(M^\top \Theta^\dagger \alpha_2 + Y_1 \Psi \alpha_2) G^v \end{bmatrix}, \quad (4.103)$$

and with Θ , $\Theta^\dagger$, α_1 , α_2 , and Ψ given by (3.40), we obtain the optimal matrices

$$\check{T} = \Theta^\dagger \alpha_1 + (\check{M}^\top)^{-1} \check{Y}_1 \Psi \alpha_1, \quad (4.104)$$

$$\check{N} = \Theta^\top \alpha_2 + (\check{M}^\top)^{-1} \check{Y}_1 \Psi \alpha_2, \quad (4.105)$$

$$\check{K} = (\check{M}^\top)^{-1} \check{Y}_2. \quad (4.106)$$

Proof. Based on Lemma 2.5.2, the square H_2 norm of the error system (4.95)-(4.96) with $s = g$ is computed as

$$\begin{aligned} \min \quad & \psi \\ \text{s.t.} \quad & \tilde{A}^\top P_1 \tilde{A} - P_1 + I_n < 0_{n \times n}, \quad (\tilde{B}^g)^\top P_1 \tilde{B}^g < H, \quad \text{tr}(H) < \psi. \end{aligned} \quad (4.107)$$

Applying Schur's complement to the inequality constraint $\tilde{A}^\top P_1 \tilde{A} - P_1 + I_n < 0_{n \times n}$, we have

$$\begin{bmatrix} I_n - P_1 & (TA - KC)^\top \\ (TA - KC) & -P_1^{-1} \end{bmatrix} < 0_{2n \times 2n}. \quad (4.108)$$

In order to let the Lyapunov matrix P_1 free over linearization, we apply a congruence transformation to (4.108) with $\Upsilon = \text{blkdiag}(I_n, M)$, and M is a slack variable. Post- and pre-multiplying (4.108) by $\Upsilon^\top$ and Υ , respectively, we obtain

$$\begin{bmatrix} I_n - P_1 & (TA - KC)^\top M \\ M^\top (TA - KC) & -M^\top P_1^{-1} M \end{bmatrix} < 0_{2n \times 2n}. \quad (4.109)$$

To treat the nonlinearity regarding P_1 , we make use of $-M^\top P_1^{-1} M < P_1 - M - M^\top$ (De Oliveira et al., 1999, Theorem 1).

Since T and N are related by (3.30), we make explicit such dependency as in (3.38)-(3.39). As the product $M^\top (TA - KC)$ is nonlinear, we define the linearizing transformations $Y_1 = M^\top S$ and $Y_2 = M^\top K$ to become (4.109) the linear matrix inequality (4.98).

Now, by applying Schur's complement to the inequality constraint $(\tilde{B}^g)^\top P_1 \tilde{B}^g < H$, we have

$$\begin{bmatrix} -H & (\tilde{B}^g)^\top \\ \tilde{B}^g & -P_1^{-1} \end{bmatrix} < 0_{2(n+m) \times 2(n+m)}. \quad (4.110)$$

Following similar steps to (4.109) to let P_1 free, we reach

$$\begin{bmatrix} -H & (\tilde{B}^g)^\top M \\ M^\top \tilde{B}^g & P_1 - M - M^\top \end{bmatrix} < 0_{2(n+m) \times 2(n+m)}. \quad (4.111)$$

As the product $M^\top \tilde{B}^g$ is nonlinear, we use the linearizations Y_1 and Y_2 to become (4.111) the linear matrix inequality (4.99).

Based on Lemma 2.5.3, the square H_∞ norm of the error system (4.95)-(4.96) with $s = z$ is computed as

$$\min \quad \gamma$$

$$\text{s.t.} \quad \begin{bmatrix} \tilde{A}^\top P_2 \tilde{A} + \mathbf{I}_n - P_2 & \tilde{A}^\top P_2 \tilde{B}^z \\ (\tilde{B}^z)^\top P_2 \tilde{A} & (\tilde{B}^z)^\top P_2 \tilde{B}^z - \gamma \mathbf{I}_\varrho \end{bmatrix} < 0_{(n+\varrho) \times (n+\varrho)}, \quad (4.112)$$

where $\varrho = n_g^w + 2n_g^v$. The inequality constraint (4.112) can be rewritten as

$$\begin{bmatrix} \mathbf{I}_n - P_2 & 0_{n \times \varrho} \\ 0_{\varrho \times n} & -\gamma \mathbf{I}_\varrho \end{bmatrix} + \begin{bmatrix} \tilde{A}^\top \\ (\tilde{B}^z)^\top \end{bmatrix} P_2 \begin{bmatrix} \tilde{A} & \tilde{B}^z \end{bmatrix} < 0_{(n+\varrho) \times (n+\varrho)}.$$

Applying Schur's complement, we obtain

$$\begin{bmatrix} \begin{bmatrix} \mathbf{I}_n - P_2 & 0_{n \times \varrho} \\ 0_{\varrho \times n} & -\gamma \mathbf{I}_\varrho \end{bmatrix} & \begin{bmatrix} \tilde{A}^\top \\ (\tilde{B}^z)^\top \end{bmatrix} \\ \begin{bmatrix} \tilde{A} & \tilde{B}^z \end{bmatrix} & -P_2^{-1} \end{bmatrix} < 0_{(2n+\varrho) \times (2n+\varrho)}.$$

Following similar steps to (4.109) to let the Lyapunov matrix P_2 free, we reach

$$\begin{bmatrix} \mathbf{I}_n - P_2 & 0_{n \times \varrho} & \tilde{A}^\top M \\ 0_{\varrho \times n} & -\gamma \mathbf{I}_\varrho & (\tilde{B}^z)^\top M \\ M^\top \tilde{A} & M^\top \tilde{B}^z & P_2 - M - M^\top \end{bmatrix} < 0_{(2n+\varrho) \times (2n+\varrho)}.$$

This matrix inequality is equivalent to the linear matrix inequality (4.100) by linearizing $M^\top \tilde{B}^z$ with Ω_3 (4.103) (analogously to Ω_2 (4.102)). Finally, by minimizing the summation of the square H_2 (ψ) and H_∞ (γ) norms, the constrained optimization problem (4.97)-(4.100) is finished, whose outputs are the design matrices $\tilde{T}$, $\tilde{N}$, and $\tilde{K}$ given by (4.104)-(4.106). ■

Remark 4.3.1. In Theorem 4.3.1, we consider different matrices P_1 and P_2 . This procedure requires to add a slack variable M to match the mixed H_2/H_∞ formulation. In doing so, we obtain a stronger formulation than with $P_1 = P_2$ because there exist more degrees of freedom to reach a better Pareto-optimal front.

Given the matrices (4.104)-(4.106), we show how to compute a confidence zonotope $\mathcal{X}_k^\alpha$ from (2.47) and a confidence box $\mathcal{I}_k^\alpha$ from (2.49) at each time step. First, the center estimate $\hat{x}_k^\alpha$ is directly given by (4.92). Second, the GRV $\hat{X}_k \sim \mathcal{N}(0_{n \times 1}, \hat{P}_k^{\text{xx}})$ is obtained through (3.33).

Third, we employ reachable sets analysis to efficiently compute the interval hull of the CZ

$$\hat{\mathcal{X}}_k = \tilde{A} \hat{\mathcal{X}}_{k-1} \oplus T \mathcal{W} \oplus (-K) \mathcal{V} \oplus (-N) \mathcal{V} = \{\hat{G}_k^\alpha, 0_{n \times 1}, \hat{A}_k^\alpha, \hat{b}_k^\alpha\} \quad (4.113)$$

without storing increasing-order matrices and vectors over iterations. Unlike (4.28), CZR is not required anymore. On the one hand, the one-step ahead estimation returns CZs at the cost of both larger reachable set and complexity (due to the order-reduction algorithms). On the other hand, the reachable set analysis here exploited provides interval hull of exact CZs (with no order reduction) through low-cost computations.

Consider the constrained zonotopic term $\hat{z}_k^x$ in (4.93). From $\hat{z}_1^x$ to $\hat{z}_k^x$, the reachability analysis leads to

$$\begin{aligned}\hat{z}_1^x &= \tilde{A}\hat{G}_0^x\xi_0^x + TG^w\xi_0^w - KG^v\xi_0^v - NG^v\xi_1^v, \\ \hat{z}_2^x &= \tilde{A}^2\hat{G}_0^x\xi_0^x + \tilde{A}TG^w\xi_0^w + TG^w\xi_1^w - \tilde{A}KG^v\xi_0^v - KG^v\xi_1^v - \tilde{A}NG^v\xi_1^v - NG^v\xi_2^v, \\ &\vdots \\ \hat{z}_k^x &= \tilde{A}^k\hat{G}_0^x\xi_0^x + \sum_{i=0}^{k-1} \tilde{A}^iTG^w\xi_{k-1-i}^w - \sum_{i=0}^{k-1} \tilde{A}^iKG^v\xi_{k-1-i}^v - \sum_{i=0}^{k-1} \tilde{A}^iNG^v\xi_{k-i}^v.\end{aligned}$$

Note that there exist common terms between ξ_{k-1-i}^v and ξ_{k-i}^v . After combining them, we reach the following recursive form:

$$\hat{z}_k^x = \tilde{A}^k\hat{G}_0^x\xi_0^x + \sum_{i=0}^{k-1} \tilde{A}^iTG^w\xi_{k-1-i}^w - \tilde{A}^{k-1}KG^v\xi_0^v - NG^v\xi_k^v - \sum_{i=0}^{k-2} \tilde{A}^i(\tilde{A}N + K)G^v\xi_{k-1-i}^v, \quad \forall k \geq 2. \quad (4.114)$$

Replacing all realizations ξ by constrained unitary boxes, we obtain

$$\begin{aligned}\hat{\mathcal{X}}_k &= \tilde{A}^k\hat{G}_0^x\mathcal{B}(\hat{A}_0^x, \hat{b}_0^x) \oplus \bigoplus_{i=0}^{k-1} \tilde{A}^iTG^w\mathcal{B}(A^w, b^w) \oplus (-\tilde{A}^{k-1}KG^v)\mathcal{B}(A^v, b^v) \oplus (-NG^v)\mathcal{B}(A^v, b^v) \\ &\oplus \bigoplus_{i=0}^{k-2} (-\tilde{A}^i(\tilde{A}N + K)G^v)\mathcal{B}(A^v, b^v), \quad \forall k \geq 2.\end{aligned} \quad (4.115)$$

From Property 2.2.1, we have

$$\begin{aligned}\square\hat{\mathcal{X}}_k &= \square\{\tilde{A}^k\hat{G}_0^x\mathcal{B}(\hat{A}_0^x, \hat{b}_0^x)\} \oplus \bigoplus_{i=0}^{k-1} \square\{\tilde{A}^iTG^w\mathcal{B}(A^w, b^w)\} \oplus \square\{-\tilde{A}^{k-1}KG^v\mathcal{B}(A^v, b^v)\} \\ &\oplus \square\{-NG^v\mathcal{B}(A^v, b^v)\} \oplus \bigoplus_{i=0}^{k-2} \square\{-\tilde{A}^i(\tilde{A}N + K)G^v\mathcal{B}(A^v, b^v)\}, \quad \forall k \geq 2.\end{aligned} \quad (4.116)$$

According to (4.116), we should store some products and summations within auxiliary variables (Π and ζ) to aid the next iterations.

Finally, given $\hat{c}_k^x$ in (4.92), $\hat{P}_k^{xx}$ in (3.33), and $\square\hat{\mathcal{X}}_k = \{\text{diag}(\text{rad}(\square\hat{\mathcal{X}}_k)), \text{mid}(\square\hat{\mathcal{X}}_k)\}$ in (4.116), we apply Definitions 2.4.3 and 2.4.4 to yield both confidence zonotope $\mathcal{X}_k^\alpha$ and confidence box $\mathcal{I}_k^\alpha$; $\mathcal{X}_k^\alpha$ is a zonotope because this is result of the Minkowski sum between a parallelotope $\mathcal{P}_k$ and a box $\square\hat{\mathcal{X}}_k$.

4.3.2 Algorithm Description

The LTIMF algorithm solves Problem 1.2.4. As the reachability analysis requires that all state evolution ($\hat{c}_k^x$, $\hat{\mathcal{X}}_k$, and $\hat{X}_k$) is traceable since $k = 1$, inequality constraints (1.4) cannot be enforced during the state estimation. In fact, this is an intrinsic limitation of LTIMF, but it is also the reason of efficient computations. We expect to compensate

the lack of inequality constraints with a good precision (due to degrees of freedom in T and N , and to the slack matrix M). Anyway, for the unconstrained cases, we just do not enforce state constraints. In comparison with LTVMF, the LTIMF is motivated for LTI descriptor systems since it demands a smaller computational cost and returns good interval hulls of CZs (due to the reachable set analysis).

Next, LTIMF is described in Algorithm 4.4 and illustrated in Figure 4.7. We assume that Theorem 4.3.1 has already been used to yield constant matrices T , N , and K . The LTIMF is presented in de Paula et al. (2022b).

Algorithm	4.4:	$\left[\hat{c}_k^x, \Pi_k^x, \hat{X}_k, \Pi_k^w, \zeta_k^w, \Pi_k^{va}, \Pi_k^{vc}, \zeta_k^{vc}, \mathcal{X}_k^\alpha, \mathcal{I}_k^\alpha \right]$	$=$
$\text{LTIMF}\left(\hat{c}_{k-1}^x, \Pi_{k-1}^x, \hat{X}_{k-1}, \Pi_{k-1}^w, \zeta_{k-1}^w, \Pi_{k-1}^{va}, \zeta_{k-1}^{vb}, \Pi_{k-1}^{vc}, \zeta_{k-1}^{vc}, T, N, K, \tilde{A}, B, u_{k-1}, \hat{X}_0, \mathcal{W}, W, y_{k-1}, y_k, \mathcal{V}, V, \alpha, k\right)$			
1	if $k == 1$ then		
2	Input: $\Pi_0^x = \hat{G}_0^x$, $\Pi_0^w = \Pi_0^{va} = \Pi_0^{vc} = \zeta_0^w = \zeta_0^{vc} = \emptyset$, $\zeta_0^{vb} = \square\{-NG^v\mathcal{B}(A^v, b^v)\}$, where $\emptyset$ represents the empty set		
3	Update: $\Pi_1^x = \tilde{A}\Pi_0^x$, $\Pi_1^w = TG^w$, $\Pi_1^{va} = -KG^v$, $\Pi_1^{vc} = \emptyset$, $\zeta_1^w = \square\{\Pi_1^w\mathcal{B}(A^w, b^w)\}$, $\zeta_1^{vc} = [0_{n \times 1}, 0_{n \times 1}]$		
4	else if $k == 2$ then		
5	$\Pi_2^x = \tilde{A}\Pi_1^x$, $\Pi_2^w = \tilde{A}\Pi_1^w$, $\Pi_2^{va} = \tilde{A}\Pi_1^{va}$, $\Pi_2^{vc} = -(\tilde{A}N + K)G^v$, $\zeta_2^w = \square\{\Pi_2^w\mathcal{B}(A^w, b^w)\} + \zeta_1^w$, $\zeta_2^{vc} = \square\{\Pi_2^{vc}\mathcal{B}(A^v, b^v)\}$		
6	else		
7	$\Pi_k^x = \tilde{A}\Pi_{k-1}^x$, $\Pi_k^w = \tilde{A}\Pi_{k-1}^w$, $\Pi_k^{va} = \tilde{A}\Pi_{k-1}^{va}$, $\Pi_k^{vc} = \tilde{A}\Pi_{k-1}^{vc}$, $\zeta_k^w = \square\{\Pi_k^w\mathcal{B}(A^w, b^w)\} + \zeta_{k-1}^w$, $\zeta_k^{vc} = \square\{\Pi_k^{vc}\mathcal{B}(A^v, b^v)\} + \zeta_{k-1}^{vc}$		
8	end		
9	$\square\hat{X}_k = \square\{\Pi_k^x\mathcal{B}(\hat{A}_0^x, \hat{b}_0^x)\} + \zeta_k^w + \square\{\Pi_k^{va}\mathcal{B}(A^v, b^v)\} + \zeta_k^{vb} + \zeta_k^{vc}$		
10	Compute the center estimate $\hat{c}_k^x$ given by (4.92)		
11	Determine the GRV $\hat{X}_k$ (4.29) with covariance given by (3.33)		
12	Determine the confidence zonotope $\mathcal{X}_k^\alpha$ and the confidence box $\mathcal{I}_k^\alpha$ using (2.47) and (2.49), respectively		

Remark 4.3.2. When state inequality constraints (1.4) imply polytopes, they can be applied to the confidence sets of LTIMF as in (4.87) and (4.88). Advantageously, this procedure improves the LTIMF precision without affecting the reachability analysis.

4.3.3 Numerical Example

In this subsection, we experiment the LTVMF and LTIMF algorithms. In order to fairly compare our algorithms with literature results, we implement the mixed-uncertainty filter proposed in Pan et al. (2022) called CSMSE. For comparison purposes, we compute the indexes T^{CPU} in (3.57), RMSE in (3.58), and r^α in (3.84).

To calculate the punctual estimate $\bar{x}$ in RMSE, (4.89) is employed. In turn, r^α is computed over confidence boxes $\mathcal{I}^\alpha$. Then, we analyze both accuracy (RMSE) and precision

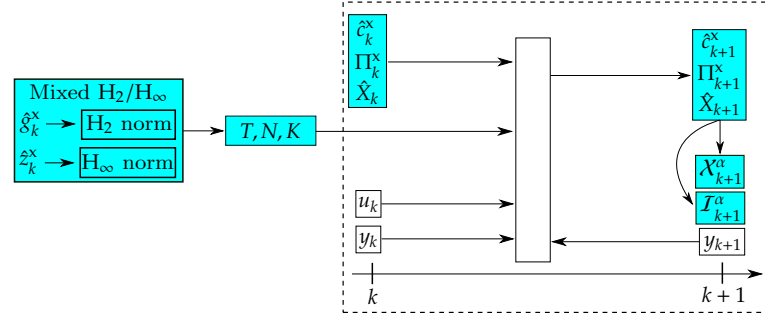


Figure 4.7: Illustrative diagram of the LTIMF algorithm, where cyan boxes are used to highlight the main items. Such state estimator assumes that the constant design matrices T , N , and K are previously defined by means of a mixed H_2/H_∞ optimization problem containing only state equality constraints. No state inequality constraint is enforced by LTIMF due to the conservation of the reachable sets promotes a computationally efficient loop. At $k + 1$, mixed state estimates are obtained and merged through a confidence zonotope $\mathcal{I}_{k+1}^\alpha$ and a confidence box $\mathcal{I}_{k+1}^\alpha$.

($r^\square$) of algorithms. Since linear programming solvers demand an expressive execution time, we opt by Remark 2.4.3 to calculate the interval hull of CZs. To provide fair analysis, Definitions 2.4.3 and 2.4.4 are employed to merge mixed estimates for CSMSE.

The following computer configuration was used: 8 GB RAM 1333 MHz, Windows 10 Pro, and AMD FX-6300 CPU 3.50 GHz. All implementations were executed in MATLAB 9.11 (Release) with Gurobi 9.1, INTLAB 12 (Rump, 1999), and CVX 2.1 (Grant & Boyd, 2014). Specifically, Gurobi is used to solve linear and quadratic programs, INTLAB is employed to sketch sets and to compute interval estimates, and CVX is used to solve semidefinite programming problems.

4.3.3.1 Setup

Consider the discrete-time LTI compartmental system (Teixeira et al., 2009)

$$\begin{aligned} x_k &= Ax_{k-1} + Bu_{k-1} + w_{k-1}, \\ y_k &= Cx_k + v_k, \end{aligned}$$

where $A = \begin{bmatrix} 0.94 & 0.028 & 0.019 \\ 0.038 & 0.95 & 0.001 \\ 0.022 & 0.022 & 0.98 \end{bmatrix}$, $B = 0_{3 \times 1}$, and $C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$. The state vector x_k obeys

the conservation law $D^e x_k = d^e$, where $D^e = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}$ and $d^e = 3$. Moreover, inherently, compartmental systems involve nonnegative quantities (Bernstein & Hyland, 1993), that is, $x_k \geq 0_{3 \times 1}$. We assume that the noise terms $w_{k-1} = z_{k-1}^w + g_{k-1}^w$ and $v_k = z_k^v + g_k^v$ are such that $z_{k-1}^w \in \mathcal{W} = \left\{ \begin{bmatrix} 0.5M & 0_{3 \times 1} \end{bmatrix}, 0_{3 \times 1} \right\}$ and $z_k^v \in \mathcal{V} = \{0.5I_2, 0_{2 \times 1}\}$ are taken from uniform distribu-

tions, with $M = \begin{bmatrix} 0.05 & -0.03 \\ -0.02 & 0.01 \\ -0.03 & 0.02 \end{bmatrix}$, whereas g_{k-1}^w and g_k^v are taken from Gaussian distributions $W \sim \mathcal{N}(0_{3 \times 1}, F^w (F^w)^\top)$ and $V \sim \mathcal{N}(0_{2 \times 1}, F^v (F^v)^\top)$ truncated in the interval hulls of confidence ellipsoids with $\alpha = 0.0027$, where $F^w = \begin{bmatrix} \frac{0.5}{\sqrt{\zeta_w}} M & 0_{3 \times 1} \end{bmatrix}$, $F^v = \frac{0.5}{\sqrt{\zeta_v}} I_2$, and $\zeta_w = \zeta_v = 11.83$. The initial state vector $x_0 = 1_{3 \times 1}$ belongs to the initial confidence CZ $\mathcal{X}_0^\alpha$ (2.47), which is characterized by the center $\hat{c}_0^x = \begin{bmatrix} 2 & 1 & 0 \end{bmatrix}^\top$, CZ $\hat{\mathcal{X}} = \{\hat{G}_0^x, 0_{3 \times 1}, \hat{A}_0^x, \hat{b}_0^x\}$, and GRV $\hat{X} \sim \mathcal{N}(0_{3 \times 1}, \frac{1}{\zeta_x} I_3)$, where $\zeta_x = 14.16$, and

$$\begin{aligned} \hat{G}_0^x &= \begin{bmatrix} 1.5I_3 & 0_{3 \times 4} & 1.5 \times 1_{3 \times 1} \end{bmatrix}, \\ \hat{A}_0^x &= \begin{bmatrix} -1.5 & 0 & 0 & -1.5 & 0 & 0 & 0 & 0 \\ 0 & -1.5 & 0 & 0 & -1.5 & 0 & 0 & 0 \\ 0 & 0 & -1.5 & 0 & 0 & -1.5 & 0 & 0 \\ 1.5 & 1.5 & 1.5 & 0 & 0 & 0 & -1.5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad \hat{b}_0^x = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -3 \\ 1 \end{bmatrix}. \end{aligned}$$

To simulate the prior system, we set $k_0 = 10$, $k_f = 1000$, and $m_s = 100$. To estimate states, we set the parameters $\varphi_c = 3$, $\varphi_g = 8$, and $\alpha = 0.0027$. On the one hand, the CSMSE algorithm solves unconstrained optimization problems, then its simulation is here executed with the dynamic equations of the compartmental system. This is a fair decision because x_k already satisfies the state constraints.

On the other hand, the LTIMF and LTVMF algorithms solve constrained optimization problems. To achieve a descriptor representation from a state-space representation, we employed the methodology discussed in Ko & Bitmead (2005, Subsection 2.1) and simulated the filters. However, numerical errors are imparted to the obtained representation, making unlikely the satisfaction of the equality constraint with respect to center, covariance, and covariation. Therefore, we decided to compensate the lack of the descriptor by augmenting the measurement model with the equality constraint such that

$$\begin{aligned} \begin{bmatrix} y_k \\ d^e \end{bmatrix} &= \begin{bmatrix} C \\ D^e \end{bmatrix} x_k + \begin{bmatrix} v_k \\ 0 \end{bmatrix}, \\ \tilde{y}_k &= \tilde{C} x_k + \tilde{v}_k. \end{aligned}$$

After, we set descriptor matrix $E = I_3$ and augmented variables $\tilde{\mathcal{V}} = \{G^\vee, 0_{3 \times 1}\}$ and $\tilde{V} \sim \mathcal{N}(0_{3 \times 1}, F^\vee (F^\vee)^\top)$, where $G^\vee = \text{blkdiag}(0.5I_2, 10^{-6})$ and $F^\vee = \text{blkdiag}(F^v, 10^{-4})$. Note that the matrices were augmented with nonzero terms to mitigate numerical issues such as Cholesky decomposition and matrix inversion. Accordingly, we add $10^{-8}I_{27}$ to matrix H in (4.38) to complete the positive definiteness. Therefore, both quadratic and explicit versions of LTVMF can be employed. By solving the optimization problem in Theorem 4.3.1, we

obtain the following design matrices for LTIMF:

$$T = \begin{bmatrix} 1.0511 & 0.0802 & 0.0804 \\ 0.0087 & 1.0014 & 0.0045 \\ 0.0257 & 0.0040 & 1.0006 \end{bmatrix}, N = \begin{bmatrix} 0.0292 & 0.0002 & -0.0805 \\ -0.0042 & 0.0031 & -0.0046 \\ -0.0251 & -0.0034 & -0.0007 \end{bmatrix},$$

$$K = \begin{bmatrix} 0.0281 & 0.0004 & 0.0723 \\ -0.0032 & 0.0029 & 0.0167 \\ -0.0251 & -0.0034 & 0.0430 \end{bmatrix}.$$

In order to experiment the different algorithms proposed in this paper, we execute two scenarios of state estimation. The first one corresponds to running LTVMF 4.2 and LTIMF 4.4 with the original halfspace constraints $x_k \geq 0_{3 \times 1}$. In this case, LTVMF is carried out with its quadratic version, while LTIMF does not enforce any state inequality constraint. In the second scenario, LTVMF 4.3 and LTIMF 4.4 enforce the polytopic constraints $\mathcal{X}^F = \{\text{diag}(\text{rad}([\Lambda]), \text{mid}([\Lambda]))\}$, where $[\Lambda] = \begin{bmatrix} [0, 5] \\ [0, 5] \end{bmatrix}$ was defined by simulation. In this case, LTVMF is carried out with its explicit version (see Subsection 4.2.2), while LTIMF enforces such constraints as in Remark 4.3.2. For both cases, CSMSE returns the same solution since it cannot enforce state constraints.

4.3.3.2 Discussion

The LTVMF, LTIMF, and CSMSE algorithms solve minimum-variance problems. Then, in comparison with the stochastic framework (Teixeira et al., 2009), the solution of those problems brings up implications to estimates of center, and covariance and covariation matrices. To test the satisfaction of equality constraint, we verify if $D^e \hat{c}_k^x = d_k^e$, $D^e \hat{P}_k^{xx} = 0_{1 \times 3}$, and $D^e \tilde{G}_k^x (\tilde{G}_k^x)^\top = 0_{1 \times 3}$, with $\tilde{G}_k^x$ being the generator matrix of zonotope used to define the minimum-variance problem, as in Theorems 4.2.1 and 4.2.2. However, since zonotopes participate only of the optimization problem, we only expect that the center estimates comply with the equality constraint.

In Figure 4.8, we illustrate that each algorithm returns different solutions. In terms of precision, CSMSE is the most conservative method. The justification is that, although the dynamic equations of the compartmental system are compatible with the equality constraints, both covariance $\hat{P}_0^{xx}$ and covariation $\tilde{G}_0^x (\tilde{G}_0^x)^\top$ matrices were not initialized in $D^e x_0 = d^e$ (Teixeira et al., 2009). The correct initialization of the covariation matrix (consistent with minimum variance) is not trivial since we do not have the explicit link between covariation matrix and the corresponding CZ, only the inverted procedure is known and makes sense. Therefore, CSMSE cannot by itself guarantee that its state estimates are consistent with invariants. This also justifies why CSMSE is the most inaccurate method, as shown by Table 4.3.

Table 4.3: Results of T^{CPU} , RMSE, and $r^\square$ for the compartmental system. The suffix “P” denotes that polytopic constraints were enforced by LTIMF.

Algorithms	T^{CPU}	RMSE ₁	RMSE ₂	RMSE ₃	$r^\square$
CSMSE (Pan et al., 2022)	6.30 ms	0.217	0.0683	6.16	39.2
LTVMF	25.2 ms	0.0546	0.0385	0.0664	0.233
LTVMF-E	12.5 ms	0.133	0.140	0.201	1.30
LTIMF	3.30 ms	0.0538	0.0296	0.0614	0.737
LTIMF-P	4.40 ms	0.0538	0.0296	0.0614	0.734

The prior discussion motivates the use of constrained algorithms to correct the initial evolution of state estimates. As illustrated by Figure 4.8 and Table 4.3, our mixed-uncertainty filters reach better precision and accuracy than CSMSE. Moreover, we guarantee that $D^e \hat{c}_k^\square = d_k^e$ for all $k \in \mathbb{Z}_+$. Among the proposed algorithms, LTVMF returns the most precise results (graph (a)). It occurs because the state nonnegativity constraints are optimally incorporated in the formulated quadratic programs. Then, the LTVMF is motivated for LTV descriptor systems whose dynamics satisfy convex polyhedral constraints and linear invariants, implying better precision and accuracy. In turn, LTVMF-E yields larger solutions (graph (b)) since the intersection between CZs (executed in the feasibility step) here attributes few correction of trajectory. That is, the improvement of precision may be deteriorated due to more order reductions and the modification of the reachable set (similar to introducing transient after each intersection). However, Table 4.3 highlights computational benefits with LTVMF-E. This algorithm is motivated for LTV descriptor systems whose main requirement is the reduction of computational cost.

We also highlight the relevance of LTIMF for the present case study. Table 4.3 shows that both LTIMF and LTIMF-P demand the smallest T^{CPU} and result in a competitive accuracy with LTVMF. These advantages motivate the use of LTIMF in LTI descriptor systems. Analogously to LTVMF-E, the LTIMF-P here brings up few improvement of precision for LTIMF at the cost of larger T^{CPU} . Conversely, LTIMF-P is motivated in the presence of polytopic constraints to refine the confidence sets. Remember that numerical corrections were applied to the quadratic programs of LTVMF, therefore there exists a loss of precision and accuracy (including LTVMF-E).

4.4 Final Remarks

In this chapter, our contributions for linear systems were presented. The main motivation was to extend algorithms existing in the literature, in order to mitigate some limitations noted over the surveys. We have introduced two novelties, namely: (i) analysis of solutions on the Pareto-optimal front; and (ii) mixed-uncertainty filtering for constrained systems. The first novelty provides a conceptual interpretation to scale biobjective functions of mixed-uncertainty algorithms. The second novelty aims at introducing inequality

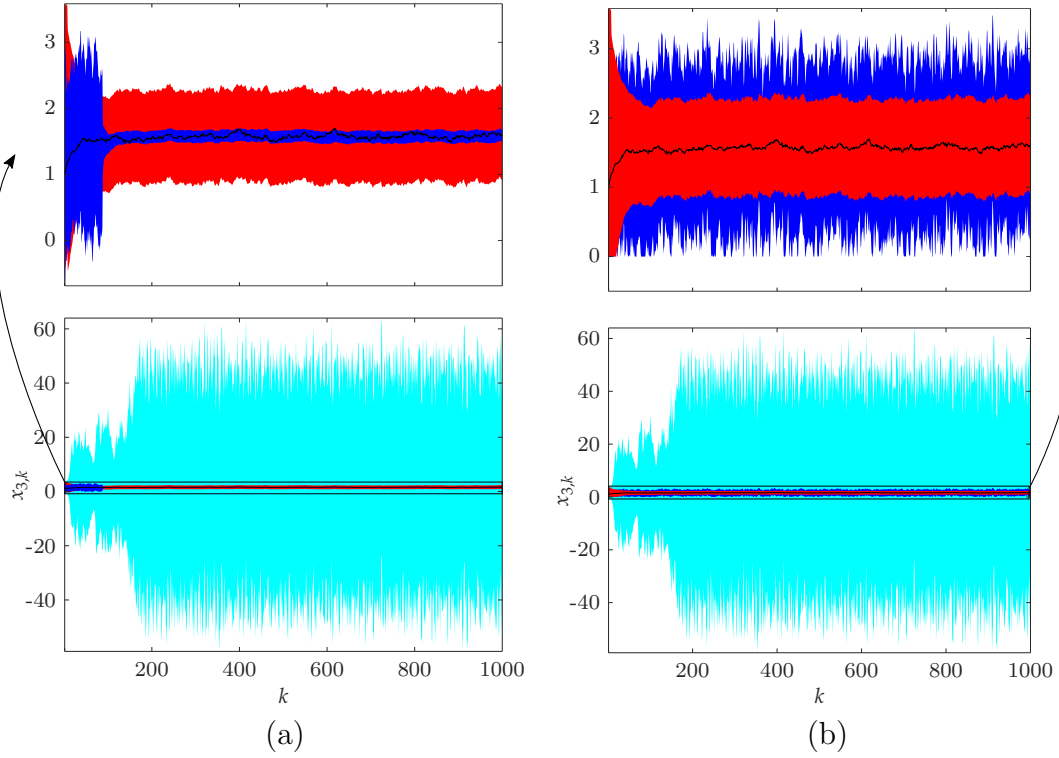


Figure 4.8: Time results of state estimation for the compartmental system regarding one separate simulation. (a) and (b) depict the same true unmeasured state $x_{3,k}$ (black solid line) involved by different confidence boxes. These estimates are computed by CSMSE (cyan color), LTVMF (blue color), and LTIMF (red color). In (a), halfspace constraints $x_{i,k} \geq 0$, $i = 1, 2, 3$ are used to execute LTVMF with its quadratic version. In (b), polytopic constraints $0 \leq x_{i,k} \leq 5$ are incorporated by LTVMF (through its explicit version) and LTIMF.

and/or equality constraints on the state vector of motivational algorithms (DGF, DTZF, DH₂GF, DH_∞ZF) to improve the precision since this information is transmitted to the design matrices, and thereby, to the state estimates. These issues were illustrated in Subsection 4.3.3 by comparing our constrained algorithms (LTIMF and LTVMF) with an unconstrained algorithm (CSMSE (Pan et al., 2022)).

5

Novel Mixed-Uncertainty Approaches for Nonlinear Systems

In this chapter, new nonlinear mixed-uncertainty state estimators are proposed. These algorithms exploit strengths of the linear filters presented in Chapters 3 and 4 to extend nonlinear estimators revisited in Appendices A-C. The main characteristics of each algorithm are highlighted in Table 5.1. Numerical examples are provided to exemplify the relevance of our mixed-uncertainty algorithms in comparison with predictors and separate approaches.

Table 5.1: Overview of the nonlinear mixed-uncertainty algorithms here proposed. The numeration of each algorithm is presented between brackets in the first column, with our proposals being differed from the existing approaches (just above) through boldface. The remaining columns characterize the algorithms regarding: uncertainty — Gaussian random vector (GRV), zonotope (Zon), constrained zonotope (CZ) —, approximation — analytical linearization (AL), first-order extension (FO), difference of convex functions (DC) programming, quasi-linear parameter varying (quasi-LPV) —, category — predictor (P), filter (F) —, constraints — equality constraints (EC), inequality constraints (IC) —, and Pareto analysis (★). The proposed state estimators are formulated under state-space equations and minimum-trace formulations.

Algorithm	Uncertainty	Approx.	Cat.	Constraints	Pareto
EZGKP [C.2]	Zon, GRV	quasi-LPV	P		
EZGKF [5.1]	Zon, GRV	quasi-LPV	F		
ZEKP [B.6]	Zon	AL	P		
FOMF [5.2]	CZ, GRV	FO	F	EC, IC	★
DCMF [5.2]	CZ, GRV	DC	F	EC, IC	★

5.1 Extended Zonotopic and Gaussian Kalman Filter

Here, we propose the extended zonotopic and Gaussian Kalman filter for nonlinear systems (EZGKF). This algorithm extends the ZGKF (Algorithm 4.1) to work on nonlinear systems, and extends the EZGKP (Section C.2) to its filter version. Therefore, we expect better precision and accuracy with respect to the EZGKP. The EZGKF is presented in de Paula et al. (2022a).

Consider the discrete-time nonlinear dynamical system in (1.1)-(1.2). Assume that the noise terms are affine, that is,

$$x_k = f(x_{k-1}, u_{k-1}) + w_{k-1}, \quad (5.1)$$

$$y_k = h(x_k) + v_k, \quad (5.2)$$

where $f : \mathbb{R}^n \times \mathbb{R}^p \rightarrow \mathbb{R}^n$ and $h : \mathbb{R}^n \rightarrow \mathbb{R}^m$ are known models, $u_{k-1} \in \mathbb{R}^p$ is the deterministic input vector, $y_k \in \mathbb{R}^m$ is the measured output vector, $w_{k-1} \in \mathbb{R}^n$ is the process noise, $v_k \in \mathbb{R}^m$ is the measurement noise, and $x_k \in \mathbb{R}^n$ is the state vector to be estimated over $k \in \mathbb{Z}_+$.

Motivated by Combastel (2016), we make the following assumptions.

Assumption 5.1.1. The initial state x_0 and the noises w_{k-1} and v_k can be decomposed as $x_0 = c_0^x + z_0^x + g_0^x$, $w_{k-1} = z_{k-1}^w + g_{k-1}^w$, and $v_k = z_k^v + g_k^v$. The terms z_0^x , z_{k-1}^w , and z_k^v are obtained from the known zero-center zonotopes $\hat{X}_0$, $\mathcal{W}_{k-1}$, and $\mathcal{V}_k$, respectively, with generator matrices $\hat{G}_0^x \in \mathbb{R}^{n \times n_g^x}$, $G_{k-1}^w \in \mathbb{R}^{n \times n_g^w}$, and $G_k^v \in \mathbb{R}^{m \times n_g^v}$, such that $G_k^v (G_k^v)^T > 0_{m \times m}$. The realizations g_0^x , g_{k-1}^w , and g_k^v are obtained from random vectors, which are approximated by the known, uncorrelated, zero-mean GRVs $\hat{X}_0$, W_{k-1} , and V_k , respectively, with covariance $\hat{P}_0^{xx} > 0_{n \times n}$,

$Q_{k-1} \geq 0_{n \times n}$, and $R_k > 0_{m \times m}$. Each GRV is also uncorrelated to itself for different time instants.

Assumption 5.1.2. The equations (5.1) and (5.2) can be transformed into a quasi-LPV system as follows:

$$x_k = A_{k-1}x_{k-1} + B_{k-1}u_{k-1} + \tilde{w}_{k-1}, \quad (5.3)$$

$$y_k = C_k x_k + \tilde{v}_k, \quad (5.4)$$

where $A_{k-1} \in \mathbb{R}^{n \times n}$, $B_{k-1} \in \mathbb{R}^{n \times p}$, and $C_k \in \mathbb{R}^{m \times n}$ are the system matrices, and $\tilde{w}_{k-1} = z_{k-1}^{\tilde{w}} + g_{k-1}^{\tilde{w}} \in \mathbb{R}^n$ and $\tilde{v}_k = z_k^{\tilde{v}} + g_k^{\tilde{v}} \in \mathbb{R}^m$ are the noises w_{k-1} and v_k augmented to incorporate the linearization errors according to the set-bounded and stochastic remainder. The matrices B_{k-1} and C_k are deterministic, whereas $A_{k-1} = A_{0,k-1} + \Delta_{k-1}$ is determined by the quasi-LPV approach using the states x_{k-1} and possible exogenous signals, with $\Delta_{k-1} = \sum_{i=1}^{n_\delta} A_{i,k-1} \delta_{i,k-1}$ being the approximated uncertainty around $A_{0,k-1}$ and $\delta_{i,k-1} \in [-1, 1]$ the scheduling parameter. The matrices $A_{0,k-1}, A_{1,k-1}, \dots, A_{n_\delta,k-1}$ are obtained from A_{k-1} .

Consider the quasi-LPV model described in (5.3)-(5.4). If the matrix A_{k-1} was known, the optimal estimator design for Problem 1.2.4 would be reached by defining the mixed state estimates $\hat{c}_k^x$, $\hat{\lambda}_k$, and $\hat{X}_k$ as in Section 4.1. However, since A_{k-1} depends on the unknown scheduling parameter δ_{k-1} , the optimal design requires modifications to yield explicit solutions. We here desire to keep both the gain matrix $K_k \in \mathbb{R}^{n \times m}$ and the center estimate $\hat{c}_k^x \in \mathbb{R}^n$ without altering the solution enclosure. Therefore, we propose the following suboptimal estimator:

$$\hat{c}_k^x = (I_n - K_k C_k) \hat{c}_{k|k-1}^x + K_k y_k, \quad (5.5)$$

$$\hat{z}_k^x = (I_n - K_k C_k) \hat{z}_{k|k-1}^x - K_k z_k^{\tilde{v}}, \quad (5.6)$$

$$\hat{g}_k^x = (I_n - K_k C_k) \hat{g}_{k|k-1}^x - K_k g_k^{\tilde{v}}, \quad (5.7)$$

where

$$\hat{c}_{k|k-1}^x \triangleq A_{0,k-1} \hat{c}_{k-1}^x + B_{k-1} u_{k-1}, \quad (5.8)$$

$$\hat{z}_{k|k-1}^x \triangleq A_{k-1} \hat{z}_{k-1}^x + \Delta_{k-1} \hat{c}_{k-1}^x + z_{k-1}^{\tilde{w}}, \quad (5.9)$$

$$\hat{g}_{k|k-1}^x \triangleq A_{k-1} \hat{g}_{k-1}^x + g_{k-1}^{\tilde{w}}. \quad (5.10)$$

Our main task is to determine the forecast estimates $\hat{\lambda}_{k|k-1}$ and $\hat{X}_{k|k-1}$ that enclose the worst case related to $\hat{z}_{k|k-1}^x$ and $\hat{g}_{k|k-1}^x$, respectively. These results are addressed next by two propositions.

Proposition 5.1.1. Let $\hat{\lambda}_{k-1} = \{\hat{c}_{k-1}^x, 0_{n \times 1}\} \ni \hat{z}_{k-1}^x$, $\tilde{\mathcal{W}}_{k-1} = \{G_{k-1}^{\tilde{w}}, 0_{n \times 1}\} \ni z_{k-1}^{\tilde{w}}$, and $\mathcal{B}^{n_\delta} \ni \delta_{k-1}$ be the known sets. Then, the elements $\hat{z}_{k|k-1}^x$ of (5.9) are enclosed by the zonotope $\hat{\lambda}_{k|k-1} =$

$\{\hat{G}_{k|k-1}^x, 0_{n \times 1}\}$ with generator matrix

$$\hat{G}_{k|k-1}^x = \begin{bmatrix} A_{0,k-1} \hat{G}_{k-1}^x & G_{k-1}^{\tilde{w}} & A_{1,k-1} \begin{bmatrix} \hat{c}_{k-1}^x & \hat{G}_{k-1}^x \end{bmatrix} & \cdots & A_{n_\delta, k-1} \begin{bmatrix} \hat{c}_{k-1}^x & \hat{G}_{k-1}^x \end{bmatrix} \end{bmatrix}. \quad (5.11)$$

Proof. By making explicit the matrices $A_{0,k-1}$ and $A_{i,k-1}$ in (5.9), we obtain

$$\hat{z}_{k|k-1}^x = A_{0,k-1} \hat{z}_{k-1}^x + z_{k-1}^{\tilde{w}} + \sum_{i=1}^{n_\delta} A_{i,k-1} (\hat{c}_{k-1}^x + \hat{z}_{k-1}^x) \delta_{i,k-1}.$$

Replacing the elements $\hat{z}_{k-1}^x$, $z_{k-1}^{\tilde{w}}$, and $\delta_{i,k-1}$ by their corresponding set yields

$$\hat{\mathcal{X}}_{k|k-1} = A_{0,k-1} \hat{\mathcal{X}}_{k-1} \oplus \tilde{\mathcal{W}}_{k-1} \oplus A_{1,k-1} (\hat{c}_{k-1}^x \oplus \hat{\mathcal{X}}_{k-1})[-1, 1] \oplus \cdots \oplus A_{n_\delta, k-1} (\hat{c}_{k-1}^x \oplus \hat{\mathcal{X}}_{k-1})[-1, 1].$$

Since $\hat{c}_{k-1}^x[-1, 1]$ contributes with a new generator and $\mathcal{B}^{n_g}[-1, 1] = \mathcal{B}^{n_g}$, with n_g being the order of $\hat{\mathcal{X}}_{k-1}$, Properties 2.2.2 and 2.2.3 imply the zero-center zonotope $\hat{\mathcal{X}}_{k|k-1}$ with generator matrix given by (5.11). $\blacksquare$

Proposition 5.1.2. Let the GRVs $\hat{X}_{k-1} \sim \mathcal{N}(0_{n \times 1}, \hat{P}_{k-1}^{xx})$ and $\tilde{W}_{k-1} \sim \mathcal{N}(0_{n \times 1}, \tilde{Q}_{k-1})$ be known representations for the uncorrelated realizations $\hat{g}_{k-1}^x$ and $g_{k-1}^{\tilde{w}}$, respectively, and $\mathcal{B}^{n_\delta} \ni \delta_{k-1}$. Then, the realizations $\hat{g}_{k|k-1}^x$ of (5.10) are represented by the GRV $\hat{X}_{k|k-1} \sim \mathcal{N}(0_{n \times 1}, \hat{P}_{k|k-1}^{xx})$ with covariance matrix

$$\hat{P}_{k|k-1}^{xx} = A_{0,k-1} \hat{P}_{k-1}^{xx} A_{0,k-1}^\top + \tilde{Q}_{k-1} + \Omega_{ii} + \Omega_{ij}, \quad (5.12)$$

where

$$\begin{aligned} \Omega_{ii} &\triangleq \sum_{i=1}^{n_\delta} \left(A_{0,k-1} \hat{P}_{k-1}^{xx} A_{i,k-1}^\top + A_{i,k-1} \hat{P}_{k-1}^{xx} A_{0,k-1}^\top + A_{i,k-1} \hat{P}_{k-1}^{xx} A_{i,k-1}^\top \right), \\ \Omega_{ij} &\triangleq \sum_{i=1}^{n_\delta-1} \sum_{j=i+1}^{n_\delta} \left(A_{i,k-1} \hat{P}_{k-1}^{xx} A_{j,k-1}^\top + A_{j,k-1} \hat{P}_{k-1}^{xx} A_{i,k-1}^\top \right). \end{aligned}$$

Proof. Since the realizations $\hat{g}_{k-1}^x$ and $g_{k-1}^{\tilde{w}}$ are uncorrelated in (5.10), Properties 2.1.1 and 2.1.2 leads to the GRV $X_{k|k-1} \sim \mathcal{N}(0_{n \times 1}, P_{k|k-1}^{xx})$ with covariance

$$P_{k|k-1}^{xx} = A_{k-1} \hat{P}_{k-1}^{xx} A_{k-1}^\top + \tilde{Q}_{k-1}.$$

However, the matrix A_{k-1} is unknown, and we are interested in reaching a covariance matrix such that $\hat{P}_{k|k-1}^{xx} \geq P_{k|k-1}^{xx}$. To achieve that, we first make explicit the matrices $A_{0,k-1}$ and $A_{i,k-1}$ in $P_{k|k-1}^{xx}$, yielding

$$\begin{aligned} P_{k|k-1}^{xx} &= A_{0,k-1} \hat{P}_{k-1}^{xx} A_{0,k-1}^\top + \tilde{Q}_{k-1} + \sum_{j=1}^{n_\delta} \delta_{j,k-1} A_{0,k-1} \hat{P}_{k-1}^{xx} A_{j,k-1}^\top \\ &+ \sum_{i=1}^{n_\delta} \delta_{i,k-1} A_{i,k-1} \hat{P}_{k-1}^{xx} A_{0,k-1}^\top + \sum_{i=1}^{n_\delta} \sum_{j=1}^{n_\delta} \delta_{i,k-1} \delta_{j,k-1} A_{i,k-1} \hat{P}_{k-1}^{xx} A_{j,k-1}^\top. \end{aligned}$$

As the summed matrices are square, the latter summation can be split up as

$$\sum_{i=1}^{n_\delta} \sum_{j=1}^{n_\delta} \delta_{i,k-1} \delta_{j,k-1} A_{i,k-1} \hat{P}_{k-1}^{\text{xx}} A_{j,k-1}^\top = \sum_{i=1}^{n_\delta} \delta_{i,k-1}^2 A_{i,k-1} \hat{P}_{k-1}^{\text{xx}} A_{i,k-1}^\top + \sum_{i < j} \delta_{i,k-1} \delta_{j,k-1} (A_{i,k-1} \hat{P}_{k-1}^{\text{xx}} A_{j,k-1}^\top + A_{j,k-1} \hat{P}_{k-1}^{\text{xx}} A_{i,k-1}^\top).$$

After gathering the single summations in a unique sum, and thereby, applying Theorem 2.1.1, we obtain the upper bounded covariance matrix (5.12), finishing the proof. ■

Remark 5.1.1. Proposition 5.1.1 could be formulated with Proposition 2.2.2 by assuming an interval matrix $[A_{k-1}]$ instead of an affine matrix given by $A_{0,k-1}, \dots, A_{n_\delta,k-1}$. In this case, we first manipulate the original zonotopic term

$$z_{k|k-1}^x = A_{k-1} \hat{z}_{k-1}^x + z_{k-1}^w$$

to obtain the corresponding zonotope

$$\diamond \{[A_{k-1}] \hat{\mathcal{X}}_{k-1}\} \oplus \tilde{\mathcal{W}}_{k-1} = \{\Psi, 0_{n \times 1}\}.$$

Second, the uncertainty $\Delta_{k-1} \hat{c}_{k-1}^x$ is shifted to the prior zonotope, yielding $\hat{\mathcal{X}}_{k|k-1}$ with generator matrix

$$\hat{G}_{k|k-1}^x = [\Psi \quad \text{diag}(\text{rad}([A_{k-1}] \hat{c}_{k-1}^x))].$$

In doing so, the order of $\hat{\mathcal{X}}_{k|k-1}$ changes from $[n_g + n_g^w + n_\delta(1 + n_g)]$ to $[n_g + n_g^w + 2n]$, with n_g being the order of $\hat{\mathcal{X}}_{k-1}$.

Thanks to Propositions 5.1.1 and 5.1.2, we achieve forecast estimates with the standard definitions of zonotopes and GRVs. Therefore, the elements $\hat{z}_k^x$ in (5.6) and the realizations $\hat{g}_k^x$ in (5.7) are, respectively, characterized by the zonotope $\hat{\mathcal{X}}_k = \{\hat{G}_k^x, 0_{n \times 1}\}$ and GRV $\hat{X}_k \sim \mathcal{N}(0_{n \times 1}, \hat{P}_k^{\text{xx}})$, where

$$\hat{G}_k^x = [(\mathbf{I}_n - K_k C_k) \hat{G}_{k|k-1}^x \quad -K_k G_k^v], \quad (5.13)$$

$$\hat{P}_k^{\text{xx}} = (\mathbf{I}_n - K_k C_k) \hat{P}_{k|k-1}^{\text{xx}} (\mathbf{I}_n - K_k C_k)^\top + K_k \tilde{R}_k K_k^\top, \quad (5.14)$$

with $\hat{G}_{k|k-1}^x$ and $\hat{P}_{k|k-1}^{\text{xx}}$ being given by (5.11) and (5.12), respectively. In this case, the uncorrelation between $\hat{g}_{k|k-1}^x$ and g_k^v was exploited.

Recall that both zonotope $\hat{\mathcal{X}}_k$ and GRV $\hat{X}_k$ are obtained by assuming a known matrix K_k . In order to optimally yield this design matrix, we propose the following theorem.

Theorem 5.1.1. Let $\hat{G}_k^x$ in (5.13) and $\hat{P}_k^{\text{xx}}$ in (5.14) be the instances of the cost function J_k^{zg} in (1.9), as well as the known weight parameter $\eta \in [0, 1]$. Then, minimizing J_k^{zg} results in the optimal gain $\check{K}_k$ given by (4.15), where $\hat{G}_{k|k-1}^x$ in (5.11) and $\hat{P}_{k|k-1}^{\text{xx}}$ in (5.12) are the instances of $\hat{P}_{k|k-1}^{\text{zg}}$ in (4.18), while G_k^v and $\tilde{R}_k$ are the instances of R_k^{zg} in (4.19).

Proof. This proof is similar to the proof of Theorem 4.1.1, with the difference being that $\hat{G}_{k|k-1}^x$, $\hat{G}_k^x$, $\hat{P}_{k|k-1}^{xx}$, and $\hat{P}_k^{xx}$ are given by (5.11), (5.13), (5.12), and (5.14), respectively. ■

Recall that the number of generators of $\hat{G}_k^x$ in (5.13) is larger than the amount of generators of the initial matrix $\hat{G}_{k-1}^x$. Iteratively, this growth may imply undesired numerical issues such as the increase of both storage and processing time. In order to mitigate the computational burden, some order reduction should be applied to either $\hat{G}_{k-1}^x$ or $\hat{G}_k^x$ to fix its order in a desired value φ_g .

In Algorithm 5.1, we show how to execute one iteration of the proposed EZGKF. Since the confidence box $\mathcal{I}_k^\alpha$ given by (2.49) has specific applications, it is also addressed as output. This algorithm solves Problem 1.2.4 without enforcing state constraints.

Algorithm 5.1: $[\hat{c}_k^x, \hat{\mathcal{X}}_k, \hat{X}_k, \mathcal{I}_k^\alpha] = \text{EZGKF}(\hat{c}_{k-1}^x, \hat{\mathcal{X}}_{k-1}, \hat{X}_{k-1}, A_{0,k-1}, A_{1,k-1}, \dots, A_{n_\delta, k-1}, B_{k-1}, u_{k-1}, \tilde{W}_{k-1}, \tilde{X}_{k-1}, y_k, C_k, \tilde{V}_k, \tilde{V}_k, \alpha, \eta, \varphi_g)$

- 1 Compute $\hat{\mathcal{X}}_{k-1}^\downarrow = \text{ZR}(\hat{\mathcal{X}}_{k-1}, \varphi_g)$, where ZR is discussed in Subsection B.1.1.1
 - 2 Obtain the forecast estimates $\hat{c}_{k|k-1}^x$ in (5.8), $\hat{G}_{k|k-1}^x$ in (5.11), and $\hat{P}_{k|k-1}^{xx}$ in (5.12)
 - 3 Apply Theorem 5.1.1 to obtain the optimal gain $\check{K}_k$
 - 4 Calculate the current state estimates $\hat{c}_k^x$ in (5.5), $\hat{G}_k^x$ in (5.13), and $\hat{P}_k^{xx}$ in (5.14), yielding $\hat{\mathcal{X}}_k = \{\hat{G}_k^x, 0_{n \times 1}\}$ and $\hat{X}_k \sim \mathcal{N}(0_{n \times 1}, \hat{P}_k^{xx})$
 - 5 Compute the confidence box $\mathcal{I}_k^\alpha$ using (2.49)
-

In Table 5.2, we show the worst-case complexity order of the proposed EZGKF. By comparing the complexity order of EZGKP and EZGKF, we note that:

$$O(\text{EZGKF}) - O(\text{EZGKP}) = nm n_g^{\tilde{v}} (1 - n_\delta) + n^2 (n_g^{\tilde{w}} - n_\delta n_g^{\tilde{v}}) - nm n_g^{\tilde{w}},$$

whose negative result will denote that EZGKF is faster than EZGKP. Since $n_\delta \geq 1$, EZGKF may achieve a better computational performance than EZGKP.

Remark 5.1.2. For the cases in which $\tilde{w}_k$ and $\tilde{v}_k$ are correlated, the strategy of Pan et al. (2022) can be applied to the proposed EZGKF to remove the correlation that appears between the state error e_k and the process noise w_k . In doing so, the current orders $n_g^{\tilde{w}}$ and $n_g^{\tilde{v}}$ should be replaced by $n_{g,k-1}^{\tilde{v}}$ and $n_{g,k}^{\tilde{v}}$, respectively. This is the unique modification occurred in the total worst-case complexity order here derived for the EZGKF. Then, the derived complexity difference between EZGKP and EZGKF does not change under $n_{g,k-1}^{\tilde{v}} = n_{g,k}^{\tilde{v}} = n_g^{\tilde{v}}$.

5.1.1 Numerical Results

Next, the proposed EZGKF is experimented over two numerical examples, which contain specific discussions about the advantages of EZGKF over EZGKP. For comparison

Table 5.2: Complexity order $O(\cdot)$ of each step from EZGKF.

Step	$O(\cdot)$
1	$n_g (n + \log(n_g))$
2	$np + n_\delta n^2 \varphi_g + n_\delta^2 n^3$
3	$n^2 (n_g^{\bar{w}} + n_\delta \varphi_g + m) + m^2 (n_g^{\bar{v}} + n + m)$
4	$n^2 (n + m + n_g^{\bar{w}} + n_\delta \varphi_g) + nm (m + n_g^{\bar{v}})$
5	$n (n_g^{\bar{w}} + n_g^{\bar{v}} + n_\delta \varphi_g)$
Total	$n_g \log(n_g) + n (n_g + p + mn_g^{\bar{v}})$ $+ n^2 (m + n_\delta \varphi_g + nn_\delta^2 + n_g^{\bar{w}}) + m^2 (n + m + n_g^{\bar{v}})$

purposes, we compute three performance indexes: T^{CPU} in (3.57), RMSE in (3.58) with $\hat{c}_k^x$ being the punctual estimate, and $r^\square$ in (3.84) with $\mathcal{I}_k^a$ (given by (2.49)) being the analyzed box. The following computer configuration was used: 8 GB RAM 1333 MHz, Windows 10 Pro, and AMD FX-6300 CPU 3.50 GHz. All implementations were executed in MATLAB 9.11 (Release) with INTLAB 12 (Rump, 1999) and MPT3 (Herceg et al., 2013). Specifically, INTLAB is used to compute interval estimates while MPT is employed to sketch sets.

5.1.1.1 Van der Pol Oscillator

This subsection points out a relevant application of mixed-uncertainty algorithms, since it attributes a real meaning to represent the uncertainties based on the available knowledge about them. Specifically, the process is affected by an unknown input, whose lower and upper bounds are assumed to be known. Therefore, such input can be represented by a zonotope. The measurements are affected by a noise with statistical properties; then, we employ a GRV to characterize it.

Consider the Euler-discretized Van der Pol oscillator given by (Teixeira et al., 2008a)

$$x_k = f(x_{k-1}) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} z_{k-1}^w, \quad (5.15)$$

where

$$f(x_{k-1}) = \begin{bmatrix} x_{1,k-1} + T_s x_{2,k-1} \\ -T_s x_{1,k-1} + (T_s + 1 - T_s x_{1,k-1}^2) x_{2,k-1} \end{bmatrix},$$

with $x_k = [x_{1,k} \ x_{2,k}]^\top \in \mathbb{R}^2$, and $z_{k-1}^w \in \mathbb{R}$ being an uncertain input whose bounds are known. The measurement model is given by

$$y_k = \begin{bmatrix} 1 & 1 \end{bmatrix} x_k + g_k^v, \quad (5.16)$$

with $g_k^v \in \mathbb{R}$ being a noise term whose values are described by the GRV $V \sim \mathcal{N}(0, 0.04)$. To

Table 5.3: Results of T^{CPU} (including the computation of matrices $A_{0,k-1}$ and $A_{1,k-1}$), RMSE, and $r^\square$ for the Van der Pol oscillator. The percentage reduction in comparison with EZGKP is indicated between parenthesis.

Algorithms	T^{CPU}	RMSE ₁	RMSE ₂	$r^\square$
EZGKP	3.1 ms	0.047	0.173	0.733
EZGKF	3.1 ms	0.045 (↓4%)	0.130 (↓25%)	0.611 (↓17%)

simulate the system, we set $x_0 = 1_{2 \times 1}$, $T_s = 0.1$ s, $k_f = 500$, $m_s = 100$, and

$$z_k^w = \begin{cases} T_s \sin(2kT_s), & \text{if } kT_s < 10 \text{ s or } kT_s \geq 30 \text{ s}, \\ T_s (\sin(2kT_s) + 0.5), & \text{if } 10 \text{ s} \leq kT_s < 20 \text{ s}, \\ T_s (\sin(2kT_s) - 0.5), & \text{if } 20 \text{ s} \leq kT_s < 30 \text{ s}. \end{cases}$$

To estimate states, we set $\hat{c}_0^x = 0_{2 \times 1}$, $\hat{X}_0 = \{0.5\mathbf{I}_2, 0_{2 \times 1}\}$, $\hat{X}_0 \sim \mathcal{N}(0_{2 \times 1}, \frac{1}{4c_x}\mathbf{I}_2)$, $c_x = 11.829$, $\mathcal{W} = \left\{ \begin{bmatrix} 0 & 0 \\ 1.5T_s & 0 \end{bmatrix}, 0_{2 \times 1} \right\}$, $\varphi_s = 100$, $k_0 = 20$, $\eta = 0.5$, and $\alpha = 0.0027$ such that $x_0 \in \mathcal{I}_0^\alpha$. To obtain $f(x_{k-1}) = A_{k-1}x_{k-1}$, we choose

$$A_{k-1} = \begin{bmatrix} 1 & T_s \\ -T_s & (T_s + 1 - T_s z_{k-1}) \end{bmatrix},$$

where $z_{k-1} = x_{1,k-1}^2$. By defining the confidence interval $[z_{k-1}] = (\mathcal{I}_{1,k-1}^\alpha)^2$ with $\alpha = 0.0027$, the partial matrices $A_{0,k-1}$ and $A_{1,k-1}$ are finally given by

$$A_{0,k-1} = \begin{bmatrix} 1 & T_s \\ -T_s & T_s + 1 - T_s \text{mid}([z_{k-1}]) \end{bmatrix}, \quad A_{1,k-1} = \begin{bmatrix} 0 & 0 \\ 0 & -T_s \text{rad}([z_{k-1}]) \end{bmatrix}.$$

For comparison purposes, Figure 5.1 already illustrates an enhancement of both accuracy and precision when employing EZGKF rather than EZGKP. Then, we expect to attain better centers and confidence domains with EZGKF. However, since those algorithms yield suboptimal solutions, EZGKP can eventually return better results, as shown by graph (b).

Table 5.3 shows that no significant difference of T^{CPU} is noted between the algorithms. This issue accords with the order comparison of the EZGKP and EZGKF algorithms, which points out zero difference for this example. The reached value of T^{CPU} is also much smaller than the sampling time, enabling the direct use of such algorithms in real-time applications. Advantageously, EZGKF returns the smallest RMSE and $r^\square$, implying centers closer to true states and less conservative confidence regions.

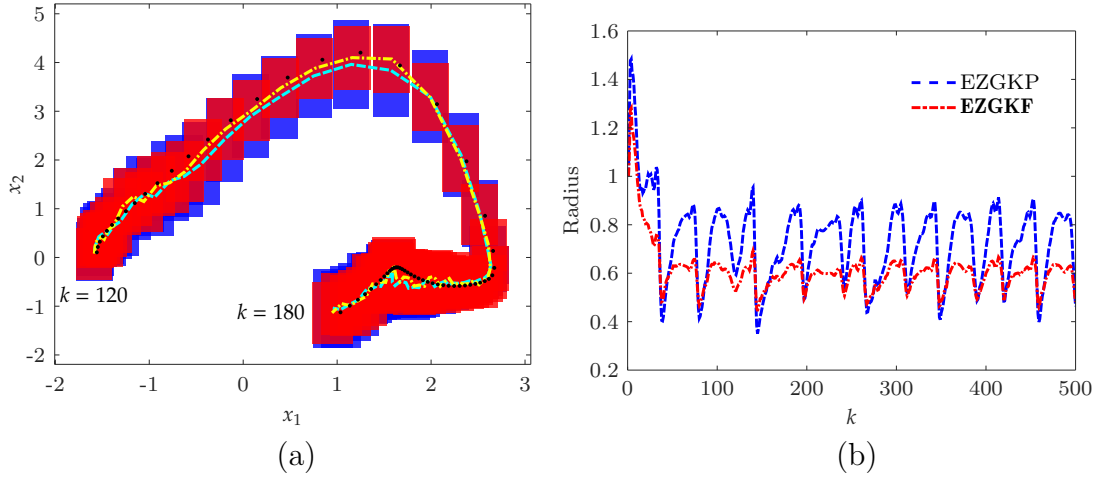


Figure 5.1: State estimation for the Van der Pol oscillator. In (a), the true states (black $\cdot$) are approximately involved by blue (EZGKP) and red (EZGKF) confidence boxes, and punctually estimated by cyan (EZGKP) and yellow (EZGKF) centers for the discrete time horizon $k \in \{120, \dots, 180\}$. The largest radius of the confidence box of each algorithm is sketched in (b).

5.1.1.2 LPV Application

Now, we illustrate a case in which Remark 5.1.1 can be interesting to compose the EZGKF; this modification is here called EZGKF-M. Based on Alamo et al. (2005), consider the discrete-time linear dynamical system given by

$$x_k = \begin{bmatrix} 0 & -0.5 \\ 1 & 1 + \delta_{k-1} \end{bmatrix} x_{k-1} + w_{k-1}, \quad (5.17)$$

$$y_k = \begin{bmatrix} -2 & 1 \end{bmatrix} x_k + v_k, \quad (5.18)$$

where $\delta_{k-1} \in [-0.3, 0.3]$. We consider that $w_{k-1} = z_{k-1}^w + g_{k-1}^w$ and $v_k = z_k^v + g_k^v$, whose characterizations are given by $\mathcal{W} = \{0.5L, 0_{2 \times 1}\}$, with $L = 0.02 \begin{bmatrix} -6 & 0 \\ 1 & 0 \end{bmatrix}$, $W \sim \mathcal{N}(0_{2 \times 1}, \frac{1}{36}LL^\top)$, $\mathcal{V} = \{0.1, 0\}$, and $V \sim \mathcal{N}(0, \frac{1}{9}0.01)$.

The system is simulated with $x_0 = [0.1 \ 0.1]^\top$, $k_f = 40$, and $m_s = 100$. The terms g_{k-1}^w and g_k^v are taken from GRVs, while the terms z_{k-1}^w and z_k^v are taken from uniform distributions. To estimate the states, we additionally set the following parameters: $\hat{c}_0^x = 0_{2 \times 1}$, $\hat{G}_0^x = 0.1I_2$, $\hat{P}_0^{xx} = \frac{0.04}{47.316}I_2$, $\varphi_g = 4000$, $k_0 = 4$, $\eta = 0.5$, $\alpha = 0.0027$, and matrices $A_{0,k-1} = \begin{bmatrix} 0 & -0.5 \\ 1 & 1 \end{bmatrix}$ and $A_{1,k-1} = \begin{bmatrix} 0 & 0 \\ 0 & 0.3 \end{bmatrix}$.

Figure 5.2 illustrates that a significant improvement of precision is promoted by EZGKF-M in comparison with EZGKP. Remark 5.1.1 is here a good option because the state matrix A_{k-1} is really interval; unlike Subsection 5.1.1.1 where the state matrix

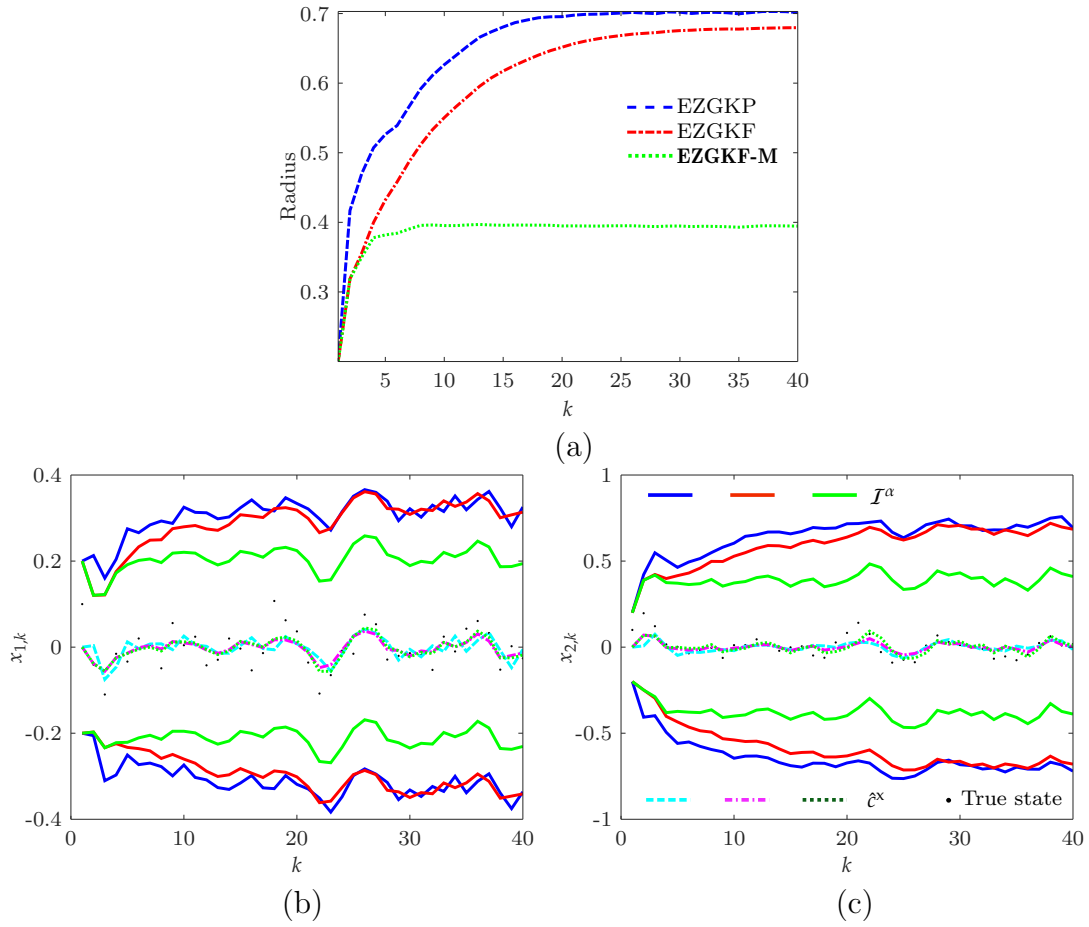


Figure 5.2: State estimation for an LPV system. Graph (a): time evolution of the largest radius of confidence boxes computed by EZGKP (blue color), EZGKF (red color), and EZGKF-M (green color). Graphs (b) and (c) show confidence boxes and centers computed by EZGKP, EZGKF, and EZGKF-M that enclose/approximate the true states.

Table 5.4: Results of T^{CPU} , RMSE, and $r^\square$ for an LPV system. The percentage reduction/increase in comparison with EZGKP is indicated between parenthesis.

Algorithms	T^{CPU}	RMSE ₁	RMSE ₂	$r^\square$
EZGKP	1.6 ms	0.0437	0.0513	0.646
EZGKF	1.6 ms	0.0348 (↓20%)	0.0449 (↓12%)	0.600 (↓7%)
EZGKF-M	1.2 ms (↓25%)	0.0328 (↓25%)	0.0419 (↓18%)	0.386 (↓40%)

is affine. Besides corroborating the prior enhancement of precision, Table 5.4 shows that EZGKF-M yields smaller RMSE and T^{CPU} . For $\varphi_g > 4000$, the difference of T^{CPU} becomes more significant.

Table 5.5: Complexity order $O(\cdot)$, and number of constraints $\tilde{n}_h$ and generators $\tilde{n}_g$ for CZs. The results of the last column are related to Proposition 2.3.2.

CZ	$\mathcal{X}^\alpha$	$\mathcal{I}^\alpha$	$\mathcal{R}$ (Rego et al., 2020a, Theorem 3)	$\mathcal{R}$ (Lemma 2.3.1)
$\tilde{n}_h$	$\tilde{n}_h = n_h$	$\tilde{n}_h = 0$	$\tilde{n}_h = 0.5n_h^2 + 1.5n_h$	$\tilde{n}_h = 0$
$\tilde{n}_g$	$\tilde{n}_g = n_g + n$	$\tilde{n}_g = n$	$\tilde{n}_g = 0.5(n_g + n)^2 + 1.5(n_g + n) + 2m$	$\tilde{n}_g = m$
$O(\cdot)$	n	$mn_g(n_h + n_g)^3$	$m(n^2(n_g + n) + n(n_g + n)^2) + n_h(n_h + n_g + n)^3 + mn_g(n_h + n_g)^3$	$n_h(n_g + n_h)^3 + n(n_g + n)(n_h + n_g + n)^3 + (n^2 + m)2^n$

5.2 Nonlinear Mixed-Uncertainty State Estimators based on CZs

In this section, we propose two mixed-uncertainty filters for constrained nonlinear dynamical systems based on GRVs and CZs. In the authors' best knowledge, this is the first work covering CZs and state linear inequality and nonlinear equality constraints for mixed-uncertainty nonlinear cases. In order to confer robustness to the executed linear approximations, we apply the first-order Taylor extension (Rego et al., 2020a) and DC programming (Lemma 2.3.1), yielding the algorithms FOMF and DCMF, respectively. Conversely, in the presence of stochastic information, the robustness of each technique relies on the preset confidence level such that our algorithms yield approximate solutions. Our main contribution is the proposal of quadratic and explicit versions to run FOMF and DCMF under the minimum-variance criterion. However, our cost normalization enables the use of other performance criteria to develop different optimization problems. Both FOMF and DCMF compose the paper de Paula et al. (2022a).

Since guaranteed methods are here employed to linearize models, the confidence box $\mathcal{I}^\alpha$, given by (2.49), must coincide with the interval hull of the confidence CZ $\mathcal{X}_k^\alpha$, that is, $\mathcal{I}^\alpha = \square\mathcal{X}^\alpha$. To achieve that, we employ Definition 2.4.3 with the interval hull of the confidence ellipsoid ($\square\mathcal{E}$) instead of the parallelotopic outer approximation ($\mathcal{P}$). Let the mixed-uncertainty vector $x \in \mathbb{R}^n$ be represented by the mixed center $c^x \in \mathbb{R}^n$, the zero-center CZ $\mathcal{X} \subset \mathbb{R}^n$ with n_h constraints and n_g generators, and the zero-mean GRV X with $P^{xx} \in \mathbb{R}^{n \times n}$. Let also $\varrho: \mathbb{R}^n \rightarrow \mathbb{R}^m$ be the nonlinear transformation of the mixed-uncertainty vector x , whose approximation error is enclosed by $\mathcal{R} \subset \mathbb{R}^m$ using the confidence sets $\mathcal{X}^\alpha$ and $\mathcal{I}^\alpha$. Thereby, Table 5.5 shows the number of constraints and generators of the involved CZs together with their worst-case complexity order.

Consider the discrete-time nonlinear dynamical system given by (1.1)-(1.4), where the terms $w_{k-1} = z_{k-1}^w + g_{k-1}^w \in \mathbb{R}^q$ and $v_k = z_k^v + g_k^v \in \mathbb{R}^r$ correspond to the process and measurement noises, respectively. These noises are resulting from the combination of a set-bounded parcel with a random parcel whose mean and covariance are known. Then, both w_{k-1} and v_k are called mixed-uncertainty vectors. By assuming the initial state $x_0 = \hat{c}_0^x + \hat{z}_0^x + \hat{g}_0^x$, with $\hat{c}_0^x$ being its center estimate, the state vector x_k also becomes mixed. Therefore, the terms z_{k-1}^w , z_k^v , and $\hat{z}_0^x$ are characterized by the known zero-center CZs $\mathcal{W}_{k-1} = \{G_{k-1}^w, 0_{q \times 1}, A_{k-1}^w, b_{k-1}^w\}$,

$\mathcal{V}_k = \{G_k^v, 0_{r \times 1}, A_k^v, b_k^v\}$, and $\hat{\mathcal{X}}_0 = \{\hat{G}_0^x, 0_{n \times 1}, \hat{A}_0^x, \hat{b}_0^x\}$, respectively, with $G_k^v (G_k^v)^\top > 0_{r \times r}$. In turn, the terms g_{k-1}^w , g_k^v , and $\hat{g}_0^x$ are characterized by the known zero-mean GRVs W_{k-1} , V_k , and $\hat{X}_0$ with covariance $Q_{k-1} \geq 0_{q \times q}$, $R_k > 0_{r \times r}$, and $\hat{P}_0^{xx} > 0_{n \times n}$. We assume that W_{k-1} and V_k define independent stochastic processes, in which g_{k-1}^w , g_k^v , and $\hat{g}_0^x$ are uncorrelated.

The mixed-uncertainty filters here proposed start with the analytical linearization of nonlinear models. This procedure reminds the approximations performed in Särkkä (2013) and Wang & Puig (2016). However, we introduce the computation of linearization enclosure and work on mixed-uncertainty vectors.

The first linearization occurs with the function f (1.1), resulting in

$$x_k = f(\bar{x}_{k-1}, u_{k-1}, \bar{w}_{k-1}) + F^x(x_{k-1} - \bar{x}_{k-1}) + F^w(w_{k-1} - \bar{w}_{k-1}) + e_{k-1}^f, \quad (5.19)$$

where $F^x = \nabla_x f(\bar{x}_{k-1}, u_{k-1}, \bar{w}_{k-1})$ and $F^w = \nabla_w f(\bar{x}_{k-1}, u_{k-1}, \bar{w}_{k-1})$ are Jacobian matrices evaluated at the points

$$\bar{x}_{k-1} \in \mathcal{I}_{k-1}^{\alpha, x} = \hat{\mathcal{C}}_{k-1}^x \oplus \square \hat{\mathcal{X}}_{k-1} \oplus \square \mathcal{E}_{k-1}^x, \quad (5.20)$$

$$\bar{w}_{k-1} \in \mathcal{I}_{k-1}^{\alpha, w} = 0_{q \times 1} \oplus \square \mathcal{W}_{k-1} \oplus \square \mathcal{E}_{k-1}^w, \quad (5.21)$$

and with $\hat{\mathcal{C}}_{k-1}^x$ being the state estimate at $k-1$. Note that the linearization with respect to the singleton u_{k-1} is zero since $\bar{u}_{k-1} = u_{k-1}$. In principle, $\bar{x}_{k-1}$ and $\bar{w}_{k-1}$ can assume any value inside their confidence set. The procedure here adopted to specific those punctual estimates is presented as follows:

$$\bar{x}_{k-1} = \arg \min_{x \in \mathcal{I}_{k-1}^{\alpha, x}} \|x - \hat{\mathcal{C}}_{k-1}^x\|_1, \quad \bar{w}_{k-1} = \arg \min_{w \in \mathcal{I}_{k-1}^{\alpha, w}} \|w\|_1. \quad (5.22)$$

The term $e_{k-1}^f \in \mathbb{R}^n$ is the linearization error. It is here approximated by a CZ $\mathcal{R}_{k-1}^{ef} \subset \mathbb{R}^n$ using the guaranteed approaches Rego et al. (2020a, Theorem 3) and Lemma 2.3.1 separately. To use these methods, we provide the confidence sets $\mathcal{X}_{k-1}^{\alpha, x}$, $\mathcal{X}_{k-1}^{\alpha, w}$, $\mathcal{I}_{k-1}^{\alpha, x}$, and $\mathcal{I}_{k-1}^{\alpha, w}$. In doing so, the desired set $\mathcal{R}_{k-1}^{ef}$ is obtained as shown in Table 5.5. The robustness of those techniques relies on the significance level $\alpha \in [0, 1]$ established to bound the GRVs. Therefore, we present the evolution of mixed-uncertainty vectors over f (1.1).

Proposition 5.2.1 (Forecast). Consider the linearized process model in (5.19). Let $x_{k-1} = \hat{\mathcal{C}}_{k-1}^x + \hat{z}_{k-1}^x + \hat{g}_{k-1}^x$ and $w_{k-1} = z_{k-1}^w + g_{k-1}^w$ be mixed uncertainties with known representations, where $\hat{g}_{k-1}^x$ and g_{k-1}^w are uncorrelated for all $k \geq 1$. Let also $\mathcal{R}_{k-1}^{ef} = \{G_{k-1}^{ef}, \hat{e}_{k-1}^f\} \ni e_{k-1}^f$ be the known linearization enclosure. Then, the *a priori* mixed state vector $x_{k|k-1} = \hat{\mathcal{C}}_{k|k-1}^x + \hat{z}_{k|k-1}^x + \hat{g}_{k|k-1}^x$ is characterized by

$$\begin{aligned} \hat{\mathcal{C}}_{k|k-1}^x &= f(\bar{x}_{k-1}, u_{k-1}, \bar{w}_{k-1}) + F^x(\hat{\mathcal{C}}_{k-1}^x - \bar{x}_{k-1}) - F^w \bar{w}_{k-1} + \hat{e}_{k-1}^f, \\ \hat{\mathcal{X}}_{k|k-1} &= F^x \hat{\mathcal{X}}_{k-1} \oplus F^w \mathcal{W}_{k-1} \oplus (\mathcal{R}_{k-1}^{ef} - \hat{e}_{k-1}^f) \end{aligned} \quad (5.23)$$

$$= \left\{ \hat{G}_{k|k-1}^x, 0_{n \times 1}, \hat{A}_{k|k-1}^x, \hat{b}_{k|k-1}^x \right\}, \quad (5.24)$$

$$\hat{X}_{k|k-1} = F^x \hat{X}_{k-1} + F^w W_{k-1} \sim \mathcal{N}(0_{n \times 1}, \hat{P}_{k|k-1}^{xx}), \quad (5.25)$$

where

$$\hat{G}_{k|k-1}^x = \begin{bmatrix} F^x \hat{G}_{k-1}^x & F^w G_{k-1}^w & G_{k-1}^{ef} \end{bmatrix}, \quad (5.26)$$

$$\hat{A}_{k|k-1}^x = \text{blkdiag}(\hat{A}_{k-1}^x, A_{k-1}^w, A_{k-1}^{ef}), \quad (5.27)$$

$$\hat{b}_{k|k-1}^x = \begin{bmatrix} \hat{b}_{k-1}^x \\ b_{k-1}^w \\ b_{k-1}^{ef} \end{bmatrix}, \quad (5.28)$$

$$\hat{P}_{k|k-1}^{xx} = F^x \hat{P}_{k-1}^{xx} (F^x)^\top + F^w Q_{k-1} (F^w)^\top. \quad (5.29)$$

Proof. From the linearized version (5.19), the mixed state vector $x_{k|k-1}$ can be split up in terms the mixed center estimate $\hat{c}_{k|k-1}^x$ in (5.23) and the remaining parcels

$$\begin{aligned} \hat{z}_{k|k-1}^x &= F^x \hat{z}_{k-1}^x + F^w z_{k-1}^w + (e_{k-1}^f - \hat{e}_{k-1}^f), \\ \hat{g}_{k|k-1}^x &= F^x \hat{g}_{k-1}^x + F^w g_{k-1}^w. \end{aligned}$$

Replacing all elements and realizations in $\hat{z}_{k|k-1}^x$ and $\hat{g}_{k|k-1}^x$ by the corresponding uncertainty representations yields

$$\begin{aligned} \hat{X}_{k|k-1} &= F^x \hat{X}_{k-1} \oplus F^w \mathcal{W}_{k-1} \oplus (\mathcal{R}_{k-1}^{ef} - \hat{e}_{k-1}^f), \\ \hat{X}_{k|k-1} &= F^x \hat{X}_{k-1} + F^w W_{k-1}. \end{aligned}$$

According to Properties 2.2.6 and 2.2.7, the CZ $\hat{X}_{k|k-1}$ (5.24) is described by (5.26)-(5.28). From Properties 2.1.1 and 2.1.2, the GRV $\hat{X}_{k|k-1}$ (5.25) is described by (5.29). ■

Having in mind Proposition 5.2.1, let the static equations (1.2) and (1.3) be unified as

$$\tilde{y}_k = \tilde{h}(x_k, v_k), \quad (5.30)$$

where

$$\begin{aligned} \tilde{y}_k &\triangleq \begin{bmatrix} y_k \\ 0_{m_c \times 1} \end{bmatrix}, \\ \tilde{h}(x_k, v_k) &\triangleq \begin{bmatrix} h(x_k, v_k) \\ g(x_k) \end{bmatrix}. \end{aligned}$$

In this case, the equality constraints (1.3) play the role of augmented measurements.

Following the procedure executed in (5.19), the function $\tilde{h}$ (5.30) is linearized as

$$\tilde{y}_k = \tilde{h}(\bar{x}_{k|k-1}, \bar{v}_k) + H^x(x_k - \bar{x}_{k|k-1}) + H^v(v_k - \bar{v}_k) + e_k^{\tilde{h}}, \quad (5.31)$$

where $H^x = \nabla_x \tilde{h}(\bar{x}_{k|k-1}, \bar{v}_k)$ and $H^v = \nabla_v \tilde{h}(\bar{x}_{k|k-1}, \bar{v}_k)$ are Jacobian matrices evaluated at the points

$$\begin{aligned} \bar{x}_{k|k-1} \in \mathcal{I}_{k|k-1}^{\alpha, x} &= \hat{c}_{k|k-1}^x \oplus \square \hat{\mathcal{X}}_{k|k-1} \oplus \square \mathcal{E}_{k|k-1}^x, \\ \bar{v}_k \in \mathcal{I}_k^{\alpha, v} &= 0_{r \times 1} \oplus \square \mathcal{V}_k \oplus \square \mathcal{E}_k^v. \end{aligned}$$

The prior punctual estimates are here yielded as follows:

$$\bar{x}_{k|k-1} = \arg \min_{x \in \mathcal{I}_{k|k-1}^{\alpha, x}} \|x - \hat{c}_{k|k-1}^x\|_1, \quad \bar{v}_k = \arg \min_{v \in \mathcal{I}_k^{\alpha, v}} \|v\|_1. \quad (5.32)$$

The linearization error $e_k^{\tilde{h}} \in \mathbb{R}^{(m+m_c)}$ is approximated by the CZ $\mathcal{R}_k^{\tilde{h}}$. This set is originated from the application of the Taylor extension or DC programming to the confidence sets $\mathcal{X}_{k|k-1}^{\alpha, x}$, $\mathcal{X}_k^{\alpha, v}$, $\mathcal{I}_{k|k-1}^{\alpha, x}$, and $\mathcal{I}_k^{\alpha, v}$.

Since the linear approximations are finished, we propose the following state estimator.

Proposition 5.2.2 (State estimator). Consider Proposition 5.2.1 and the linearized measurement model (5.31). Let $x_{k|k-1}$ and v_k be mixed-uncertainty vectors with known representations, whose Gaussian realizations $\hat{g}_{k|k-1}^x$ and g_k^v are uncorrelated for all $k \geq 1$. Let $\mathcal{R}_k^{\tilde{h}} = \{G_k^{\tilde{h}}, c_k^{\tilde{h}}\} \ni e_k^{\tilde{h}}$ be the known linearization enclosure. Let also the mixed center estimate $\hat{c}_k^x$ be set as

$$\hat{c}_k^x = \hat{c}_{k|k-1}^x + K_k(\tilde{y}_k - \hat{c}_{k|k-1}^{\tilde{y}}),$$

where $K_k \in \mathbb{R}^{n \times (m+m_c)}$ is a known gain matrix and

$$\hat{c}_{k|k-1}^{\tilde{y}} = \tilde{h}(\bar{x}_{k|k-1}, \bar{v}_k) + H^x(\hat{c}_{k|k-1}^x - \bar{x}_{k|k-1}) - H^v \bar{v}_k + c_k^{\tilde{h}}. \quad (5.33)$$

Then, the *a posteriori* mixed state vector $x_k = \hat{c}_k^x + \hat{z}_k^x + \hat{g}_k^x$ is characterized by

$$\hat{c}_k^x = (I_n - K_k H^x) \hat{c}_{k|k-1}^x + K_k(\tilde{y}_k - \tilde{h}(\bar{x}_{k|k-1}, \bar{v}_k) + H^x \bar{x}_{k|k-1} + H^v \bar{v}_k - c_k^{\tilde{h}}), \quad (5.34)$$

$$\begin{aligned} \hat{\mathcal{X}}_k &= (I_n - K_k H^x) \hat{\mathcal{X}}_{k|k-1} \oplus (-K_k H^v \mathcal{V}_k) \oplus (-K_k(\mathcal{R}_k^{\tilde{h}} - c_k^{\tilde{h}})) \\ &= \{\hat{G}_k^x, 0_{n \times 1}, \hat{A}_k^x, \hat{b}_k^x\}, \end{aligned} \quad (5.35)$$

$$\hat{\mathcal{X}}_k = (I_n - K_k H^x) \hat{\mathcal{X}}_{k|k-1} - K_k H^v \mathcal{V}_k \sim \mathcal{N}(0_{n \times 1}, \hat{P}_k^{xx}), \quad (5.36)$$

where

$$\hat{G}_k^x = [(I_n - K_k H^x) \hat{G}_{k|k-1}^x \quad -K_k H^v G_k^v \quad -K_k G_k^{\tilde{h}}], \quad (5.37)$$

$$\hat{A}_k^x = \text{blkdiag}(\hat{A}_{k|k-1}^x, A_k^v, A_k^{\tilde{h}}), \quad (5.38)$$

$$\hat{b}_k^x = \begin{bmatrix} \hat{b}_{k|k-1}^x \\ b_k^v \\ b_k^{e^h} \end{bmatrix}, \quad (5.39)$$

$$\hat{P}_k^{xx} = (\mathbf{I}_n - K_k H^x) \hat{P}_{k|k-1}^{xx} (\mathbf{I}_n - K_k H^x)^\top + K_k H^v R_k (H^v)^\top K_k^\top. \quad (5.40)$$

Proof. Defining $\tilde{y}_k \triangleq \hat{c}_{k|k-1}^{\bar{y}} + \hat{z}_{k|k-1}^{\bar{y}} + \hat{g}_{k|k-1}^{\bar{y}}$ and relating it with (5.31), we obtain the following terms:

$$\begin{aligned} \hat{c}_{k|k-1}^{\bar{y}} &= \tilde{h}(\bar{x}_{k|k-1}, \bar{v}_k) + H^x (\hat{c}_{k|k-1}^x - \bar{x}_{k|k-1}) - H^v \bar{v}_k + c_k^{e^h}, \\ \hat{z}_{k|k-1}^{\bar{y}} &= H^x \hat{z}_{k|k-1}^x + H^v z_k^v + (e_k^h - c_k^{e^h}), \\ \hat{g}_{k|k-1}^{\bar{y}} &= H^x \hat{g}_{k|k-1}^x + H^v g_k^v. \end{aligned}$$

By assigning a Kalman-like gain K_k , the center estimates $\hat{c}_{k|k-1}^x$ and $\hat{c}_{k|k-1}^{\bar{y}}$ can be combined as

$$\hat{c}_k^x = \hat{c}_{k|k-1}^x + K_k (\tilde{y}_k - \hat{c}_{k|k-1}^{\bar{y}}),$$

yielding the error terms

$$\begin{aligned} \hat{z}_k^x &= (\mathbf{I}_n - K_k H^x) \hat{z}_{k|k-1}^x - K_k H^v z_k^v - K_k (e_k^h - c_k^{e^h}), \\ \hat{g}_k^x &= (\mathbf{I}_n - K_k H^x) \hat{g}_{k|k-1}^x - K_k H^v g_k^v. \end{aligned}$$

After replacing these elements and realizations by their uncertainty representation, and finally, employing Properties 2.1.1, 2.1.2, 2.2.6, and 2.2.7, we obtain the zero-center CZ $\hat{\mathcal{X}}_k$ and the zero-mean GRV $\hat{X}_k$ described by (5.37)-(5.40). The result in (5.40) is verified for all $k \geq 1$ since x_0^g , g_{k-1}^w , and g_k^v are assumed uncorrelated. ■

Proposition 5.2.2 requires the knowledge of the design matrix K_k . The choice of K_k should take both constrained zonotopic and Gaussian parcels into account. CZs have been successfully applied to the trace function through of the outer approximation from CZs by zonotopes (Pan et al., 2022). This function is defined as J_k^z in (1.8). From stochastic filtering (Särkkä, 2013), Gaussian parcels are commonly evaluated by means of the cost function J_k^g defined in (1.5).

There exist different ways to outer-approximate CZs by zonotopes as shown in Example 4.2.1. Nevertheless, the fact of the matrix K_k being unknown makes the generator matrix $\hat{G}_k^x$ in (5.37) be also unknown. Therefore, we propose the following methodology.

Proposition 5.2.3. Let $[\zeta^L, \zeta^U] \triangleq \square \mathcal{B}(\hat{A}_k^x, \hat{b}_k^x)$ be the interval hull of the constrained unitary

box $\mathcal{B}(\hat{A}_k^x, \hat{b}_k^x)$ of the CZ $\hat{\mathcal{X}}_k$ in (5.35), where

$$[\zeta^L, \zeta^U] = \begin{bmatrix} [\zeta_{k|k-1}^{Lx}, \zeta_{k|k-1}^{Ux}] \\ [\zeta_k^{Lv}, \zeta_k^{Uv}] \\ [\zeta_k^{Le^h}, \zeta_k^{Ue^h}] \end{bmatrix} = \begin{bmatrix} \square\mathcal{B}(\hat{A}_{k|k-1}^x, \hat{b}_{k|k-1}^x) \\ \square\mathcal{B}(A_k^v, b_k^v) \\ \square\mathcal{B}(A_k^{e^h}, b_k^{e^h}) \end{bmatrix}. \quad (5.41)$$

By employing the enclosure (5.41), the zonotope $\tilde{\mathcal{X}}_k \supset \hat{\mathcal{X}}_k$ is obtained as follows:

$$\tilde{\mathcal{X}}_k = \{\tilde{G}_k^x, \tilde{c}_k^x\} = \hat{G}_k^x [\zeta^L, \zeta^U], \quad (5.42)$$

where

$$\tilde{c}_k^x = \frac{1}{2} \hat{G}_k^x (\zeta^L + \zeta^U), \quad (5.43)$$

$$\begin{aligned} \tilde{G}_k^x &= \frac{1}{2} \hat{G}_k^x \text{diag}(\zeta^U - \zeta^L) \\ &= \begin{bmatrix} (\mathbf{I}_n - K_k H^x) \tilde{G}_{k|k-1}^x & -K_k H^v \tilde{G}_k^v & -K_k \tilde{G}_k^{e^h} \end{bmatrix}, \end{aligned} \quad (5.44)$$

with $\tilde{G}_{k|k-1}^x = \frac{1}{2} \hat{G}_{k|k-1}^x \text{diag}(\zeta_{k|k-1}^{Ux} - \zeta_{k|k-1}^{Lx})$, $\tilde{G}_k^v = \frac{1}{2} G_k^v \text{diag}(\zeta_k^{Uv} - \zeta_k^{Lv})$, and $\tilde{G}_k^{e^h} = \frac{1}{2} G_k^{e^h} \text{diag}(\zeta_k^{Ue^h} - \zeta_k^{Le^h})$.

Proof. Since the CZ $\hat{\mathcal{X}}_k$ given by (5.35) is resulting from the operations of linear mapping and Minkowski sum, the equality constraints given by $\hat{A}_k^x$ (5.38) and $\hat{b}_k^x$ (5.39) are not affected by the matrix K_k . This allows the computation of the interval hull $\square\mathcal{B}(\hat{A}_k^x, \hat{b}_k^x)$, which can be dismembered in smaller calculus due to the block-diagonal characteristic of $\hat{A}_k^x$. After replacing $\mathcal{B}(\hat{A}_k^x, \hat{b}_k^x)$ by $\square\mathcal{B}(\hat{A}_k^x, \hat{b}_k^x)$, and thereby, employing affine arithmetic to represent boxes as zero-constraint CZs, the zonotope $\tilde{\mathcal{X}}_k$ in (5.42) is finally characterized by the center $\tilde{c}_k^x$ and generator matrix $\tilde{G}_k^x$ given by (5.43) and (5.44), respectively. ■

Remark 5.2.1. Let $\hat{\mathcal{X}}_{k|k-1}$, $\mathcal{V}_k$, and $(\mathcal{R}_k^{e^h} - c_k^{e^h})$ be zero-center CZs with $(n_h, n_h^v, n_h^{e^h})$ constraints and $(n_g, n_g^v, n_g^{e^h})$ generators. If solvers of LPs with the same complexity order of Kelner & Spielman (2006) are used in Proposition 5.2.3, we have

$$\begin{aligned} O([\zeta^L, \zeta^U]) &= O([\zeta_{k|k-1}^{Lx}, \zeta_{k|k-1}^{Ux}]) + O([\zeta_k^{Lv}, \zeta_k^{Uv}]) + O([\zeta_k^{Le^h}, \zeta_k^{Ue^h}]) \\ &= O(n_g^2 (n_h + n_g)^3 + (n_g^v)^2 (n_h^v + n_g^v)^3 + (n_g^{e^h})^2 (n_h^{e^h} + n_g^{e^h})^3). \end{aligned}$$

If the iterative method proposed in Scott et al. (2016, Algorithm 1) is used otherwise, we reach a smaller computational cost:

$$O([\zeta^L, \zeta^U]) = O(n_h^2 n_g + n_h n_g^2 + (n_h^v)^2 n_g^v + n_h^v (n_g^v)^2 + (n_h^{e^h})^2 n_g^{e^h} + n_h^{e^h} (n_g^{e^h})^2).$$

For both cases, note that the partition of $\square\mathcal{B}(\hat{A}_k^x, \hat{b}_k^x)$ in three smaller boxes implies reduction of computational cost.

For mixed uncertainties, there exist two conflicting objectives to attend (at the same time) with the same design matrix K_k . That is, the minimization of both J_k^z (1.8) and J_k^s (1.5). According to the biobjective interpretation presented in Section 4.2, we make use of the cost functions J_k^{zg} and $\tilde{J}_k^{zg}$ defined in (1.9) and (4.59), respectively.

Next, we present the main results of the general estimator. The first theorem solves the optimization problem

$$\min_{K_k} J_k^{zg} \quad \text{s.t.} \quad D_k \hat{C}_k^x \leq d_k, \quad (5.45)$$

from which we can obtain the pairs (J_k^{Lz}, J_k^{Ug}) and (J_k^{Uz}, J_k^{Lg}) doing $\eta = 0$ and $\eta = 1$, respectively. The second Theorem yields the desired solution $\check{K}_k$ by solving the optimization problem

$$\min_{\check{K}_k} \tilde{J}_k^{zg} \quad \text{s.t.} \quad D_k \hat{C}_k^x \leq d_k. \quad (5.46)$$

In both cases, the equality constraints (1.3) are embedded in the augmented measurement model (5.30).

Theorem 5.2.1. Consider the cost function (1.9), the known weight parameter $\eta \in [0, 1]$, and the inequality (1.4). Let $z = \text{vec}(K_k) \in \mathbb{R}^{(nm+nm_c)}$ be the variable vector, where $\text{vec}(\cdot)$ is the vectorization operator that stacks up columns of a matrix starting from the first one to the last one, and $M \in \mathbb{R}^{(nm+nm_c) \times (nm+nm_c)}$ be the vectorized transpose matrix such that $\text{vec}(K_k^\top) = Mz$. Let $\otimes$ be the Kronecker product, and $\text{reshape}(z, n, m + m_c)$ be the operator that takes the $n(m + m_c)$ elements of z to build a matrix with dimension $n \times (m + m_c)$. Then, solving the optimization problem (5.45) is equivalent to solving the quadratic optimization problem

$$\min \frac{1}{2} z^\top H z + f^\top z \quad \text{s.t.} \quad \check{D}z \leq \check{d}, \quad (5.47)$$

where

$$H = 2M^\top \check{P}M, \quad (5.48)$$

$$f^\top = -2\text{vec}\left(H^* \hat{P}_{k|k-1}^{zg}\right)^\top M, \quad (5.49)$$

$$\check{P} = I_n \otimes \hat{S}_{k|k-1}^{yy}, \quad (5.50)$$

$$\check{D} = \left(\tilde{y}_k - \hat{c}_{k|k-1}^y\right)^\top \otimes D_k, \quad (5.51)$$

$$\check{d} = d_k - D_k \hat{C}_{k|k-1}^x, \quad (5.52)$$

$$\hat{P}_{k|k-1}^{zg} = (1 - \eta) \tilde{G}_{k|k-1}^x \left(\tilde{G}_{k|k-1}^x\right)^\top + \eta \hat{P}_{k|k-1}^{xx}, \quad (5.53)$$

$$R_k^{zg} = (1 - \eta) \left(H^v \tilde{G}_k^v \left(\tilde{G}_k^v\right)^\top (H^v)^\top + \tilde{G}_k^{e^h} \left(\tilde{G}_k^{e^h}\right)^\top\right) + \eta H^v R_k (H^v)^\top, \quad (5.54)$$

$$\hat{S}_{k|k-1}^{yy} = H^* \hat{P}_{k|k-1}^{zg} (H^*)^\top + R_k^{zg}, \quad (5.55)$$

from which we obtain the optimal vector $\check{z}$, and thereby, the optimal matrix

$$\check{K}_k = \text{reshape}(\check{z}, n, m + m_c). \quad (5.56)$$

Proof. First, the application of (1.4) to the mixed center estimate $\hat{c}_k^x = \hat{c}_{k|k-1}^x + K_k (\hat{y}_k - \hat{c}_{k|k-1}^y)$ produces $D_k K_k (\hat{y}_k - \hat{c}_{k|k-1}^y) \leq d_k - D_k \hat{c}_{k|k-1}^x$. These state constraints are imparted to the gain vector $z = \text{vec}(K_k)$ by employing Proposition 2.5.1, whose result is expressed in (5.47).

Second, the expansion of $J_k^{zg} = \text{tr}(P_k^{zg})$, given by (1.9), yields

$$\begin{aligned} J_k^{zg} &= \text{tr} \left((I_n - K_k H^x) \hat{P}_{k|k-1}^{zg} (I_n - K_k H^x)^\top + K_k R_k^{zg} K_k^\top \right) \\ &= \text{tr} \left(\hat{P}_{k|k-1}^{zg} - K_k H^x \hat{P}_{k|k-1}^{zg} - \hat{P}_{k|k-1}^{zg} (H^x)^\top K_k^\top + K_k \hat{S}_{k|k-1}^{yy} K_k^\top \right) \\ &= \text{tr} \left(\hat{P}_{k|k-1}^{zg} \right) - 2\text{tr} \left(\hat{P}_{k|k-1}^{zg} (H^x)^\top K_k^\top \right) + \text{tr} \left(K_k \hat{S}_{k|k-1}^{yy} K_k^\top \right). \end{aligned}$$

As $\text{tr}(\hat{P}_{k|k-1}^{zg})$ does not depend on K_k , it does not influence the computation of K_k ; then, it is disregarded. According to Proposition 2.5.1, the remaining traces can be rewritten as

$$\begin{aligned} \text{tr} \left(\hat{P}_{k|k-1}^{zg} (H^x)^\top K_k^\top \right) &= \text{vec} \left(H^x \hat{P}_{k|k-1}^{zg} \right)^\top \text{vec} (K_k^\top), \\ \text{tr} \left(K_k \hat{S}_{k|k-1}^{yy} K_k^\top \right) &= \text{vec} (K_k^\top)^\top (I_n \otimes \hat{S}_{k|k-1}^{yy}) \text{vec} (K_k^\top). \end{aligned}$$

Therefore,

$$J_k = \text{vec} (K_k^\top)^\top \check{P} \text{vec} (K_k^\top) - 2\text{vec} \left(H^x \hat{P}_{k|k-1}^{zg} \right)^\top \text{vec} (K_k^\top),$$

where $\check{P}$ is a symmetric and square matrix given by (5.50). In order to make explicit z in J_k , we take a matrix M such that $\text{vec}(K_k^\top) = M \text{vec}(K_k)$. Thereby, we define the matrix H in (5.48) and the vector $f^\top$ in (5.49) to achieve the format of (5.47). Since the quadratic optimization problem (5.47) is solved, the design matrix (5.56) is obtained by applying a reshape in elements from $\check{z}$, which is an inverse procedure to vectorization. ■

Theorem 5.2.2. Consider the normalized cost function (4.59) and the inequality (1.4). Let $z = \text{vec}(\tilde{K}_k) \in \mathbb{R}^{(nm+nm_c)}$ be the variable vector, where $\text{vec}(\cdot)$ is the vectorization operator that stacks up columns of a matrix starting from the first one to the last one, and $M \in \mathbb{R}^{(nm+nm_c) \times (nm+nm_c)}$ be the vectorized transpose matrix such that $\text{vec}(\tilde{K}_k^\top) = Mz$. Let $\otimes$ be the Kronecker product, and $\text{reshape}(z, n, m + m_c)$ be the operator that takes the $n(m + m_c)$ elements of z to build a matrix with dimension $n \times (m + m_c)$. Then, solving the optimization problem (5.46) is equivalent to solving the quadratic optimization problem

$$\min \frac{1}{2} z^\top H z + f^\top z \quad \text{s.t.} \quad \check{D} z \leq \check{d}, \quad (5.57)$$

where H , $f^\top$, $\check{D}$, and $\check{d}$ are given by (5.48), (5.49), (5.51), and (5.52), respectively, assuming that

$$\hat{P}_{k|k-1}^{zg} = \frac{1}{J_k^{\text{Uz}} - J_k^{\text{Lz}}} \tilde{G}_{k|k-1}^x \left(\tilde{G}_{k|k-1}^x \right)^\top + \frac{1}{J_k^{\text{Ug}} - J_k^{\text{Lg}}} \hat{P}_{k|k-1}^{\text{xx}}, \quad (5.58)$$

$$R_k^{zg} = \frac{1}{J_k^{\text{Uz}} - J_k^{\text{Lz}}} \left(H^\top \tilde{G}_k^v \left(\tilde{G}_k^v \right)^\top + \tilde{G}_k^{e^{\hat{h}}} \left(\tilde{G}_k^{e^{\hat{h}}} \right)^\top \right) + \frac{1}{J_k^{\text{Ug}} - J_k^{\text{Lg}}} H^\top R_k (H^\top)^\top, \quad (5.59)$$

from which we obtain the optimal vector $\check{z}$, and thereby, the optimal matrix

$$\check{K}_k = \text{reshape}(\check{z}, n, m + m_c). \quad (5.60)$$

Proof. By making explicit the costs J_k^z (1.8) and J_k^g (1.5) in (4.59), and disregarding the constant terms, we obtain

$$\tilde{J}_k = \frac{1}{J_k^{Uz} - J_k^{Lz}} J_k^z + \frac{1}{J_k^{Ug} - J_k^{Lg}} J_k^g.$$

This cost reminds J_k^g in (1.9), whose difference is the replacement of $(1-\eta)$ by $\frac{1}{J_k^{Uz} - J_k^{Lz}}$, and of η by $\frac{1}{J_k^{Ug} - J_k^{Lg}}$. Then, the remaining proof is given by executing the prior change in Theorem 5.2.1, resulting in the modified matrices (5.58) and (5.59). ■

5.2.1 Algorithm Description

The algorithms FOMF and DCMF have a similar structure (Proposition 5.2.2), with the difference being the linearization enclosure $\mathcal{R}$. Concerning the computation of the CZ $\mathcal{R}$, we rise a discussion. On the one hand, the first-order extension (Rego et al., 2020a) manipulates nonlinearities through interval arithmetic. By convenience, interval algebra may be applied directly, whose result may be conservative due to the dependency effect. A manner of reducing such effect is to use the truncated Taylor expansion (Moore et al., 2009). On the other hand, DC programming (Section 2.3) requires the usage of the difference of convex functions. It means that a function f must be rewritten in terms of two convex functions f^a and f^b , that is, $f = f^a - f^b$. This choice is not unique and leads to different results. For practical and precision reasons, functions f^a and f^b can be defined as in Proposition 2.3.3.

Given the starting CZ $\hat{\mathcal{X}}_{k-1}$, recall that the CZ $\hat{\mathcal{X}}_k$ (Proposition 5.2.2) is returned with a larger number of constraints and generators. To control this order growth over the state estimation, we apply CZR (given by (2.31)) to $\hat{\mathcal{X}}_{k-1}$, yielding the CZ $\mathcal{X}_{k-1}^l$ with φ_c constraints and φ_g generators. Soon after, we shift the center of the reduced-order CZ $\mathcal{X}_{k-1}^l$ to $\hat{c}_{k-1}^x$, keeping CZs with zero center. Thenceforth, $\mathcal{X}_{k-1}^l$ plays the role of $\hat{\mathcal{X}}_{k-1}$.

Analogously to Rego et al. (2021), we can enhance the starting mixed estimates $\hat{c}_{k-1}^x$, $\hat{\mathcal{X}}_{k-1}$, and $\hat{\mathcal{X}}_{k-1}$ thanks to the knowledge of the initial measurement $\tilde{y}_0$. Specifically, at $k = 0$, the forecast step is skipped to directly execute the data-assimilation step (Proposition 5.2.2). Next, an iteration of the general mixed-uncertainty filter (MF) is presented in Algorithm 5.2, which solves Problem 1.2.4.

5.2.2 Special Case

Assume that the state inequality constraints (1.4) describe a convex polytope. This feasibility set can be equivalently represented by the CZ $\mathcal{X}_k^F$. Therefore, we present a

Algorithm 5.2: $[\hat{\mathbf{c}}_k^x, \hat{\mathbf{X}}_k, \hat{\mathbf{X}}_k^\alpha, \mathcal{I}_k^\alpha] = \text{MF}(\hat{\mathbf{c}}_{k-1}^x, \hat{\mathbf{X}}_{k-1}, \hat{\mathbf{X}}_{k-1}, D_k, d_k, M, f, u_{k-1}, \mathcal{W}_{k-1}, W_{k-1}, \tilde{\mathbf{y}}_k, \tilde{\mathbf{h}}, \mathcal{V}_k, V_k, \varphi_c, \varphi_g, \alpha)$

- 1 Compute $\mathcal{X}_{k-1}^l$ as in (4.67), and shift the resulting center to $\hat{\mathbf{c}}_{k-1}^x$
 - 2 Apply the first-order Taylor extension (Rego et al., 2020a, Theorem 3) or DC programming (Lemma 2.3.1) to linearize the models f (1.1) and $\tilde{\mathbf{h}}$ (5.30)
 - 3 Apply Proposition 5.2.1 to yield the *a priori* mixed state estimates $\hat{\mathbf{c}}_{k|k-1}^x$, $\hat{\mathbf{X}}_{k|k-1}$, and $\hat{\mathbf{X}}_{k|k-1}^\alpha$, where $\hat{\mathbf{c}}_{k|k-1}^x$ is given by (5.23), $\hat{\mathbf{X}}_{k|k-1}$ is given by (5.24), (5.26)–(5.28), and $\hat{\mathbf{X}}_{k|k-1}^\alpha$ is given by (5.25), (5.29)
 - 4 Apply Proposition 5.2.3 to obtain the generator matrices $\tilde{\mathbf{G}}_{k|k-1}^x$, $\tilde{\mathbf{G}}_k^v$, and $\tilde{\mathbf{G}}_k^{\tilde{\mathbf{h}}}$
 - 5 Given $\eta = 0$, compute the covariance matrices $\hat{\mathbf{P}}_{k|k-1}^{zg}$, $\mathbf{R}_k^{zg}$, and $\hat{\mathbf{S}}_{k|k-1}^{yy}$ using (5.53)–(5.55)
 - 6 Apply Theorem 5.2.1 to compute the gain matrix $\check{\mathbf{K}}_k$
 - 7 Compute $\tilde{\mathbf{G}}_k^x$ in (5.44) and $\hat{\mathbf{P}}_k^{xx}$ in (5.40) to determine $\mathbf{J}_k^{Lz}$ and $\mathbf{J}_k^{Ug}$ given by (1.8) and (1.5), respectively
 - 8 Execute the steps (5)–(7) with $\eta = 1$ to compute $\mathbf{J}_k^{Uz}$ and $\mathbf{J}_k^{Lg}$ using (1.8) and (1.5), respectively
 - 9 Apply Theorem 5.2.2 to compute the final matrix $\check{\mathbf{K}}_k$
 - 10 Apply Proposition 5.2.2 to compute the *a posteriori* mixed state estimates $\hat{\mathbf{c}}_k^x$, $\hat{\mathbf{X}}_k$, and $\hat{\mathbf{X}}_k^\alpha$, where $\hat{\mathbf{c}}_k^x$ is given by (5.34), $\hat{\mathbf{X}}_k$ is given by (5.35), (5.37)–(5.39) and $\hat{\mathbf{X}}_k^\alpha$ is given by (5.36), (5.40)
 - 11 Compute the confidence CZ $\mathcal{X}_k^\alpha$ and the confidence box $\mathcal{I}_k^\alpha$ using (2.47) and (2.49), respectively
-

variation from Algorithm 5.2 in which the quadratic optimization problems (Theorems 5.2.1 and 5.2.2) are dismembered to enforce equality and inequality constraints in separate steps. Our first motivation is to reduce computational cost by replacing quadratic solvers with explicit solutions, and the second one is to obtain a better refinement of uncertainties. For the unconstrained cases, we just do not enforce state constraints.

Next, two corollaries are presented to calculate gain matrices. They are special cases of Theorems 5.2.1 and 5.2.2 since the inequality constraints (1.4) are disregarded. Remember that the equality constraints (1.3) are embedded in (5.30), therefore we reach an unconstrained optimization problem.

Corollary 5.2.1. Consider the cost function (1.9) and the known weight parameter $\eta \in [0, 1]$. Let $\mathbf{K}_k \in \mathbb{R}^{n \times (m+m_c)}$ be the variable matrix. Then, the minimization of (1.9) produces the optimal gain

$$\check{\mathbf{K}}_k = \hat{\mathbf{P}}_{k|k-1}^{zg} (\mathbf{H}^x)^\top (\hat{\mathbf{S}}_{k|k-1}^{yy})^{-1}, \quad (5.61)$$

where $\hat{\mathbf{P}}_{k|k-1}^{zg}$ and $\hat{\mathbf{S}}_{k|k-1}^{yy}$ are given by (5.53) and (5.55), respectively.

Proof. From the proof of Theorem 5.2.1, we have

$$\mathbf{J}_k^{zg} = \text{tr}(\hat{\mathbf{P}}_{k|k-1}^{zg}) - 2\text{tr}(\hat{\mathbf{P}}_{k|k-1}^{zg} (\mathbf{H}^x)^\top \mathbf{K}_k^\top) + \text{tr}(\mathbf{K}_k \hat{\mathbf{S}}_{k|k-1}^{yy} \mathbf{K}_k^\top).$$

According to Proposition 2.5.1, the operation $\frac{\partial f_k^{zg}}{\partial K_k} = 0$ yields

$$H^x \hat{P}_{k|k-1}^{zg} = \hat{S}_{k|k-1}^{yy} K_k^\top,$$

in which the symmetry of both $\hat{P}_{k|k-1}^{zg}$ and $\hat{S}_{k|k-1}^{yy}$ has been exploited. After a matrix transpose, the prior result becomes $K_k \hat{S}_{k|k-1}^{yy} = \hat{P}_{k|k-1}^{zg} (H^x)^\top$, whose optimal solution $\check{K}_k$ is given by (5.61). The existence of this solution is certified by $\hat{S}_{k|k-1}^{yy} > 0_{(m+m_c) \times (m+m_c)}$, and the uniqueness of $\check{K}_k$ is originated from the matrix inversion. ■

Corollary 5.2.2. Consider the normalized cost function (4.59) and let $\check{K}_k \in \mathbb{R}^{n \times (m+m_c)}$ be the variable matrix. Then, the minimization of (4.59) produces the optimal gain

$$\check{K}_k = \hat{P}_{k|k-1}^{zg} (H^x)^\top (\hat{S}_{k|k-1}^{yy})^{-1}, \quad (5.62)$$

where $\hat{S}_{k|k-1}^{yy}$ and $\hat{P}_{k|k-1}^{zg}$ are given by (5.55) and (5.58), respectively, considering R_k^{zg} given by (5.59).

Proof. Disregarding the constant parcels of $\tilde{J}_k^{zg}$ in (4.59) yields

$$\tilde{J}_k = \frac{1}{J_k^{Uz} - J_k^{Lz}} J_k^z + \frac{1}{J_k^{Ug} - J_k^{Lg}} J_k^g,$$

whose difference with respect to Corollary 5.2.1 is the replacement of $(1 - \eta)$ by $\frac{1}{J_k^{Uz} - J_k^{Lz}}$, and of η by $\frac{1}{J_k^{Ug} - J_k^{Lg}}$. Carrying out this change, the remaining proof is similar to the proof of Corollary 5.2.1 with the covariance matrices $\hat{P}_{k|k-1}^{zg}$ and R_k^{zg} given by (5.58) and (5.59), respectively. ■

With the acquisition of $\check{K}_k$ in (5.62), Proposition 5.2.2 is employed to compute the mixed state estimates satisfying the equality constraints (1.3). The posterior step is to enforce the inequality constraints $\mathcal{X}_k^F = \{G_k^{x^F}, c_k^{x^F}, A_k^{x^F}, b_k^{x^F}\}$ on the state estimates $\hat{c}_k^x$, $\hat{X}_k$, and $\hat{X}_k$. The methodology used here is the same one than Section 4.2.

Next, the explicit version of the mixed-uncertainty filter (MF-E) is described in Algorithm 5.3. In order to highlight differences and similarities, both quadratic and explicit versions are illustrated in Figure 5.3.

5.2.3 Numerical Results

In this subsection, the FOMF and DCMF algorithms are experimented over three numerical examples. We employ the methodology discussed in Subsection 3.2.4 to obtain the punctual estimates $\bar{z}$, and thereby, approximate nonlinear models. Furthermore, DCMF is executed with Proposition 2.3.3. In order to compare our algorithms with existing stochastic, deterministic, and mixed-uncertainty approaches, we implement the

Algorithm 5.3: $[\hat{c}_k^x, \hat{\mathcal{X}}_k, \hat{\mathcal{X}}_k^c, \mathcal{I}_k^c] = \text{MF-E}(\hat{c}_{k-1}^x, \hat{\mathcal{X}}_{k-1}, \hat{\mathcal{X}}_{k-1}^c, \mathcal{X}_k^F, f, u_{k-1}, \mathcal{W}_{k-1}, W_{k-1}, \tilde{y}_k, \tilde{h}, \mathcal{V}_k, V_k, \varphi_c, \varphi_g, \alpha, \kappa)$

- 1 Apply the steps 1–10 of Algorithm 5.2 with Corollaries 5.2.1 and 5.2.2 (instead of Theorems 5.2.1 and 5.2.2) to compute the mixed state estimates $\hat{c}_k^x$ (5.34), $\hat{\mathcal{X}}_k$ (5.35), (5.37)–(5.39), and $\hat{\mathcal{X}}_k^c$ (5.36), (5.40)
 - 2 Apply intersection between the CZs $\hat{\mathcal{X}}_k^+ = \{\hat{G}_k^x, c_k^{x^F}, \hat{A}_k^x, \hat{b}_k^x\}$ and $\mathcal{X}_k^F = \{G_k^{x^F}, c_k^{x^F}, A_k^{x^F}, b_k^{x^F}\}$ to yield the CZ $\hat{\mathcal{X}}_k^{t+} = \hat{\mathcal{X}}_k^+ \cap \mathcal{X}_k^F$
 - 3 Annul the center of $\hat{\mathcal{X}}_k^{t+}$ to obtain the zero-center CZ $\hat{\mathcal{X}}_k^t$
 - 4 Apply ICUT (Subsection A.2.2.2) to $\hat{\mathcal{X}}_k^+ \sim \mathcal{N}(\hat{c}_k^x, \hat{P}_k^{xx})$, yielding the estimates $\hat{c}_k^{xt}$ and $\hat{P}_k^{xxt}$
 - 5 Let the mixed state estimates be given by $\hat{c}_k^x = \hat{c}_k^{xt}$, $\hat{\mathcal{X}}_k = \hat{\mathcal{X}}_k^t$, and $\hat{P}_k^{xx} = \hat{P}_k^{xxt}$
 - 6 Compute the constrained confidence CZ $\mathcal{X}_k^c$ and the constrained confidence box $\mathcal{I}_k^c$ using (4.87) and (4.88), respectively
-

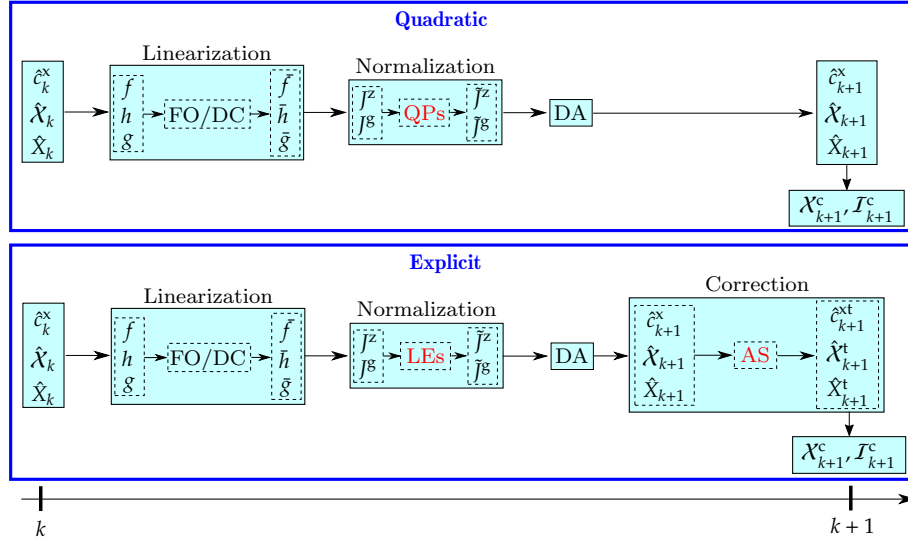


Figure 5.3: Illustration of an iteration of the mixed-uncertainty filter (MF) with the quadratic (Algorithm 5.2) or explicit (Algorithm 5.3) version. Initially, MF linearizes the models f in (1.1), h in (1.2), and g in (1.3) based on the first-order extension (FO) or the difference of convex functions (DC). Thenceforth, the versions differ with respect to the cost normalization, whose step is composed of quadratic programs (QPs) or linear equations (LEs). Both formulations cover the state equality constraints (1.3), but only QPs include the state inequality constraints (1.4). This is the reason why the explicit version adds the admissibility step (AS), in which exclusively polytopic constraints are used to intersect CZs ($\hat{\mathcal{X}}_{k+1}$) and apply ICUT to GRVs ($\hat{\mathcal{X}}_{k+1}$) and mixed center estimate $\hat{c}_{k+1}^x$. Besides estimating mixed uncertainties after the data-assimilation (DA) step, the estimators combine such estimates to yield the constrained sets $\mathcal{X}_{k+1}^c$ in (4.87) and $\mathcal{I}_{k+1}^c$ in (4.88).

algorithms GCUKF (Teixeira et al., 2008a), CZMV (Rego et al., 2021), and EZGKF (Algorithm 5.1) for the first example. GCUKF is a gain-constrained Kalman estimator to approach stochastic systems with unknown inputs aiming at unbiased estimates; CZMV

is a set-membership estimator based on CZs and mean value extension; EZGKF is a mixed-uncertainty estimator based on zonotopes and quasi-LPV approach. For comparison purposes, we compute three performance indexes: T^{CPU} in (3.57), RMSE in (3.58), and $r^\square$ in (3.84).

To calculate the punctual estimate $\bar{x}$ in RMSE, we here solve $\min_{x_k \in \mathcal{X}_k^\alpha} \|x_k - \hat{c}_k^\alpha\|_1$, with $\mathcal{X}_k^\alpha$ being given by (2.47). In turn, $r^\square$ is computed over confidence boxes $\mathcal{I}_k^\alpha$ given by (2.49). For GCUKF, $\bar{x}$ is chosen as the *a posteriori* mean estimate, and $r^\square$ is computed from the interval hull of confidence ellipsoids. These ellipsoids are here traced with 99.73% confidence level ($\alpha = 0.0027$). Then, we analyze both accuracy (RMSE) and precision ($r^\square$) of algorithms. The following computer configuration was used: 8 GB RAM 1333 MHz, Windows 10 Pro, and AMD FX-6300 CPU 3.50 GHz. All implementations were executed in MATLAB 9.11 (Release) with Gurobi 9.1, INTLAB 12 (Rump, 1999), and MPT3 (Herceg et al., 2013). Specifically, Gurobi is employed to solve linear and quadratic programs, INTLAB is used to compute interval estimates, and MPT is used to sketch sets.

5.2.3.1 Van der Pol Oscillator

Consider the Van der Pol oscillator discussed in (5.15)-(5.16), whose process equation is affected by a set-bounded uncertainty $\mathcal{W}$ while the measurement equation is affected by a stochastic-nature uncertainty V . To estimate states, we set $\hat{c}_0^\alpha = 0_{2 \times 1}$, $\hat{X}_0 = \{0.5\mathbf{I}_2, 0_{2 \times 1}\}$, $\hat{X}_0 \sim \mathcal{N}(0_{2 \times 1}, \frac{1}{47.316}\mathbf{I}_2)$, $\varphi_c = 3$, $\varphi_g = 8$, $k_0 = 10$, and $\alpha = 0.0027$, such that $x_0 \in \mathcal{X}_0^\alpha$. For DCMF, we define DC functions as in Proposition 2.3.3 and use the center of the confidence box as linearization point. Both FOMF and DCMF are executed with their explicit version because there is no state constraint. For GCUKF (Teixeira et al., 2008a), we set $\hat{P}_0^{\text{xx}} = \frac{1}{11.829}\mathbf{I}_2$ and $Q_{k-1} = 1 \times 10^{-6}\mathbf{I}_2$ to improve the numerical stability. For CZMV (Rego et al., 2021), we set $\hat{X}_0 = \{\mathbf{I}_2, 0_{2 \times 1}\}$ and $\mathcal{V} = \{0.6, 0\}$ to contain the worst-case realizations of the GRV V with a confidence level at least 99.73%. For EZGKF, we set the same matrices $A_{0,k-1}$ and $A_{1,k-1}$ proposed in Subsection 5.1.1.1 and employ the cost normalization of MF. The order of generators for zonotopes and CZs is here fixed using the algorithm proposed in Scott et al. (2016), and interval operations are computed through the direct application of interval arithmetic.

First of all, we remind that the stochastic, deterministic, and mixed-uncertainty approaches have different bases and motivations to estimate states. Then, the comparisons among them should be more conceptual rather than numerical. This is our direction to justify the differences on performance indices.

In Figure 5.4, the state estimates are depicted for one specific simulation. Boxes are sketched instead of CZs due to computational issues. The available illustrations aid to understand some differences verified in Table 5.6. For instance, in this case study, CZMV returns the largest sets. A justification is that set-based operations can imply conservatism to cover all possible states x_k in a set $\mathcal{X}_k$. GCUKF is a stochastic algorithm whose solutions

Table 5.6: Results of T^{CPU} , RMSE, and $r^\square$ for the Van der Pol oscillator using FOMF and DCMF.

Algorithms	T^{CPU}	RMSE ₁	RMSE ₂	$r^\square$
FOMF-E	7.00 ms	0.0500	0.143	0.680
DCMF-E	3.40 ms	0.0444	0.142	0.650
EZGKF	3.60 ms	0.0450	0.130	0.603
GCUKF (Teixeira et al., 2008a)	4.20 ms	0.0610	0.208	0.712
CZMV (Rego et al., 2021)	20.7 ms	0.0932	0.236	1.47

address a confidence region. Since $\alpha > 0$, bounded regions can be represented by confidence ellipsoids. These regions are not guaranteed, but reduce the conservatism of CZMV. EZGKF produces better solutions than GCUKF because the bounds of z_{k-1}^w are embedded in a zonotopic variable (instead of rejecting the disturbance z_{k-1}^w as does GCUKF), denoting the mixed character of the estimator.

In this case study, there are no state constraints and the set-bounded uncertainties are zonotopic. Therefore, the main difference noted among EZGKF and our novel algorithms (FOMF and DCMF) is the type of approximation (quasi-LPV or analytical linearization). In comparison with EZGKF, DCMF here contributes with the smallest processing time; this is an advantage from the explicit version over the quadratic version. DCMF also contributes with a better initial precision in comparison with FOMF, and therefore its $r^\square$ is better as shown in Table 5.6. However, FOMF yields competitive precision with EZGKF after the initialization effect at the cost of a larger T^{CPU} . As corroborated by Table 5.6, no significant difference of RMSE is realized among mixed-uncertainty algorithms due to the fact that they are designed under the same filter structure. Then, for scenarios as this one where the conservativeness introduced by interval arithmetic and by order-reduction methods is not significant, FOMF is motivated to reach tight results. In turn, DCMF provides a tradeoff between precision and processing time. In general terms, both FOMF and DCMF are motivated to work on asymmetric mixed uncertainties and on state constraints.

5.2.4 Batch Reactor

The reversible gas-phase reaction in a batch reactor is considered (Alamo et al., 2005; de Paula et al., 2018), $2\text{R}^{\text{A}} \xrightleftharpoons[k_2]{k_1} \text{R}^{\text{B}}$, where $k_1 = 0.16 \text{ min}^{-1}\text{atm}^{-1}$ and $k_2 = 0.0064 \text{ min}^{-1}$. The reactions occur in isothermal and constant volume conditions. Defining the state vector $x(t) \in \mathbb{R}_+^2$ as the partial pressures of R^{A} and R^{B} in atm, the system is modeled as

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} -2k_1x_1^2(t) + 2k_2x_2(t) \\ k_1x_1^2(t) - k_2x_2(t) \end{bmatrix}. \quad (5.63)$$

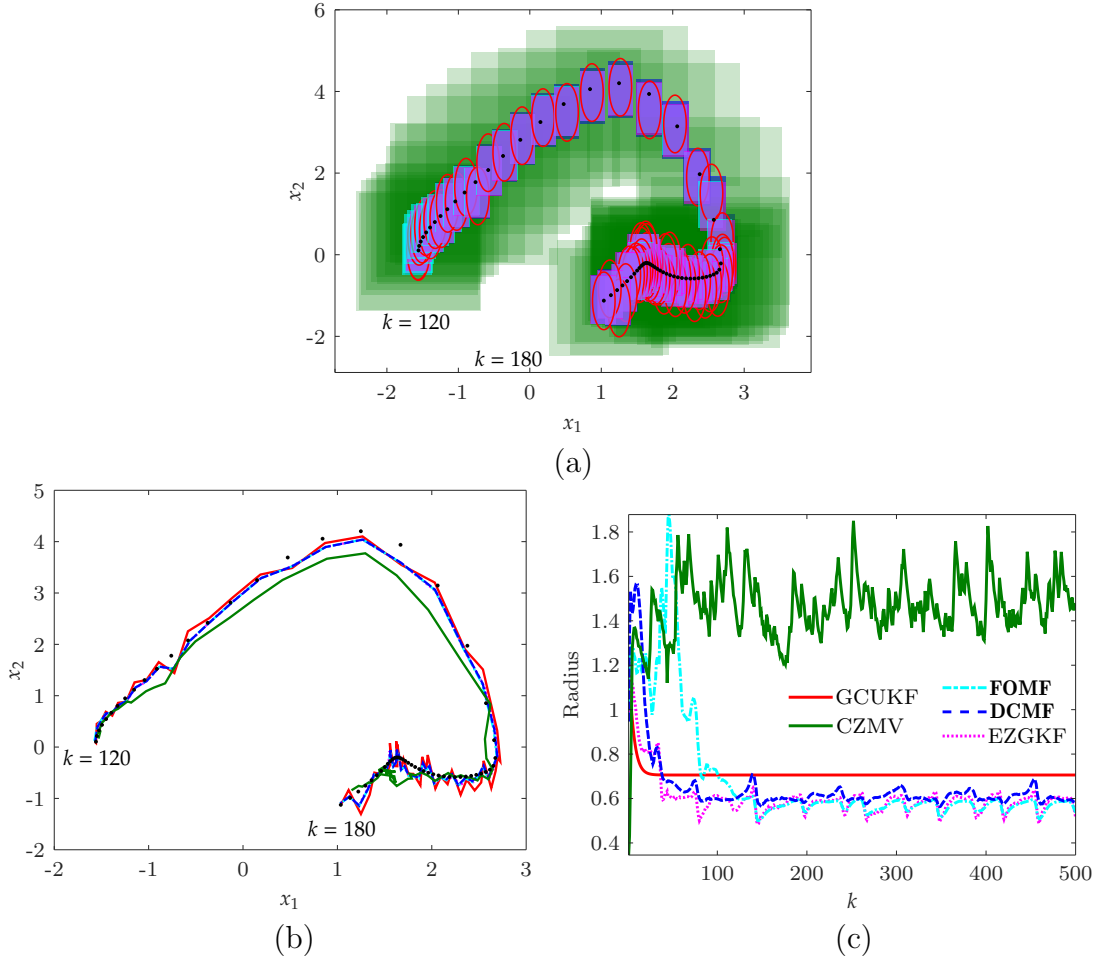


Figure 5.4: State estimation for the Van der Pol oscillator with FOMF and DCMF. In (a), the true states (black $\cdot$) are approximately involved by confidence boxes, computed by FOMF (cyan color), DCMF (blue color), and EZGKF (magenta color), interval hulls of CZs, computed by CZMV (green color), and confidence ellipsoids, computed by GCUKF (red color), for the discrete time horizon $k \in \{120, \dots, 180\}$. In (b), specific points are taken from the prior sets to approximate the true states. The largest radius of the interval hull of each set is sketched in (c).

At $t = kT_s$, the output pressure is measured as

$$y(kT_s) = \begin{bmatrix} 1 & 1 \end{bmatrix} x(kT_s) + v^z(kT_s) + v^s(kT_s), \quad (5.64)$$

whose noise terms $v^z(kT_s)$ and $v^s(kT_s)$ are represented by $\mathcal{V} = \{0.15, 0\}$ and $V \sim \mathcal{N}(0, \frac{1}{36}0.15^2)$, respectively. In order to discretize the continuous dynamics of (5.63), we employ the Euler method with sampling time $T_s = 6$ s. No process error is added to the discretized version. Moreover, the realizations $v^z(kT_s)$ and $v^s(kT_s)$ in (5.64) are taken from uniform and Gaussian distributions, respectively.

To simulate the true states, we set $x(0) = \begin{bmatrix} 3 & 1 \end{bmatrix}^\top$, $k_f = 100$, and $m_s = 70$. To estimate states, we employ the quadratic version of FOMF and DCMF with $\hat{c}_0^x = \begin{bmatrix} 2.5 & 1.5 \end{bmatrix}^\top$, $\hat{X}_0 =$

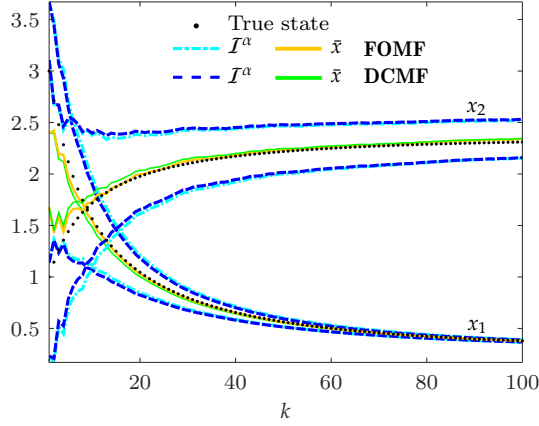


Figure 5.5: State estimation for the batch reactor with FOMF and DCMF.

$\{\hat{G}_0^x, 0_{2 \times 1}, \hat{A}_0^x, 0.0711\}$, where $\hat{G}_0^x = \begin{bmatrix} 0.1790 & 0.2988 & 0.5778 \\ -0.5011 & -0.3061 & -0.8629 \end{bmatrix}$ and $\hat{A}_0^x = [-1.9448 \quad -0.6657 \quad -0.4298]$, $\hat{P}_0^{xx} = \begin{bmatrix} 0.0385 & -0.0575 \\ -0.0575 & 0.0921 \end{bmatrix}$, $k_0 = 10$, $\alpha = 0.0027$, $\varphi_c = 1$, and $\varphi_g = 100$. Once the pressures are nonnegative, we enforce the halfspace constraints $D^i x_k \leq d^i$, where $D^i = -I_2$ and $d^i = 0_{2 \times 1}$. As the discretized process model is nonlinear, it is here rewritten as the DC function $x_k = f^a - f^b$, where $f^a = \begin{bmatrix} x_{1,k-1} + 2T_s k_2 x_{2,k-1} \\ x_{2,k-1} + T_s k_1 x_{1,k-1}^2 \end{bmatrix}$ and $f^b = \begin{bmatrix} 2T_s k_1 x_{1,k-1}^2 \\ T_s k_2 x_{2,k-1} \end{bmatrix}$, and thereby, we employ Proposition 2.3.2 to outer-approximate confidence CZs by parallelotopes in DCMF during the computation of the linearization remainder $\mathcal{R}_{k-1}^f$.

Figure 5.5 illustrates the true states being involved by confidence domains and approximated by punctual estimates $\tilde{x}$. In principle, no significant difference of precision is expected between FOMF and DCMF, but the accuracy may be different. As shown by Table 5.7, DCMF provides a slightly better precision (smaller $r^\square$) than FOMF demanding less computational effort (smaller T^{CPU}). In turn, FOMF reaches a better accuracy (smaller RMSE) than DCMF.

Recall that the initial CZ $\hat{X}_0$ was stated with an equality constraint, and therefore we set $\varphi_c = 1$. Once there are no polytopic constraints, DCMF only propagates that equality constraint over the state estimation. On the other hand, FOMF introduces new equality constraints due to the linearization remainder (see Table 5.5 for details), being necessary that some equality constraints are eliminated by the order-reduction algorithm. This elimination makes the reduced CZ become more symmetric, and due to simplicity of the case study, the order-reduction algorithm identifies that the equality constraints for FOMF are redundant around $k = 100$ and could be removed. As discussed in Subsection 3.2.2, the asymmetry of sets may contribute with a worse accuracy, and this justifies why DCMF reached a larger RMSE in Table 5.7. Conversely, the fact of FOMF returns CZs with a larger order than DCMF implies more generator reductions, resulting in a larger demand of T^{CPU} and in loss of precision due to the wrapping effect.

Table 5.7: Results of T^{CPU} , RMSE, and $r^{\square}$ for the batch reactor.

Algorithms	T^{CPU}	RMSE ₁	RMSE ₂	$r^{\square}$
FOMF	7.95 s	0.0114	0.0254	0.329
DCMF	0.0601 s	0.0268	0.0412	0.324

5.2.4.1 Simple Pendulum

Consider the continuous-time undamped and unforced pendulum described by (Teixeira et al., 2009; Simon & Simon, 2010)

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} x_2(t) \\ -\frac{a_g}{L} \sin(x_1(t)) \end{bmatrix}, \quad (5.65)$$

where $x_1(t)$ is the angular position, $x_2(t)$ is the angular speed, a_g is the gravity acceleration, and L is the length. By conservation of energy in (5.65), we have

$$M(0) = -m_q a_g L \cos(x_1(t)) + \frac{m_q L^2}{2} x_2^2(t), \quad (5.66)$$

where m_q is the pendulum mass and $M(0)$ is the total mechanical energy. We assume that noisy measurements of the angular speed are provided at $t = kT_s$ as follows:

$$y(kT_s) = x_2(kT_s) + g^v(kT_s), \quad (5.67)$$

with $g^v(kT_s)$ being realizations of the GRV $V \sim \mathcal{N}(0, R)$, for $R = \frac{1}{9} 16.558^2 \times 10^{-6}$, truncated in the box $\{16.558 \times 10^{-3}, 0\}$; this noise level is realistic with respect to an Inertial Measurement Unit (Rego et al., 2020a). By applying the fixed-step fourth-order Runge-Kutta integration to (5.65) with a step size equals to the sampling time T_s , we reach an explicit discretized model such that $M_k \triangleq M(kT_s) \approx M(0)$ for all $k \geq 1$.

To simulate the true states x_k , we set $a_g = 9.81 \text{ m/s}^2$, $L = 1 \text{ m}$, $m_q = 1 \text{ kg}$, $x(0) = [3\pi/4 \quad \pi/50]^\top$, $T_s = 10 \text{ ms}$, $k_f = 500$, and $m_s = 100$. This choice implies $M(0) = 6.9387 \text{ J}$. To estimate states, we employ DCMF with $\hat{X}_0 = \{0.5\mathbf{I}_2, 0_{2 \times 1}, [-0.1 \quad 0.1 \quad 0.1]^\top, -0.1\}$, $\hat{c}_0^x = [3\pi/5 \quad \pi/10]^\top$, $\hat{P}_0^{xx} = \frac{1}{4\varsigma_x} \mathbf{I}_2$, $\varsigma_x = 11.829$, $\alpha = 0.0027$, $\varphi_c = 1$, and $\varphi_g = 100$. We also add the GRV $W \sim \mathcal{N}(0_{2 \times 1}, 1 \times 10^{-12})$ to mitigate numerical issues related to the discretization. FOMF is not executed because it requires large φ_c and φ_g to reach convergence. Therefore, for comparison purposes, we implement the UKF based on measurement augmentation (MAUKF), proposed in Teixeira et al. (2009), considering the initial state estimate $\hat{X}_0 \sim \mathcal{N}(\hat{c}_0^x, \mathbf{I}_2)$ and the augmented noise $\tilde{V} \sim \mathcal{N}(0_{2 \times 1}, \text{blkdiag}(R, 1 \times 10^{-40}))$.

Figure 5.6 depicts the tight confidence domains returned by each algorithm. Due to the reached precision, the state estimators attain a good accuracy (small RMSE), as shown by Table 5.8. The main difference between the algorithms is the type of state estimator: DCMF is mixed whereas MAUKF is stochastic. As MAUKF has fixed-order covariance

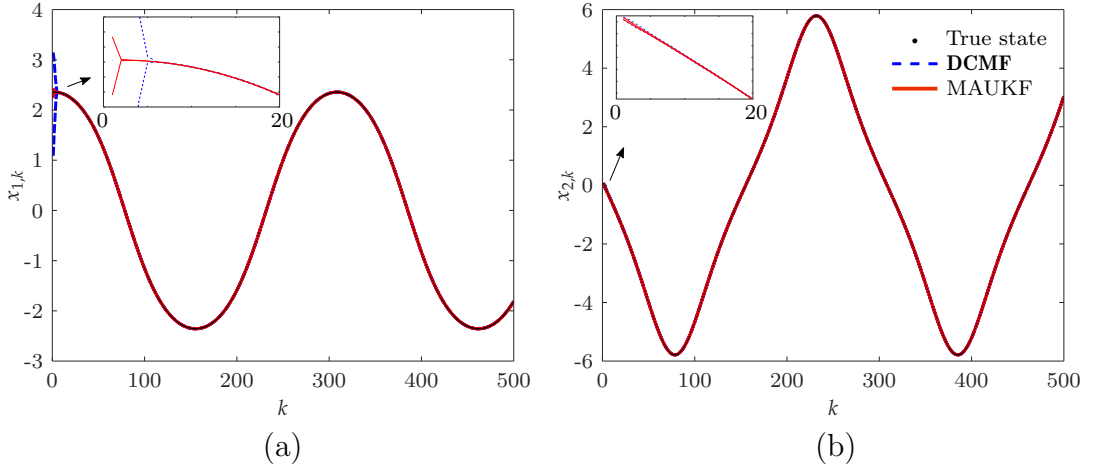


Figure 5.6: State estimation for the pendulum with DCMF and MAUKF. In (a) and (b), true states are involved by confidence intervals.

Table 5.8: Results of T^{CPU} , RMSE, and $r^{\square}$ for the pendulum using DCMF and MAUKF.

Algorithms	T^{CPU}	RMSE ₁	RMSE ₂	$r^{\square}$
DCMF-E	0.11 s	0.00070	0.0011	0.0089
MAUKF	4.9 ms	0.00010	0.00040	0.0017

matrix, it is indeed the fastest algorithm. Although the performance indexes with MAUKF were the smallest ones, MAUKF does not correctly capture the approximation errors. This is a deficiency inherent from methods as EKF and UKF, being necessary to add noise uncertainty to mitigate numerical issues such as divergence and inconsistency of estimates (Xiong et al., 2006). Differently, DCMF guarantees that, for the preset confidence level, the linearization remainder is enclosed; this is a significant advantage with respect to stochastic estimators. Moreover, the fact of that a larger state uncertainty is returned with DCMF is here expected, because we attribute an extra covariance in $\tilde{V}$ to only aid the convergence of MAUKF, but the procedure does not avoid inconsistency.

In this case study, FOMF is not experimented because of the required computational effort. Conversely, note that DCMF is able to work on large φ_g , being this an advantage of DCMF over FOMF. By convenience, we execute the explicit version of DCMF, but its quadratic version should return the same state estimates at the cost of a larger T^{CPU} .

5.3 Final Remarks

Our contributions for nonlinear systems were presented over this chapter. In short, three filters (EZGKF, DCMF and FOMF) were proposed to extend predictors based on either mixed uncertainty or zonotopic uncertainty. The new algorithms aim at providing better precision and accuracy in comparison with the extended algorithms (EZGKF (Combastel, 2016), EKF (Särkkä, 2013) and (Wang & Puig, 2016)). Each proposal is based

on a type of approximation and brings up some specific novelties. These possibilities can be exploited by the user according to the convenience. For instance, for unconstrained or LPV systems, EZGKF is a good option to be implemented; for constrained systems, FOMF and DCMF should be preferred. FOMF may return better accuracy while DCMF may demand smaller processing time.

6

Conclusions

6.1 Concluding Remarks

This work aimed at developing linear and nonlinear state estimators with mixed representation of uncertainty. The relevance of this topic, as well as some open issues in the literature, were discussed in Chapter 1. Since mixed uncertainties are here related to stochastic and set-bounded approaches, Kalman and set-based estimators were revisited in Appendices A and B, respectively, to posteriorly present mixed-uncertainty estimators in Appendix C.

Our main contributions were concentrated in Chapters 3-5 by means of new state estimators, which are summarized in Table 6.1. In Chapter 3, some algorithms revisited in Appendices A and B were extended to mitigate possible limitations. For instance, the predictors KP (Kalman, 1960) and ZKP (Combastel, 2015) were extended to the filters DGF and ZKF, respectively, in order to reach a better precision. The DGF is based on the consistency equation (3.3), while the ZKF considers a slight modification on the predictor structure. Aiming at reducing computational cost without significantly deteriorating the precision of DGF and ZKF, we also proposed the fixed-gain algorithms DH₂GF and NH_∞ZF. DRZF exploits the degrees of freedom related to (3.3) to introduce the radius minimization on the estimator structure of DTZF (Tang et al., 2020), yielding appealing improvements of precision. In special, the filters designed under (3.3) provide a direct way to treat descriptor systems through non-descriptor representation. All of these algorithms can be classified as interval because the estimation basis is to correct

Table 6.1: Summary of all state estimators proposed in this thesis. The numeration of each algorithm is presented between brackets in the first column, with our proposals being differed from the existing approaches (just above) through boldface. The remaining columns characterize the algorithms regarding: uncertainty — Gaussian random vector (GRV), zonotope (Zon), constrained zonotope (CZ), strip —, system — state space (SS), descriptor (D), linear time invariant (LTI), linear time varying (LTV) —, approximation — analytical linearization (AL), mean value extension (MV), first-order extension (FO), difference of convex functions (DC) programming, quasi-linear parameter varying (quasi-LPV) —, criterion — minimum trace (MT), minimum radius (MR), H_2 , H_∞ , H_2/H_∞ , segment minimization (SM), volume minimization (VM) —, category — predictor (P), filter (F) —, constraints — equality constraints (EC), inequality constraints (IC) —, and Pareto analysis ($\star$). The proposed state estimators are divided in four groups: (i) stochastic, (ii) set based, (iii) linear mixed, and (iv) nonlinear mixed.

Algorithm	Uncertainty	System	Approx.	Criterion	Cat.	Constraints	Pareto
KP [A.1]	GRV	SS, LTV		MT	P		
DGF [3.1]	GRV	D , LTV		MT	F	EC	
DH₂GF [3.2]	GRV	D , LTI		H₂	F	EC	
DTZF [B.3]	Zon	D, LTV		MT	F	EC	
DRZF [3.3]	Zon	D, LTV		MR	F	EC	
ZKP [B.2]	Zon	SS, LTV		MT	P		
ZKF [3.4]	Zon	SS, LTV		MT	F		
NH_∞ZF [3.5]	Zon	SS, LTI		H_∞	F		
ZMV [B.13]	Zon, Strip	SS	MV	SM, VM1	F		
IZMV [3.6]	Zon, Strip	SS	MV	SM, VM3	F	IC	
CZLF [B.5]	CZ	SS, LTV			F		
CZDC [3.8]	CZ	SS	DC		F	EC, IC	
ZGKP [C.1]	Zon, GRV	SS, LTV		MT	P		
ZGKF [4.1]	Zon, GRV	SS, LTV		MT	F		
DTZF [B.3]	Zon	D, LTV		MT	F	EC	
LTVMF [4.2]	CZ, GRV	D, LTV		MT	F	EC, IC	★
H _∞ ZF [B.4]	Zon	SS, LTI		H _∞	F		
LTIMF [4.4]	CZ, GRV	D , LTI		H₂/H_∞	F	EC, IC	★
EZGKP [C.2]	Zon, GRV	SS	quasi-LPV	MT	P		
EZGKF [5.1]	Zon, GRV	SS	quasi-LPV	MT	F		
ZEKP [B.6]	Zon	SS	AL	MT	P		
FOMF [5.2]	CZ, GRV	SS	FO	MT	F	EC, IC	★
DCMF [5.2]	CZ, GRV	SS	DC	MT	F	EC, IC	★

predictions using a Kalman-like gain applied to a measurement error.

Concerning still Chapter 3, two nonlinear set-based algorithms were developed: IZMV and CZDC. They can be classified as set membership because the estimation basis is to intersect prediction and measurement sets. IZMV is based on the mean value extension and its proposal was to enforce, in principle, state interval constraints on the output sets of the ZMV (Alamo et al., 2005). To achieve that, we used a similar procedure to the usual intersection among strips and zonotopes, which can be optimized via segment or volume

minimization. Since this latter criterion requires online optimization, we also proposed a computational improvement in which the original nonlinear problem was rewritten as an LP. Aiming at practical implementation of discrete-time algorithms, we rose a discussion in Appendix D, not restricted to ZMV, on possibilities and consequences of estimating states for sampled-data continuous-time nonlinear dynamical systems. This content is separate from the main text to promote a better reading experience. CZDC is based on CZs (Scott et al., 2016) and on a DC programming approach (Alamo et al., 2008). CZs were employed to efficiently represent convex polytopes, while DC programming was used to bound the approximation errors. Thereby, we reached an algorithm with competitive or better precision than the state-of-the-art algorithms CZFO and CZMV (Rego et al., 2021). To auxiliary the approximation steps of CZDC, an outer approximation of CZs by parallelotopes was proposed in Chapter 2 (Proposition 2.3.2). CZDC is able to enforce both compact feasible sets and nonlinear equality constraints (called invariants).

Having in mind the prior theoretical advances, novel mixed-uncertainty algorithms were developed in Chapters 4 and 5. To support these developments, new confidence sets were defined in Chapter 2 (Definitions 2.4.3 and 2.4.4). Both ZGKF and EZGKF were our initial contributions, whose motivation was to extend their corresponding predictor version, ZGKP (Combastel, 2015) and EZGKP (Combastel, 2016), to achieve better precision and accuracy. By means of a complexity analysis, we showed that the EZGKF (based on quasi-LPV approach) may imply computational benefits in comparison with the EZGKP (Combastel, 2016). By exploiting the degrees of freedom related to (3.3), two linear mixed-uncertainty filters were proposed: LTVMF and LTIMF. Both algorithms cover descriptor systems and use CZs to represent set-bounded uncertainties. For LTVMF, the main novelties were the proposal of a Pareto analysis to justify the biobjective optimization problems, and the formulation of quadratic and explicit versions to incorporate both state linear equality and inequality constraints. For LTIMF, we also proposed a Pareto analysis to formulate the biobjective problems. Since the cost normalization cannot be employed by LTIMF, we added slack variables to the optimization problem, resulting in a Pareto curve with a better dominance. The idea of the LTIMF is to combine low-cost computation (via rechability analysis) with interval outputs, while the LTVMF is motivated to more general contexts such as LTV systems, state polyhedral constraints, and the combination of costs with different meanings.

Ultimately, two nonlinear mixed-uncertainty filters based on analytical linearization and CZs were proposed: DCMF and FOMF. They were result from advances reached in CZDC, ZGKF, and LTVMF. The difference between those algorithms is the use of either DC programming (DCMF) or first-order extension (FOMF) to approximate the nonlinear models. They bring up many advantages such as cost normalization to combine formulations beyond minimum trace, quadratic and explicit versions, with this latter being motivated to reduce computational burden, and treatment of state polyhedral con-

straints and invariants. Differently from CZDC and CZFO that enforce invariants using generalized intersection, both DCMF and FOMF use augmented measurements. All the mixed-uncertainty filters here proposed were developed under the interval setting since both stochastic and set-based uncertainties can be connected.

In order to clarify the discussions presented over this thesis, many numerical examples were provided. An additional contribution was the development of the worst-case complexity of algorithms revisited in Appendices A-C, whose results were proved in Appendix E.

6.2 Future Directions

Although Section 1.4 points out some advances already reached in this thesis, there exist further issues to be investigated. Therefore, we address the following proposals of continuity.

- In Subsection 3.1.1, the DGF was developed under the mean estimate given by (3.5), whose structure uses the measurements y_{k-1} and y_k . Other possibility is to start from a structure that uses only y_k as in Wang et al. (2018a). From the comparison of precision between ZKF and H_∞ ZF in Example 3.1.7, we expect that the new algorithm reaches a precision better than or equal to the DGF.
- Following the prior idea, we could redesign the steady-state DGF (DH₂GF) using only the measurement y_k . These modifications are also valid to both DTZF and H_∞ ZF revisited in Appendix B, and to propose new mixed-uncertainty filters.
- The NH $_\infty$ ZF (Subsection 3.1.6.2) is formulated under the filter structure. However, it has not been motivated yet due to the lack of case studies that justify the increase of precision and/or accuracy in comparison with H_∞ ZF. The degrees of freedom present in H_∞ ZF motivate the reformulation of the NH $_\infty$ ZF, enabling the algorithm to cover descriptor systems and reducing the current H_∞ norm.
- The method of Adjiman & Floudas (1996) makes Proposition 2.3.3 be computationally costly. In order to mitigate this issue, the choice of lower bounds for eigenvalues could be investigated onto the practical method of Rohn (1998).
- In Section 5.1, the EZGKF was proposed. This algorithm could be extended to enforce equality and inequality constraints on the state vector, which could be an optional via to FOMF and DCMF.
- Motivated by similarities between quasi-LPV and mean value extension approaches, in the sense that the system matrix contains the approximation remainder, those techniques could be combined to yield novel mixed-uncertainty algorithms.

- With the proposal of DRZF in Subsection 3.1.4, an optional cost to minimum trace was introduced to mixed-uncertainty filtering. Thereby, we expect a better reduction of set-bounded uncertainty.
- Novel mixed-uncertainty algorithms can be investigated by assuming that stochastic and interval filters with a similar structure are devoted to estimate states from stochastic and set-bounded parcels, respectively. Next, the design matrices are forced to be equal. This advance is relevant to mitigate connections between mixed-uncertainty state estimation and superposition principle.
- Following the prior idea, stochastic and set-based filters with different structures could be used to estimate states from stochastic and set-based parcels, respectively. However, the most challenge may be the proposal of techniques to mix the state estimators.
- In Subsection 5.2.3.1, the FOMF and DCMF algorithms were compared with stochastic, set-based, and mixed-uncertainty approaches to better motivate the introduction of new mixed-uncertainty methods. Since the differences are not obvious, additional comparisons could be made with linear and/or nonlinear approaches.
- Although state linear inequality constraints were here considered, there exist cases in which the inequality constraints can be nonlinear (Li et al., 2020) and/or temporal (Hu et al., 2022). Therefore, new contributions with mixed-uncertainty algorithms could be introduced in this sense.
- In this thesis, state estimators were presented or proposed for discrete-time dynamical systems. Further contributions could be achieved with the continuous-time case.
- Many algorithms related to the interval observers framework were here revisited or proposed with no stability proof. This investigation could be intended in the future to understand the possible connections with the convex problems derived.
- In order to experiment the state estimators in challenging case studies, the ProVANT Simulator (open access in GitHub) could be employed.

References

- Adjiman, C. S. & Floudas, C. A. (1996). Rigorous convex underestimators for general twice-differentiable problems. *Journal of Global Optimization*, 9(1), 23–40.
- Alamo, T., Bravo, J. M., & Camacho, E. F. (2005). Guaranteed state estimation by Zonotopes. *Automatica*, 41(6), 1035–1043.
- Alamo, T., Bravo, J. M., Redondo, M., & Camacho, E. F. (2008). A set-membership state estimation algorithm based on DC programming. *Automatica*, 44(1), 216–224.
- Althoff, M. (2015). On computing the Minkowski difference of zonotopes. *arXiv preprint arXiv:1512.02794*.
- Althoff, M. & Krogh, B. H. (2011). Zonotope bundles for the efficient computation of reachable sets. In *2011 IEEE 50th Conference on Decision and Control and European Control Conference (CDC-ECC)* (pp. 6814–6821).
- 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.
- Arasaratnam, I. & Haykin, S. (2009). Cubature Kalman filters. *IEEE Transactions on automatic control*, 54(6), 1254–1269.
- Balestrino, A., Caiti, A., & Crisostomi, E. (2006). Particle filtering within a set-membership approach to state estimation. In *2006 IEEE 14th Mediterranean Conference on Control and Automation* (pp. 1–6).
- Bastogne, T., Samson, A., Vallois, P., Wantz-Mézières, S., Pinel, S., Bechet, D., & Barberi-Heyob, M. (2010). Phenomenological modeling of tumor diameter growth based on a mixed effects model. *Journal of theoretical biology*, 262(3), 544–552.
- Beadle, E. R. & Djuric, P. M. (1997). A fast-weighted Bayesian bootstrap filter for nonlinear model state estimation. *IEEE Transactions on Aerospace and Electronic Systems*, 33(1), 338–343.

- Bernstein, D. S. (2009). *Matrix Mathematics: Theory, Facts, and Formulas Ed. 2*. Princeton University Press.
- Bernstein, D. S. & Hyland, D. C. (1993). Compartmental modeling and second-moment analysis of state space systems. *SIAM Journal on Matrix Analysis and Applications*, 14(3), 880–901.
- Bertsimas, D. & Tsitsiklis, J. N. (1997). *Introduction to linear optimization*. Athena Scientific, Belmont, Massachusetts.
- Bo, Z., Kun, Q., Xu-Dong, M., & Xian-Zhong, D. (2013). A new nonlinear set membership filter based on guaranteed bounding ellipsoid algorithm. *Acta Automatica Sinica*, 39(2), 146–154.
- Bolker, E. D. (1969). A class of convex bodies. *Transactions of the American Mathematical Society*, 145, 323–345.
- Boyd, S., Boyd, S. P., & Vandenberghe, L. (2004). *Convex optimization*. Cambridge university press.
- Boyd, S., El Ghaoui, L., Feron, E., & Balakrishnan, V. (1994). *Linear matrix inequalities in system and control theory*, volume 15. Siam.
- Bravo, J., Alamo, T., & Camacho, E. (2004). Bounded error identification of systems with time-varying parameters. In *2004 IEEE 43rd Conference on Decision and Control (CDC)*, volume 4 (pp. 4063–4068).
- 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.
- Chabane, S. B., Maniu, C. S., Alamo, T., Camacho, E. F., & Dumur, D. (2014). Improved set-membership estimation approach based on zonotopes and ellipsoids. In *2014 European Control Conference (ECC)* (pp. 993–998).: IEEE.
- Chen, C. T. (1999). *Linear Systems Theory and Design*. New York: Oxford.
- Chen, G., Wang, J., & Shieh, L. S. (1997). Interval Kalman filtering. *IEEE Transactions on Aerospace and Electronic Systems*, 33(1), 250–259.
- Chisci, L., Garulli, A., & Zappa, G. (1996). Recursive state bounding by parallelotopes. *Automatica*, 32(7), 1049–1055.
- Combastel, C. (2003). A state bounding observer based on zonotopes. In *2003 European Control Conference (ECC)* (pp. 2589–2594).: IEEE.

- Combastel, C. (2005). A state bounding observer for uncertain non-linear continuous-time systems based on zonotopes. In *2005 IEEE 44th Conference on Decision and Control* (pp. 7228–7234).
- Combastel, C. (2015). Merging Kalman filtering and zonotopic state bounding for robust fault detection under noisy environment. *IFAC-PapersOnLine*, 48(21), 289–295.
- Combastel, C. (2015). Zonotopes and Kalman observers: Gain optimality under distinct uncertainty paradigms and robust convergence. *Automatica*, 55, 265–273.
- Combastel, C. (2016). An Extended Zonotopic and Gaussian Kalman Filter (EZGKF) merging set-membership and stochastic paradigms: Toward non-linear filtering and fault detection. *Annual Reviews in Control*, 42, 232–243.
- De Oliveira, M. C., Bernussou, J., & Geromel, J. C. (1999). A new discrete-time robust stability condition. *Systems & control letters*, 37(4), 261–265.
- de Paula, A. A. (2018). Estimação de estados com restrições intervalares utilizando abordagens estocástica e zonotópica. Master’s thesis, Universidade Federal de Minas Gerais.
- de Paula, A. A., Campos, V. C. d. S., Raffo, G. V., & Teixeira, B. O. S. (2020). Novel Gaussian state estimator based on H_2 norm and steady-state variance. In *XXIII Congresso Brasileiro de Automática, 2020, Porto Alegre* (pp. 1–7).: Anais do XXIII Congresso Brasileiro de Automática.
- de Paula, A. A., Raffo, G. V., & Teixeira, B. O. S. (2018). State estimation based on stochastic and zonotopic approaches: part II - nonlinear systems. In *XXII Congresso Brasileiro de Automática, 2018, João Pessoa* (pp. 1–8).: Anais do XXII Congresso Brasileiro de Automática.
- de Paula, A. A., Raffo, G. V., & Teixeira, B. O. S. (2019). Zonotopic and Gaussian state estimator: the predictor and filter algorithms. In *14o Simpósio Brasileiro de Automação Inteligente, 2019, Ouro Preto* (pp. 1–6).: Anais do 14o Simpósio Brasileiro de Automação Inteligente.
- de Paula, A. A., Raffo, G. V., & Teixeira, B. O. S. (2022a). Extended Mixed Filtering based on Zonotopic and Gaussian Uncertainties for Discrete-Time Nonlinear Systems. Submitted.
- de Paula, A. A., Raffo, G. V., & Teixeira, B. O. S. (2022b). Minimum-Radius Criterion for a Zonotopic State Estimator based on Degrees of Freedom. To be submitted.
- de Paula, A. A., Raffo, G. V., & Teixeira, B. O. S. (2022c). Zonotopic Filtering for Uncertain Nonlinear Systems: Fundamentals, Implementation Aspects, and Extensions [Applications of Control]. *IEEE Control Systems*, 42(1), 19–51.

- de Paula, A. A., Raimondo, D. M., Raffo, G. V., & Teixeira, B. O. S. (2022d). Set-based state estimation for discrete-time constrained nonlinear systems: an approach based on constrained zonotopes and DC programming. Submitted.
- de Paula et al. (2022a). Mixed Nonlinear State Estimators with Linear Approximations. To be submitted.
- de Paula et al. (2022b). State Estimators for Discrete-Time Descriptor Linear Systems with Mixed Uncertainties and State Constraints. To be submitted.
- Di Marco, M., Garulli, A., Giannitrapani, A., & Vicino, A. (2004). A set theoretic approach to dynamic robot localization and mapping. *Autonomous robots*, 16(1), 23–47.
- D’Jorge, A., Santoro, B. F., Anderson, A., González, A. H., & Ferramosca, A. (2020). Stochastic model predictive control for tracking linear systems. *Optimal Control Applications and Methods*, 41(1), 65–83.
- Durieu, C., Polyak, B. T., & Walter, E. (1996). Trace versus determinant in ellipsoidal outer-bounding, with application to state estimation. *IFAC Proceedings Volumes*, 29(1), 3975–3980.
- Dyer, M. E. (1983). The complexity of vertex enumeration methods. *Mathematics of Operations Research*, 8(3), 381–402.
- Efimov, D., Perruquetti, W., Raïssi, T., & Zolghadri, A. (2013). On interval observer design for time-invariant discrete-time systems. In *2013 European Control Conference (ECC)* (pp. 2651–2656).: IEEE.
- García, R., Orihuela, L., Millán, P., Rubio, F., & Ortega, M. (2020). Guaranteed estimation and distributed control of vehicle formations. *International Journal of Control*, (pp. 1–14).
- Ghosh, M., Wu, L., Hao, Q., & Zhou, Q. (2022). A random forest with multi-fidelity gaussian process leaves for modeling multi-fidelity data with heterogeneity. *Computers & Industrial Engineering*, 174, 108746.
- Girard, A. (2005). Reachability of uncertain linear systems using zonotopes. In *International Workshop on Hybrid Systems: Computation and Control* (pp. 291–305).: Springer.
- Golub, G. H. & Van Loan, C. F. (2013). *Matrix computations*. JHU press.
- Grant, M. & Boyd, S. (2014). CVX: Matlab software for disciplined convex programming, version 2.1. <http://cvxr.com/cvx>.

- Grocholsky, B., Stump, E., & Kumar, V. (2006). An extensive representation for range-only slam. In *International Symposium on Experimental Robotics, 2006, Rio de Janeiro*.
- Gustafsson, F. & Hendeby, G. (2011). Some relations between extended and unscented Kalman filters. *IEEE Transactions on Signal Processing*, 60(2), 545–555.
- Hanebeck, U. D. & Horn, J. (2001). New estimators for mixed stochastic and set theoretic uncertainty models: the general case. In *2001 American Control Conference (ACC)*, volume 6 (pp. 5040–5045).
- Henningsson, T. (2008). Recursive state estimation for linear systems with mixed stochastic and set-bounded disturbances. In *2008 47th IEEE Conference on Decision and Control (CDC)* (pp. 678–683).
- Herceg, M., Kvasnica, M., Jones, C., & Morari, M. (2013). Multi-parametric toolbox 3.0. In *Proc. of the European Control Conference* (pp. 502–510). Zürich, Switzerland. <http://control.ee.ethz.ch/~mpt>.
- Hertz, D. (1992). The extreme eigenvalues and stability of real symmetric interval matrices. *IEEE Transactions on Automatic Control*, 37(4), 532–535.
- Horst, R. & Thoai, N. V. (1999). DC programming: overview. *Journal of Optimization Theory and Applications*, 103(1), 1–43.
- Hu, C., Liang, Y., & Xu, L. (2022). Hybrid sampling-based particle filtering with temporal constraints. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*.
- Ifqir, S., Puig, V., Ichalal, D., Ait-Oufroukh, N., & Mammar, S. (2020). Zonotopic set-membership estimation for switched systems based on wi-radius minimization: Vehicle application. *IFAC-PapersOnLine*, 53(2), 7446–7451.
- Jaulin, L. (2009). Robust set-membership state estimation; application to underwater robotics. *Automatica*, 45(1), 202–206.
- Jaulin, L. & Kieffer, M. (2001). *Applied Interval Analysis*. Springer.
- Jazwinski, A. H. (2007). *Stochastic processes and filtering theory*. Courier Corporation.
- Julier, S. J. & Uhlmann, J. K. (1997). New extension of the Kalman filter to nonlinear systems. In *Signal processing, sensor fusion, and target recognition VI*, volume 3068 (pp. 182–194).: International Society for Optics and Photonics.
- Julier, S. J. & Uhlmann, J. K. (2004). Unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, 92(3), 401–422.

- Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 82(1), 35–45.
- Kay, S. M. (1993). *Fundamentals of statistical signal processing*. Prentice Hall PTR.
- Kelner, J. A. & Spielman, D. A. (2006). A randomized polynomial-time simplex algorithm for linear programming. In *Proceedings of the thirty-eighth annual ACM symposium on Theory of computing* (pp. 51–60).
- Khargonekar, P. P., Rotea, M. A., & Baeyens, E. (1996). Mixed H_2/H_∞ filtering. *International Journal of Robust and Nonlinear Control*, 6(4), 313–330.
- Ko, S. & Bitmead, R. R. (2005). State estimation of linear systems with state equality constraints. *IFAC Proceedings Volumes*, 38(1), 241–246.
- Kühn, W. (1998). Rigorously computed orbits of dynamical systems without the wrapping effect. *Computing*, 61(1), 47–67.
- Le, V. T. H., Alamo, T., Camacho, E. F., Stoica, C., & Dumur, D. (2012). Zonotopic set-membership estimation for interval dynamic systems. In *2012 American Control Conference (ACC)* (pp. 6787–6792).: IEEE.
- Le, V. T. H., Stoica, C., Alamo, T., Camacho, E. F., & Dumur, D. (2013a). Zonotope-based set-membership estimation for multi-output uncertain systems. In *IEEE International Symposium on Intelligent Control (ISIC)* (pp. 212–217).
- Le, V. T. H., Stoica, C., Alamo, T., Camacho, E. F., & Dumur, D. (2013b). *Zonotopes: From guaranteed state-estimation to control*. John Wiley & Sons.
- Le, V. T. H., Stoica, C., Alamo, T., Camacho, E. F., & Dumur, D. (2013c). Zonotopic guaranteed state estimation for uncertain systems. *Automatica*, 49(11), 3418–3424.
- Lefebvre*, T., Bruyninckx, H., & De Schutter, J. (2004). Kalman filters for non-linear systems: a comparison of performance. *International journal of Control*, 77(7), 639–653.
- Li, S., Stouraitis, T., Gienger, M., Vijayakumar, S., & Shah, J. A. (2021). Set-based state estimation with probabilistic consistency guarantee under epistemic uncertainty. *arXiv preprint arXiv:2110.09584*.
- Li, X., Yuan, X., Meng, F., Liao, Y., Liuc, H., Zhang, X., & Shen, X. (2020). Set-membership filtering for nonlinear dynamic systems with quadratic inequality constraints. *IEEE Access*, 8, 13375–13386.
- Marseglia, G. R., Scott, J. K., Magni, L., Braatz, R. D., & Raimondo, D. M. (2014). A hybrid stochastic-deterministic approach for active fault diagnosis using scenario optimization. *IFAC Proceedings Volumes*, 47(3), 1102–1107.

- Martinez, J. J., Loukkas, N., & Meslem, N. (2020). H-infinity set-membership observer design for discrete-time lpv systems. *International journal of control*, 93(10), 2314–2325.
- Mastronardi, N. & Dooren, P. V. (2017). Computing the jordan structure of an eigenvalue. *SIAM Journal on Matrix Analysis and Applications*, 38(3), 949–966.
- Menegaz, H. M., Ishihara, J. Y., Borges, G. A., & Vargas, A. N. (2015). A systematization of the unscented Kalman filter theory. *IEEE Transactions on Automatic Control*, 60(10), 2583–2598.
- Merrikh Bayat, F., Hoskins, B., & Strukov, D. B. (2015). Phenomenological modeling of memristive devices. *Applied Physics A*, 118(3), 779–786.
- Mesbah, A. (2016). Stochastic model predictive control: An overview and perspectives for future research. *IEEE Control Systems Magazine*, 36(6), 30–44.
- Meslem, N., Loukkas, N., & Martinez, J. (2018). Using set invariance to design robust interval observers for discrete-time linear systems. *International Journal of Robust and Nonlinear Control*, 28(11), 3623–3639.
- Meslem, N. & Martinez, J. J. (2019). Hybrid interval-ellipsoidal set-membership state estimators for uncertain discrete-time linear systems. In *2019 IEEE 4th Conference on Control and Fault Tolerant Systems (SysTol)* (pp. 348–353).
- Mistler, V., Benallegue, A., & M’sirdi, N. (2001). Exact linearization and noninteracting control of a 4 rotors helicopter via dynamic feedback. In *2001 IEEE 10th International Workshop on Robot and Human Interactive Communication* (pp. 586–593).
- Moore, R. E. (1966). *Interval analysis*. Prince-Hall, Englewood Cliffs, NJ.
- Moore, R. E., Kearfott, R. B., & Cloud, M. J. (2009). *Introduction to interval analysis*. SIAM.
- Morrell, D. R. & Stirling, W. C. (1991). Set-values filtering and smoothing. *IEEE Transactions on Systems, Man, and Cybernetics*, 21(1), 184–193.
- Nair, G. N. (2013). A nonstochastic information theory for communication and state estimation. *IEEE Transactions on Automatic Control*, 58(6), 1497–1510.
- Noack, B., Pfaff, F., & Hanebeck, U. D. (2012). Optimal Kalman gains for combined stochastic and set-membership state estimation. In *2012 IEEE 51st Annual Conference on Decision and Control (CDC)* (pp. 4035–4040).
- Orihuela, L., Millán, P., Roshany-Yamchi, S., & Garcia, R. A. (2018). Negotiated distributed estimation with guaranteed performance for bandwidth-limited situations. *Automatica*, 87, 94–102.

- Ozbay, H. & Bachmann, G. R. (1994). h^2/h^∞ controller design for a two-dimensional thin airfoil flutter suppression. *Journal of Guidance, Control, and Dynamics*, 17(4), 722–728.
- Pan, Z., Luan, X., & Liu, F. (2022). Confidence set-membership state estimation for lpv systems with inexact scheduling variables. *ISA Transactions*, 122, 38–48.
- Peherstorfer, B., Kramer, B., & Willcox, K. (2017). Combining multiple surrogate models to accelerate failure probability estimation with expensive high-fidelity models. *Journal of Computational Physics*, 341, 61–75.
- Pourasghar, M., Puig, V., & Ocampo-Martinez, C. (2019). Interval observer versus set-membership approaches for fault detection in uncertain systems using zonotopes. *International Journal of Robust and Nonlinear Control*, 29(10), 2819–2843.
- Prakash, J., Patwardhan, S. C., & Shah, S. L. (2010). Constrained nonlinear state estimation using ensemble Kalman filters. *Industrial & Engineering Chemistry Research*, 49(5), 2242–2253.
- Rapaport, A. & Gouzé, J. (2003). Parallelotopic and practical observers for non-linear uncertain systems. *International journal of control*, 76(3), 237–251.
- 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 *2018 IEEE Conference on Decision and Control (CDC)* (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 *2018 European Control Conference (ECC)* (pp. 1584–1589).: IEEE.
- 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. *IFAC-PapersOnLine*, 53(2), 4291–4296.
- Rego, B. S., Scott, J. K., Raimondo, D. M., & Raffo, G. V. (2021). Set-valued state estimation of nonlinear discrete-time systems with nonlinear invariants based on constrained zonotopes. *Automatica*, 129, 109638.

- Rengaswamy, R., Narasimhan, S., & Kuppuraj, V. (2013). Receding-horizon nonlinear Kalman (RNK) filter for state estimation. *IEEE Transactions on Automatic Control*, 58(8), 2054–2059.
- Ristic, B., Arulampalam, S., & Gordon, N. (2004). *Beyond the Kalman filter: Particle filters for tracking applications*. Artech house.
- Rohn, J. (1998). Bounds on eigenvalues of interval matrices. *ZAMM-Zeitschrift für Angewandte Mathematik und Mechanik*, 78(3), S1049.
- Rump, S. (1999). INTLAB - INTerval LABoratory. In T. Csendes (Ed.), *Developments in Reliable Computing* (pp. 77–104). Dordrecht: Kluwer Academic Publishers. <http://www.ti3.tuhh.de/rump/>.
- Särkkä, S. (2013). *Bayesian filtering and smoothing*, volume 3. Cambridge University Press.
- 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., Marseglia, G. R., Magni, L., Braatz, R. D., & Raimondo, D. M. (2013). A hybrid stochastic-deterministic input design method for active fault diagnosis. In *2013 52nd IEEE Conference on Decision and Control (CDC)* (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.
- Shimada, N., Shirai, Y., Kuno, Y., & Miura, J. (1998). Hand gesture estimation and model refinement using monocular camera-ambiguity limitation by inequality constraints. In *1998 IEEE 3rd International Conference on Automatic Face and Gesture Recognition* (pp. 268–273).
- Simon, D. (2006). *Optimal state estimation: Kalman, H infinity, and nonlinear approaches*. John Wiley & Sons.
- Simon, D. (2010). Kalman filtering with state constraints: a survey of linear and nonlinear algorithms. *IET Control Theory & Applications*, 4(8), 1303–1318.
- Simon, D. & Simon, D. L. (2010). Constrained Kalman filtering via density function truncation for turbofan engine health estimation. *International Journal of Systems Science*, 41(2), 159–171.

- Spletzer, J. R. & Taylor, C. J. (2003). A bounded uncertainty approach to multi-robot localization. In *2003 IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 2 (pp. 1258–1265).
- Sun, L., Alkhatib, H., Kargoll, B., Kreinovich, V., & Neumann, I. (2019). Ellipsoidal and Gaussian Kalman Filter Model for Discrete-Time Nonlinear Systems. *Mathematics*, 7(12), 1168.
- Tang, C., Liu, C., Yuan, L., & Xing, Z. (2016). High precision low complexity matrix inversion based on Newton iteration for data detection in the massive MIMO. *IEEE Communications Letters*, 20(3), 490–493.
- Tang, W., Wang, Z., & Shen, Y. (2019a). Interval estimation for discrete-time linear systems: a two-step method. *Systems & Control Letters*, 123, 69–74.
- Tang, W., Wang, Z., Wang, Y., Raïssi, T., & Shen, Y. (2019b). Interval estimation methods for discrete-time linear time-invariant systems. *IEEE Transactions on Automatic Control*, 64(11), 4717–4724.
- Tang, W., Wang, Z., Zhang, Q., & Shen, Y. (2020). Set-membership estimation for linear time-varying descriptor systems. *Automatica*, 115, 108867.
- Tao, P. D. & An, L. T. H. (1997). Convex analysis approach to DC programming: theory, algorithms and applications. *Acta mathematica vietnamica*, 22(1), 289–355.
- Teixeira, B. (2008). Kalman filters [ask the experts]. *IEEE Control Systems Magazine*, 28(2), 16–18.
- Teixeira, B. O. S., Chandrasekar, J., Palanthandalam-Madapusi, H. J., Tôrres, L. A. B., Aguirre, L. A., & Bernstein, D. S. (2008a). Gain-constrained Kalman filtering for linear and nonlinear systems. *IEEE Transactions on Signal Processing*, 56(9), 4113–4123.
- 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.
- Teixeira, B. O. S., Tôrres, L. A. B., Aguirre, L. A., & Bernstein, D. S. (2008b). Unscented filtering for interval-constrained nonlinear systems. In *2008 IEEE 47th Conference on Decision and Control* (pp. 5116–5121).
- Teixeira, B. O. S., Tôrres, L. A. B., Aguirre, L. A., & Bernstein, D. S. (2010). On unscented Kalman filtering with state interval constraints. *Journal of Process Control*, 20(1), 45–57.

- Tottoli, A., de Paula, A. A., Raffo, G. V., Teixeira, B. O. S., & Raimondo, D. M. (2022). Set-based state estimation of a Li-ion cell using DC programming and constrained zonotopes. Submitted.
- Vachhani, P., Narasimhan, S., & Rengaswamy, R. (2006). Robust and reliable estimation via unscented recursive nonlinear dynamic data reconciliation. *Journal of process control*, 16(10), 1075–1086.
- 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.
- Villanueva, M. E. & Houska, B. (2020). On stochastic linear systems with zonotopic support sets. *Automatica*, 111, 108652.
- Wang, Y. & Puig, V. (2016). Zonotopic extended Kalman filter and fault detection of discrete-time nonlinear systems applied to a quadrotor helicopter. In *2016 IEEE 3rd Conference on Control and Fault-Tolerant Systems (SysTol)* (pp. 367–372).
- 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., Wang, Z., Puig, V., & Cembrano, G. (2018b). Zonotopic set-membership state estimation for discrete-time descriptor lpv systems. *IEEE Transactions on Automatic Control*, 64(5), 2092–2099.
- Wang, Z., Lim, C.-C., & Shen, Y. (2018c). Interval observer design for uncertain discrete-time linear systems. *Systems & Control Letters*, 116, 41–46.
- Wolf, A., Swift, J. B., Swinney, H. L., & Vastano, J. A. (1985). Determining Lyapunov exponents from a time series. *Physica D: Nonlinear Phenomena*, 16(3), 285–317.
- Xiong, K., Zhang, H., & Chan, C. (2006). Performance evaluation of UKF-based nonlinear filtering. *Automatica*, 42(2), 261–270.
- Xu, S. & Lam, J. (2006). *Robust control and filtering of singular systems*. Springer.
- Yang, X. & Scott, J. K. (2018). A comparison of zonotope order reduction techniques. *Automatica*, 95, 378–384.



Stochastic State Estimators: a Review

This appendix contains two sections related to the stochastic setting. Linear algorithms are presented in the first section (as the Kalman filter), while nonlinear algorithms are revisited in the posterior section. The nonlinear algorithms employ the unscented transformation (UT) to propagate Gaussian random vectors (GRVs) over nonlinear mapping using sigma points. In order to fulfill state interval constraints, two approaches called PDF truncation (PT) and interval-constrained unscented transformation (ICUT) are also reviewed. Both PT and ICUT apply a correction to the mean and the covariance of GRVs according to the interval constraints. In fact, these methods aim to correct the probability regions, not bounding the GRVs, since their realizations can assume any real value by definition. On the one hand, the PT works on bilaterally truncated GRVs to concentrate the most likely region into the constraints. On the other hand, the ICUT computes sigma points with different weights, in order to attribute a smaller importance to vectors that exceed the constraints. In order to better characterize the algorithms, we also derived their worst-case complexity order.

A.1 Linear Algorithms

Next, two linear state estimators based on GRVs are presented. Consider the discrete-time uncertain linear dynamical system

$$x_k = A_{k-1}x_{k-1} + B_{k-1}u_{k-1} + w_{k-1}, \quad (\text{A.1})$$

$$y_k = C_k x_k + v_k, \quad (\text{A.2})$$

where $A_{k-1} \in \mathbb{R}^{n \times n}$, $B_{k-1} \in \mathbb{R}^{n \times p}$, and $C_k \in \mathbb{R}^{m \times n}$ are the system matrices, $u_{k-1} \in \mathbb{R}^p$ is the input vector, $y_k \in \mathbb{R}^m$ is the output vector, $w_{k-1} \in \mathbb{R}^n$ is the process noise, $v_k \in \mathbb{R}^m$ is the measurement noise, and $x_k \in \mathbb{R}^n$ is the state vector. We assume that the system matrices and the input and output vectors are known over $k \in \mathbb{Z}_+$. The terms x_0 , w_{k-1} , and v_k are realizations of the GRVs $\hat{X}_0 \sim \mathcal{N}(\hat{x}_0, \hat{P}_0^{\text{xx}})$, $W_{k-1} \sim \mathcal{N}(0_{n \times 1}, Q_{k-1})$, and $V_k \sim \mathcal{N}(0_{m \times 1}, R_k)$, respectively. In addition, we assume that $\hat{X}_0$ is uncorrelated to W_{k-1} and V_k . For completeness, we assume that W_{k-1} and V_k compose stochastic processes that satisfy $\mathbb{E}[W_k W_j^\top] = 0_{n \times n}$ and $\mathbb{E}[V_k V_j^\top] = 0_{m \times m}$ for all $k \neq j$.

The goal is to solve Problem 1.2.1 to obtain the *a posteriori* mean $\hat{x}_k$ and covariance $\hat{P}_k^{\text{xx}}$ estimates. According to the minimum-variance criterion (1.5), the Kalman gain $K_k \in \mathbb{R}^{n \times m}$ will be determined based on a predictor/corrector structure.

A.1.1 Kalman Predictor

For (A.1)-(A.2), we have the following mean estimates:

$$\hat{x}_{k|k-1} = A_{k-1} \hat{x}_{k-1} + B_{k-1} u_{k-1}, \quad (\text{A.3})$$

$$\hat{y}_{k-1} = C_{k-1} \hat{x}_{k-1}. \quad (\text{A.4})$$

Therefore, the following state estimate is designed (Kalman, 1960):

$$\hat{x}_k = \hat{x}_{k|k-1} + K_{k-1} (y_{k-1} - \hat{y}_{k-1}). \quad (\text{A.5})$$

By defining the estimation error as $e_k \triangleq x_k - \hat{x}_k$, we have

$$e_k = (A_{k-1} - K_{k-1} C_{k-1}) e_{k-1} + w_{k-1} - K_{k-1} v_{k-1}. \quad (\text{A.6})$$

Let $\hat{X}_k^g \sim \mathcal{N}(0_{n \times 1}, \hat{P}_k^{\text{xx}})$ represent e_k . Then, replacing each realization by its GRV leads to

$$\hat{X}_k^g = (A_{k-1} - K_{k-1} C_{k-1}) \hat{X}_{k-1}^g + W_{k-1} - K_{k-1} V_{k-1}, \quad (\text{A.7})$$

whose covariance matrix is given by

$$\hat{P}_k^{\text{xx}} = (A_{k-1} - K_{k-1} C_{k-1}) \hat{P}_{k-1}^{\text{xx}} (A_{k-1} - K_{k-1} C_{k-1})^\top + Q_{k-1} + K_{k-1} R_{k-1} K_{k-1}^\top. \quad (\text{A.8})$$

To find the optimal gain $\check{K}_{k-1}$, we carry out $\frac{\partial \text{tr}(\hat{P}_k^{\text{xx}}(K_{k-1}))}{\partial K_{k-1}} = 0$, yielding

$$\check{K}_{k-1} = \hat{P}_{k-1}^{\text{xy}} \left(\hat{P}_{k-1}^{\text{yy}} \right)^{-1}, \quad (\text{A.9})$$

where

$$\hat{P}_{k-1}^{xy} = A_{k-1} \hat{P}_{k-1}^{xx} C_{k-1}^\top, \quad (\text{A.10})$$

$$\hat{P}_{k-1}^{yy} = C_{k-1} \hat{P}_{k-1}^{xx} C_{k-1}^\top + R_{k-1}. \quad (\text{A.11})$$

By convenience, we rewrite the *a posteriori* covariance matrix (A.8) as (Combastel, 2015)

$$\hat{P}_k^{xx} = \hat{P}_{k|k-1}^{xx} - \check{K}_{k-1} \left(\hat{P}_{k-1}^{xy} \right)^\top, \quad (\text{A.12})$$

where

$$\hat{P}_{k|k-1}^{xx} = A_{k-1} \hat{P}_{k-1}^{xx} A_{k-1}^\top + Q_{k-1}. \quad (\text{A.13})$$

The mean $\hat{x}_{k|k-1}$ and covariance $\hat{P}_{k|k-1}^{xx}$ estimates are called *a priori* data, since they are related to the process model (A.1). The mean $\hat{x}_k$ and covariance $\hat{P}_k^{xx}$ estimates are called *a posteriori* data, since the *a priori* data are corrected by the measurement data.

Next, we present the Kalman predictor (KP) in Algorithm A.1 with two steps, namely: (i) *forecast step*; and (ii) *data-assimilation step*. This algorithm solves Problem 1.2.1 disregarding state constraints. Its complexity order is given by

$$O(\text{KP}) = O(n^3 + m^3 + np + n^2m + nm^2).$$

Algorithm A.1: $\hat{X}_k = \text{KP}(\hat{X}_{k-1}, A_{k-1}, B_{k-1}, W_{k-1}, C_{k-1}, V_{k-1}, u_{k-1}, y_{k-1})$ (Kalman, 1960)

- 1 Compute the mean estimates $\hat{x}_{k|k-1}$ in (A.3) and $\hat{y}_{k-1}$ in (A.4), and the covariance estimates $\hat{P}_{k|k-1}^{xx}$ in (A.13), $\hat{P}_{k-1}^{xy}$ in (A.10), and $\hat{P}_{k-1}^{yy}$ in (A.11)
 - 2 Compute the Kalman gain (A.9), and thereby, the *a posteriori* mean $\hat{x}_k$ and covariance $\hat{P}_k^{xx}$ estimates given by (A.5) and (A.12), respectively, to obtain $\hat{X}_k \sim \mathcal{N}(\hat{x}_k, \hat{P}_k^{xx})$
-

A.1.2 Kalman Filter

For (A.1)-(A.2), we have the mean estimates $\hat{x}_{k|k-1}$ in (A.3) and

$$\hat{y}_{k|k-1} = C_k \hat{x}_{k|k-1}. \quad (\text{A.14})$$

Therefore, the following state estimate is designed (Kalman, 1960):

$$\hat{x}_k = \hat{x}_{k|k-1} + K_k (y_k - \hat{y}_{k|k-1}). \quad (\text{A.15})$$

From the estimation error $e_k \triangleq x_k - \hat{x}_k$, we have

$$e_k = (I_n - K_k C_k) (A_{k-1} e_{k-1} + w_{k-1}) - K_k v_k. \quad (\text{A.16})$$

Let $\hat{X}_k^g \sim \mathcal{N}(0_{n \times 1}, \hat{P}_k^{xx})$ represent e_k . Then, replacing each realization by its GRV leads to

$$\hat{X}_k^g = (\mathbf{I}_n - K_k C_k) (\hat{X}_{k-1}^g + W_{k-1}) - K_k V_k, \quad (\text{A.17})$$

whose covariance matrix is given by

$$\hat{P}_k^{xx} = (\mathbf{I}_n - K_k C_k) \hat{P}_{k|k-1}^{xx} (\mathbf{I}_n - K_k C_k)^\top + K_k R_k K_k^\top. \quad (\text{A.18})$$

with $\hat{P}_{k|k-1}^{xx}$ being given by (A.13).

To find the optimal gain $\check{K}_k$, we carry out $\frac{\partial \text{tr}(\hat{P}_k^{xx}(K_k))}{\partial K_k} = 0$, yielding

$$\check{K}_k = \hat{P}_{k|k-1}^{xy} (\hat{P}_{k|k-1}^{yy})^{-1}, \quad (\text{A.19})$$

where

$$\hat{P}_{k|k-1}^{xy} = \hat{P}_{k|k-1}^{xx} C_k^\top, \quad (\text{A.20})$$

$$\hat{P}_{k|k-1}^{yy} = C_k \hat{P}_{k|k-1}^{xx} C_k^\top + R_k. \quad (\text{A.21})$$

We may rewrite the *a posteriori* covariance matrix (A.18) as (Teixeira, 2008)

$$\hat{P}_k^{xx} = \hat{P}_{k|k-1}^{xx} - \check{K}_k \hat{P}_{k|k-1}^{yy} \check{K}_k^\top. \quad (\text{A.22})$$

Next, we recast the equations above as the Kalman filter (KF) in Algorithm A.2. This algorithm solves Problem 1.2.1 disregarding state constraints. Its complexity order is given by

$$O(\text{KF}) = O(n^3 + m^3 + np + n^2m + nm^2).$$

Note that $O(\text{KF}) = O(\text{KP})$.

Algorithm A.2: $\hat{X}_k = \text{KF}(\hat{X}_{k-1}, A_{k-1}, B_{k-1}, W_{k-1}, C_k, V_k, u_{k-1}, y_k)$ (Kalman, 1960)

- 1 Compute the mean estimates $\hat{x}_{k|k-1}$ in (A.3) and $\hat{y}_{k|k-1}$ in (A.14), and the covariance estimates $\hat{P}_{k|k-1}^{xx}$ in (A.13), $\hat{P}_{k|k-1}^{xy}$ in (A.20), and $\hat{P}_{k|k-1}^{yy}$ in (A.21)
 - 2 Compute the Kalman gain (A.19), and thereby, the *a posteriori* center $\hat{x}_k$ and covariance $\hat{P}_k^{xx}$ estimates given by (A.15) and (A.22), respectively, to obtain $\hat{X}_k \sim \mathcal{N}(\hat{x}_k, \hat{P}_k^{xx})$
-

A.1.3 Stationary-Gain Kalman Estimators

Next, we present the stationary version of the KP (SKP) and KF (SKF) algorithms, whose continuous version is discussed in Khargonekar et al. (1996). Consider that (A.1)-(A.2) depict an LTI dynamical system. Therefore, the KP can be reduced to the SKP,

whose estimates of mean, error, and covariance are, respectively, given by

$$\hat{x}_k = (A - KC) \hat{x}_{k-1} + Bu_{k-1} + Ky_{k-1}, \quad (\text{A.23})$$

$$e_k = (A - KC) e_{k-1} + F^w v_{k-1}^w - KF^v v_{k-1}^v, \quad (\text{A.24})$$

$$\hat{P}_k^{xx} = (A - KC) \hat{P}_{k-1}^{xx} (A - KC)^\top + Q + KRK^\top, \quad (\text{A.25})$$

where F^w and F^v are the Cholesky decomposition of covariance matrices Q and R , $v_{k-1}^w \in \mathbb{R}^n$ and $v_{k-1}^v \in \mathbb{R}^m$ are the realizations of zero-mean GRVs with identity covariance, and $K \in \mathbb{R}^{n \times m}$ is the constant gain to be determined. By relating (A.24) to Lemma 2.5.2, we have $\check{A} = (A - KC)$, $\check{B} = [F^w \quad -KF^v]$, and $\check{C} = I_n$. In doing so, the following result is derived.

Corollary A.1.1 (Adapted from Khargonekar et al. (1996)). Consider a parameter $\psi > 0$ and matrices $P = P^\top > 0_{n \times n}$, $Y_2 \in \mathbb{R}^{n \times m}$, and $Y_4 = Y_4^\top > 0_{m \times m}$. Consider also the LMIs

$$\text{tr}((F^w)^\top P F^w) + \text{tr}((F^v)^\top Y_4 F^v) < \psi, \quad (\text{A.26})$$

$$\begin{bmatrix} -Y_4 & Y_2^\top \\ Y_2 & -P \end{bmatrix} < 0_{(n+m) \times (n+m)}, \quad (\text{A.27})$$

$$\begin{bmatrix} I_n - P & A^\top P - C^\top Y_2^\top \\ PA - Y_2 C & -P \end{bmatrix} < 0_{2n \times 2n}. \quad (\text{A.28})$$

By solving the constrained optimization problem

$$\min \psi \quad \text{s.t.} \quad (\text{A.26})-(\text{A.28}), \quad (\text{A.29})$$

we obtain the optimal matrix $\check{K} = P^{-1}Y_2$.

Let the instant k be enough large to assume the steady state. Therefore, the KF can be reduced to the SKF, whose estimates of mean, error, and covariance are, respectively, given by

$$\hat{x}_k = (I_{n \times n} - KC) (A \hat{x}_{k-1} + Bu_{k-1}) + Ky_k, \quad (\text{A.30})$$

$$e_k = (I_{n \times n} - KC) (A e_{k-1} + F^w v_{k-1}^w) - KF^v v_k^v, \quad (\text{A.31})$$

$$\hat{P}_k^{xx} = (I_{n \times n} - KC) (A \hat{P}_{k-1}^{xx} A^\top + Q) (I_{n \times n} - KC)^\top + KRK^\top, \quad (\text{A.32})$$

with the constant gain $K \in \mathbb{R}^{n \times m}$ to be found. By relating (A.31) to Lemma 2.5.2, we have $\check{A} = (A - KCA)$, $\check{B} = [(F^w - KCF^w) \quad -KF^v]$, and $\check{C} = I_n$. In doing so, the following result is derived.

Corollary A.1.2 (Adapted from Khargonekar et al. (1996)). Consider a parameter $\psi > 0$ and matrices $P = P^\top > 0_{n \times n}$, $Y_2 \in \mathbb{R}^{n \times m}$, and $Y_4 = Y_4^\top > 0_{m \times m}$. Consider also the LMIs

$$\text{tr}((F^w)^\top M F^w) + \text{tr}((F^v)^\top Y_4 F^v) < \psi, \quad (\text{A.33})$$

$$\begin{bmatrix} -Y_4 & Y_2^\top \\ Y_2 & -P \end{bmatrix} < 0_{(n+m) \times (n+m)}, \quad (\text{A.34})$$

$$\begin{bmatrix} I_n - P & A^\top P - A^\top C^\top Y_2^\top \\ PA - Y_2 CA & -P \end{bmatrix} < 0_{2n \times 2n}, \quad (\text{A.35})$$

where

$$M = P - Y_2 C - C^\top Y_2^\top + C^\top Y_4 C. \quad (\text{A.36})$$

By solving the constrained optimization problem

$$\min \psi \quad \text{s.t.} \quad (\text{A.33})-(\text{A.35}), \quad (\text{A.37})$$

we obtain the optimal matrix $\check{K} = P^{-1}Y_2$.

Assuming that the optimization problems (A.29) and (A.37) have been offline solved, SKP and SKF must only compute estimates of mean and covariance. Then, their complexity order is given by

$$O(\text{SKP}) = O(\text{SKF}) = O(n^3 + np + n^2m + nm^2).$$

A.2 Nonlinear Algorithms

In this section, two nonlinear Kalman estimators are reviewed. Consider the discrete-time uncertain nonlinear dynamical system

$$x_k = f(x_{k-1}, u_{k-1}, w_{k-1}), \quad (\text{A.38})$$

$$y_k = h(x_k, v_k), \quad (\text{A.39})$$

where $f: \mathbb{R}^n \times \mathbb{R}^p \times \mathbb{R}^q \rightarrow \mathbb{R}^n$ and $h: \mathbb{R}^n \times \mathbb{R}^r \rightarrow \mathbb{R}^m$ are the known process and measurement models, respectively, and $x_k \in \mathbb{R}^n$ is the state vector to be estimated, which is assumed to satisfy

$$a_{i,k} \leq x_{i,k} \leq b_{i,k}, \quad i = 1, \dots, n, \quad (\text{A.40})$$

where $a \in \mathbb{R}^n$ and $b \in \mathbb{R}^n$ are the lower and upper bounds of the states, respectively. The terms $w_{k-1} \in \mathbb{R}^q$ and $v_k \in \mathbb{R}^r$ represent process and measurement noise terms, respectively. For $k \in \mathbb{Z}_+$, the measurement vector $y_k \in \mathbb{R}^m$, the input vector $u_{k-1} \in \mathbb{R}^p$, and the functions f and h are assumed to be known.

The noise terms w_{k-1} and v_k are characterized by the random vectors W_{k-1} and V_k , whose means are $0_{q \times 1}$ and $0_{r \times 1}$, and whose covariances are $Q_{k-1} \in \mathbb{R}^{q \times q}$ and $R_k \in \mathbb{R}^{r \times r}$, respectively. Moreover, the initial state term x_0 is characterized by the random vector $\hat{X}_0$, whose mean $\hat{x}_0$ and covariance $\hat{P}_0^{\text{xx}}$ are known. We assume that the stochastic processes related to

W_{k-1} and V_k are independent among them and themselves for all $k \in \mathbb{Z}_+$. Moreover, $\hat{X}_0$ is assumed to be uncorrelated to W_{k-1} and V_k .

Analogously to the KF, the goal is to solve Problem 1.2.1 to estimate the *a posteriori* mean $\hat{x}_k$ and covariance $\hat{P}_k^{\text{xx}}$. Therefore, the next algorithms are addressed to the KF by means of a linearization procedure.

A.2.1 Unscented Kalman Filter

A.2.1.1 Unscented Transformation

GRVs are not closed under nonlinear transformations. There exist many methods to propagate the mean and the covariance of a random vector through a nonlinear transformation, such as Monte Carlo simulation, analytical linearization, statistical linearization, and sigma-points approaches (Särkkä, 2013). The unscented transformation (UT) is an algorithm based on sigma points.

Consider the nonlinear transformation of the prior random vectors X_1 and X_2 as well as the deterministic vector $d \in \mathbb{R}^p$ as

$$y = h(x_1, x_2, d), \quad (\text{A.41})$$

where $h : \mathbb{R}^n \times \mathbb{R}^r \rightarrow \mathbb{R}^m$ is known, $\hat{x}_1 \in \mathbb{R}^n$ and $\hat{x}_2 \in \mathbb{R}^r$ are the means, and $P^{x_1 x_1} \in \mathbb{R}^{n \times n}$ and $P^{x_2 x_2} \in \mathbb{R}^{r \times r}$ are the covariances. We now describe the steps of the UT in Algorithm A.3, according to the systematization proposed by Arasaratnam & Haykin (2009) and Menegaz et al. (2015). By assuming that the evaluation $h(\cdot)$ has complexity $O(1)$, we reach

$$O(\text{UT}) = O((n+r)^3 + m^2(n+r) + mn(n+r)).$$

An example is proposed to illustrate the nonlinear transformation of GRVs using different approaches.

Example A.2.1. Consider the nonlinear transformation from polar to Cartesian coordinates

$$y = h(x_1, x_2) = \begin{bmatrix} x_1 \cos(x_2) & x_1 \sin(x_2) \end{bmatrix}^\top, \quad (\text{A.42})$$

where x_1 is the radius and x_2 is the angle. Also, consider the GRVs $X_1 \sim \mathcal{N}(3, 1)$ and $X_2 \sim \mathcal{N}(0, 1)$. Both X_1 and X_2 compose the GRV $X \sim \mathcal{N}(\hat{x}, P^{\text{xx}})$, where $\hat{x} = \begin{bmatrix} 3 & 0 \end{bmatrix}^\top$ and $P^{\text{xx}} = I_2$.

Three different approaches are employed to transform the GRVs X_1 and X_2 over (A.42), namely the UT, analytical linearization, and Monte Carlo simulation. The former estimates the transformed random vector $\hat{Y} \sim \mathcal{N}(\hat{y}, \hat{P}^{\text{yy}})$ through Algorithm A.3. The second one is based on the first-order truncated Taylor series of (A.42), yielding $\tilde{y} = h(\hat{x}) + F(x - \hat{x})$, with $F = \nabla_x h(\hat{x})$ being the Jacobian matrix of $h(x)$ evaluated at $\hat{x}$, that

Algorithm A.3: : $[\hat{y}, \hat{P}^{yy}, \hat{P}^{x_1y}] = \text{UT}(\hat{x}_1, \hat{x}_2, P^{x_1x_1}, P^{x_2x_2}, d, h)$ (Menegaz et al., 2015)

- 1 $\hat{x} = \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix}$ is the augmented mean estimate
 - 2 $P^{\bar{x}\bar{x}} = \begin{bmatrix} P^{x_1x_1} & 0_{n \times r} \\ 0_{r \times n} & P^{x_2x_2} \end{bmatrix}$ is the augmented covariance matrix
 - 3 $X^{\text{sp}} = \hat{x}1_{1 \times 2\tilde{n}} + \sqrt{\tilde{n}} \begin{bmatrix} (P^{\bar{x}\bar{x}})^{1/2} & - (P^{\bar{x}\bar{x}})^{1/2} \end{bmatrix}$ is the sigma-point matrix, where $\tilde{n} = n + r$, $1_{1 \times 2\tilde{n}} \in \mathbb{R}^{1 \times 2\tilde{n}}$ is the vector of unitary elements, and $(\cdot)^{1/2}$ is the Cholesky decomposition such that $P^{\bar{x}\bar{x}} = (P^{\bar{x}\bar{x}})^{1/2} (P^{\bar{x}\bar{x}})^{\top/2}$
 - 4 $\gamma = \frac{1}{2\tilde{n}} 1_{2\tilde{n} \times 1}$ is the weight vector
 - 5 $Y^{\text{sp}} = h \left(X_{1:m, 1:2\tilde{n}}^{\text{sp}}, X_{n+1:n+r, 1:2\tilde{n}}^{\text{sp}}, d \right)$ is the transformed sigma-point matrix
 - 6 $\hat{y} = Y^{\text{sp}} \gamma$ is the transformed mean estimate
 - 7 $\hat{P}^{yy} = ((Y^{\text{sp}} - \hat{y}1_{1 \times 2\tilde{n}}) \odot 1_{m \times 1} \gamma^{\top}) (Y^{\text{sp}} - \hat{y}1_{1 \times 2\tilde{n}})^{\top}$ is the transformed covariance matrix, where $\odot$ represents the elementwise product between matrices with same dimension and m is the dimension of $\hat{y}$
 - 8 $\hat{P}^{x_1y} = ((X_{1:m, 1:2\tilde{n}}^{\text{sp}} - \hat{x}_1 1_{1 \times 2\tilde{n}}) \odot 1_{n \times 1} \gamma^{\top}) (Y^{\text{sp}} - \hat{y}1_{1 \times 2\tilde{n}})^{\top}$ is the cross correlation between x_1 and y
-

is, $F = \begin{bmatrix} \cos(\hat{x}_2) & -x_1 \sin(\hat{x}_2) \\ \sin(\hat{x}_2) & x_1 \cos(\hat{x}_2) \end{bmatrix}$. In this case, the transformed random vector $\check{Y}$ is characterized by $\check{Y} \sim \mathcal{N}(h(\hat{x}), F P^{\bar{x}\bar{x}} F^{\top})$. The latter is performed by yielding random samples from X_1 and X_2 . After, each sample is transformed by (A.42), whose same weight contribution leads to the transformed random vector $\tilde{Y}$ through sample mean and covariance equations.

In Figure A.1, the random vectors before and after each transformation are illustrated in two dimensions. Graph (a) presents the mean (o) and covariance matrix (—) of X before the transformation. Also, 2,000 random samples (.) from X are yielded. In graph (b), all three transformations are illustrated. Based on the transformed samples, the mean (o) and covariance matrix (—) of $\tilde{Y}$ are taken as reference to compare the remaining results. The UT provides mean ($\diamond$) and covariance matrix (--) for $\hat{Y}$, whose results are close to the reference. On the other hand, the estimates of mean (*) and covariance matrix (-.-) for $\check{Y}$, provided by the analytical linearization, are not close to the expected results. It suggests that, if the analytical linearization is employed over time with (A.42), then we may have the inconsistency problem.

Note that the region generated by the samples is not an ellipse, such that the actual transformed random vector is not Gaussian. Therefore, approaches based on GRVs seek to estimate the two first moments of the unknown transformed random vector.

A.2.1.2 Algorithm

The unscented Kalman filter (UKF) was proposed by Julier & Uhlmann (1997) motivated by limitations related to the EKF, mainly due to analytical linearization. However, the UKF method also provides approximated solutions, since GRVs are not closed under

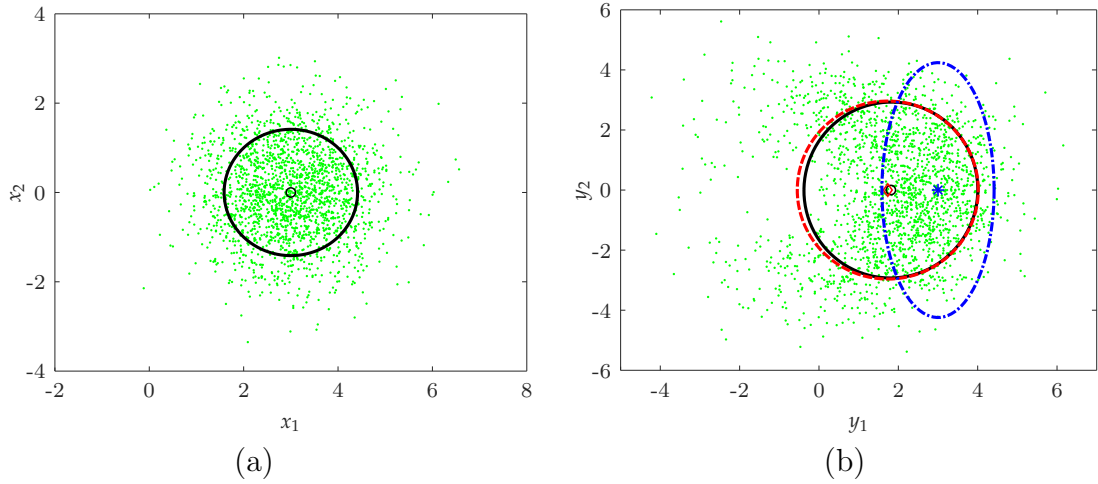


Figure A.1: One-step nonlinear transformation of GRVs. Let the nonlinear transformation be $y = [x_1 \cos(x_2) \ x_1 \sin(x_2)]^T$. Graph (a) illustrates the mean $\hat{x} = [3 \ 0]^T$ (o) and covariance $P^{xx} = I_2$ (—) of the GRV X . Also, 2,000 random samples (.) from X are illustrated. The most of the samples concentrate into the covariance around the mean. In graph (b), those samples are transformed over y . Based on the Monte Carlo simulation approach, the same weight contribution of such samples leads to the transformed random vector with new mean (o) and covariance (—). These results are considered as reference to compare the other ones due to their accuracy. From the unscented transformation (UT), the estimates of mean ($\diamond$) and covariance (--) are close to the reference. On the other hand, the first-order truncated Taylor series lead to the estimates of mean (*) and covariance (-.-) away from the reference. This is an example of inconsistency, where the actual covariance is not included into the estimated covariance. Therefore, the analytical linearization can be changed by the quadratic approach, in order to inflate the estimated covariance and, then, close it to the reference covariance. Another possibility is slightly to inflate the estimated covariance matrix.

nonlinear transformations.

The UT is used to propagate the mean and the covariance of the random vector $\hat{X}_{k-1}$ over the nonlinear functions (A.38) and (A.39), yielding the corresponding random vectors $\hat{X}_{k|k-1}$ and $\hat{Y}_{k|k-1}$. We now review the UKF in Algorithm A.4, whose complexity order is given by

$$O(\text{UKF}) = O(n^3 + q^3 + r^3 + m^3 + n^2(q + r) + n(q^2 + r^2) + (m^2 + mn)(n + r)).$$

This algorithm solves Problem 1.2.1 disregarding state constraints.

Algorithm A.4: $\hat{X}_k = \text{UKF}(\hat{X}_{k-1}, W_{k-1}, V_k, u_{k-1}, y_k, f, h)$ (Julier & Uhlmann, 1997)

- 1 $[\hat{x}_{k|k-1}, \hat{P}_{k|k-1}^{xx}, [\cdot]] = \text{UT}(\hat{x}_{k-1}, 0_{q \times 1}, \hat{P}_{k-1}^{xx}, Q_{k-1}, u_{k-1}, f)$, where $[\cdot]$ corresponds to a disregarded element, and UT is given by Algorithm A.3
 - 2 $[\hat{y}_{k|k-1}, \hat{P}_{k|k-1}^{yy}, \hat{P}_{k|k-1}^{xy}] = \text{UT}(\hat{x}_{k|k-1}, 0_{r \times 1}, \hat{P}_{k|k-1}^{xx}, R_k, 0_{p \times 1}, h)$
 - 3 Compute the Kalman gain $\tilde{K}_k$ in (A.19) to estimate the mean $\hat{x}_k$ in (A.15) and covariance $\hat{P}_k^{xx}$ in (A.22), obtaining $\hat{X}_k \sim \mathcal{N}(\hat{x}_k, \hat{P}_k^{xx})$
-

The UKF presents a competitive processing time compared to the EKF, and accuracy similar to or better than EKF (Julier & Uhlmann, 2004). As the EKF, the UKF may diverge due to inconsistency problems (Lefebvre* et al., 2004; Xiong et al., 2006). In this case, the numerical stability can be improved by slightly enlarging the covariance matrices Q_{k-1} and R_k (Lefebvre* et al., 2004; Xiong et al., 2006).

A.2.2 Truncated Unscented Kalman Filter

A.2.2.1 Probability Density Function Truncation

The PDF truncation (PT) method is revisited in Algorithm A.5 following steps similar to Simon (2006), in which both mean $\hat{x}$ and covariance $\hat{P}^{xx}$ estimates of the random vector $\hat{X}$ are redefined according to the interval constraints (A.40). In this algorithm, the PT is applied to the random vector $\hat{X}$ using n individual truncation steps, yielding the random vector $\hat{X}^t$. The complexity order of the PT is given by $O(n^4)$.

The PDF truncation procedure is now exemplified as in Teixeira et al. (2010).

Example A.2.2. Consider the random vector $\hat{X}$ with mean $\hat{x} = [0.25 \ 1]^\top$ and covariance $\hat{P}^{xx} = I_2$. Also, consider the interval constraints $a = [-0.5 \ 0]^\top$ and $b = [2 \ 2.5]^\top$. In Figure A.2, although $\hat{x}$ satisfies (A.40), the covariance $\hat{P}^{xx}$ violates the constraints a and b . After the truncation, the corresponding truncated mean and covariance estimates are

$$\hat{x}^t = \begin{bmatrix} 0.5430 \\ 1.1452 \end{bmatrix}, \quad \hat{P}^{xxt} = \begin{bmatrix} 0.4003 & 0 \\ 0 & 0.4157 \end{bmatrix},$$

which characterize the random vector $\hat{X}^t$ that is more precise than $\hat{X}$.

A.2.2.2 Interval-Constrained Unscented Transformation

This subsection revisits an alternative method to PT (Algorithm A.5) to enforce interval constraints on GRVs. Given a GRV $\hat{X} \sim \mathcal{N}(\hat{x}, \hat{P}^{xx})$ with realization $x \in \mathbb{R}^n$, consider the interval constraints (A.40). In order to correct the estimates $\hat{x}$ and $\hat{P}^{xx}$ according to the interval constraints a and b , we resume the interval-constrained unscented transformation (ICUT) in Algorithm A.6 following steps similar to Teixeira et al. (2010). In this approach, the sigma-point matrix $X^{sp} \in \mathbb{R}^{n \times (2n+1)}$ is yielded, whose j th column, $X_{:,j}^{sp}$, for $j = 1, \dots, (2n+1)$, defines a vector called sigma point. Each sigma point $X_{:,j}^{sp}$ is attributed a weight γ_j . At the end, the sigma-point matrix X^{sp} is used to determine the weighted GRV $\hat{Y} \sim \mathcal{N}(\hat{y}, \hat{P}^{yy})$, whose covariance is shaped by the parameter $\kappa > -n$. The complexity order of the ICUT is given by $O(n^3)$.

The ICUT algorithm is now exemplified as in Teixeira et al. (2010).

Example A.2.3. Consider a GRV $\hat{X} \sim \mathcal{N}(\hat{x}, \hat{P}^{xx})$, where $\hat{x} = [0.25 \ 1]^\top$ and $\hat{P}^{xx} = I_2$. Consider

Algorithm A.5: $\hat{X}^t = \text{PT}(\hat{X}, a, b)$ (Shimada et al., 1998)

Initialization: $\Theta \in \mathbb{R}^{n \times n}$

```

1   $\hat{x}_1^t = \hat{x}$ 
2   $\hat{p}_{1,\text{xt}}^{\text{xt}} = \hat{p}^{\text{xt}}$ 
3  for  $i = 1, \dots, n$  do
4       $[S, W] = \text{jordan}(\hat{p}_i^{\text{xt}})$ , where jordan returns the orthogonal matrix  $S \in \mathbb{R}^{n \times n}$  and
        the diagonal matrix  $W \in \mathbb{R}^{n \times n}$ 
5       $\Theta_{1,:} = \frac{1}{\sqrt{\hat{p}_{(i,i),i}^{\text{xt}}}} S_{i,:} W^{1/2}$ 
6      for  $j = 2, \dots, n$  do
7           $\psi = 0_{n \times 1}$ 
8          for  $q = 1, \dots, (j-1)$  do
9               $\psi = \psi + \left( e_j^\top (\Theta_{q,:})^\top \right) (\Theta_{q,:})^\top$ , where  $e_j = (\mathbf{I}_n)_{:,j}$ 
10             end
11              $\Theta_{j,:} = (e_j - \psi)^\top$ 
12             if  $\Theta_{j,:} = 0_{1 \times n}$  then
13                  $\psi = 0_{n \times 1}$ 
14                 for  $q = 1, \dots, (j-1)$  do
15                      $\psi = \psi + \left( e_1^\top (\Theta_{q,:})^\top \right) (\Theta_{q,:})^\top$ 
16                 end
17                  $\Theta_{j,:} = (e_1 - \psi)^\top$ 
18             end
19              $\Theta_{j,:} = \frac{1}{\|\Theta_{j,:}\|_2} \Theta_{j,:}$ 
20         end
21          $c_i = \frac{1}{\sqrt{\hat{p}_{(i,i),i}^{\text{xt}}}} (a_i - \hat{x}_{i,i}^t)$ 
22          $d_i = \frac{1}{\sqrt{\hat{p}_{(i,i),i}^{\text{xt}}}} (b_i - \hat{x}_{i,i}^t)$ 
23          $v_i = \frac{\sqrt{2}}{\sqrt{\pi} \left[ \text{erf}\left(\frac{d_i}{\sqrt{2}}\right) - \text{erf}\left(\frac{c_i}{\sqrt{2}}\right) \right]}$ , with  $\text{erf}(r) = \frac{2}{\sqrt{\pi}} \int_0^r \exp(-\tau^2) d\tau$  being the error function
24          $\mu_i = v_i \left[ \exp\left(\frac{-c_i^2}{2}\right) - \exp\left(\frac{-d_i^2}{2}\right) \right]$ 
25          $\sigma_i^2 = v_i \left[ \exp\left(\frac{-c_i^2}{2}\right) (c_i - 2\mu_i) - \exp\left(\frac{-d_i^2}{2}\right) (d_i - 2\mu_i) \right] + \mu_i^2 + 1$ 
26          $\hat{z}_i^t = [\mu_i \quad 0_{1 \times (n-1)}]^\top$ 
27          $\hat{p}_i^{\text{zzt}} = \text{diag}\left( [\sigma_i^2 \quad 1_{1 \times (n-1)}]^\top \right)$ 
28          $\hat{x}_{i+1}^t = S W^{1/2} \Theta^\top \hat{z}_i^t + \hat{x}_i^t$ 
29          $\hat{p}_{i+1}^{\text{xt}} = S W^{1/2} \Theta^\top \hat{p}_i^{\text{zzt}} \Theta W^{1/2} S^\top$ 
30     end
31      $\hat{x}^t = \hat{x}_{n+1}^t$ 
32      $\hat{p}^{\text{xt}} = \hat{p}_{n+1}^{\text{xt}}$ 
33      $\hat{X}^t \sim \mathcal{N}(\hat{x}^t, \hat{p}^{\text{xt}})$ 

```

also the interval constraints $a \leq x \leq b$, with $a = [-0.5 \quad 0]^\top$ and $b = [2 \quad 2.5]^\top$. In Figure A.3, the random vector $\hat{X}$ is illustrated with its own standard deviation. Note that the

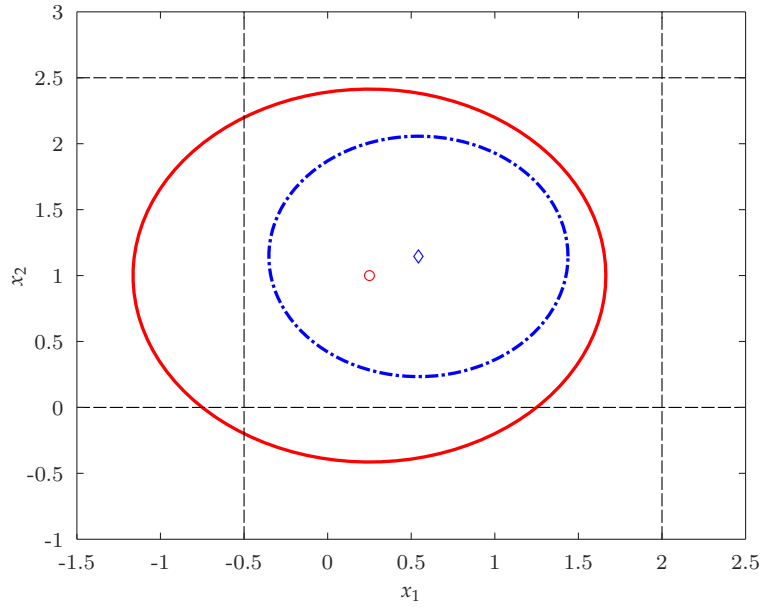


Figure A.2: Example of the PDF truncation procedure applied to the unconstrained GRV $\hat{X}$, characterized by mean $\hat{x} = [0.25 \ 1]^\top$ ($\circ$) and covariance $\hat{P}^{xx} = I_2$ ($—$), yielding the estimates of mean $\hat{x}^t = [0.5430 \ 1.1452]^\top$ ($\diamond$) and covariance $\hat{P}^{xxt} = \text{diag}([0.4003 \ 0.4157]^\top)$ ($- \cdot -$) of the truncated GRV $\hat{X}^t$.

mean $\hat{x}$ ($\circ$) satisfies the interval constraints ($--$), but the covariance $\hat{P}^{xx}$ ($—$) exceeds the lower bounds. After applying ICUT given by Algorithm A.6, the GRV $\hat{Y}$ is yielded with new mean $\diamond$ and covariance ($- \cdot -$). This GRV is not centrally symmetric, since some sigma points ($\times$) are projected onto the exceeded constraints. The different weights (γ) contribute to correct the probability region related to $\hat{X}$, since the largest weights are attributed to regions into the interval constraints, as γ_1 and γ_2 , or to less exceeded directions, as γ_4 .

A.2.2.3 Algorithm

Often, state estimation problems are formulated with state constraints. In particular, the interval constraints, or box constraints, denote that the states are restricted to a bounded region of the Euclidean space. This information is useful to improve both accuracy and feasibility of probability regions estimated by algorithms as the EKF and UKF, since the estimates are corrected onto the interval constraints.

The truncated unscented Kalman filter (TUKF) appends the PDF truncation step (Shimada et al., 1998; Teixeira et al., 2008b) to the UKF equations (Julier & Uhlmann, 1997) in order to enforce the interval constraints (A.40). That is, the *a posteriori* random vector $\hat{X}_k$ is written as the truncated random vector $\hat{X}_k^t$ in order to confine the largest PDF region into the interval region. Next, the TUKF is reviewed in Algorithm A.7, whose

Algorithm A.6: $\hat{Y} = \text{ICUT}(\hat{X}, a, b, \kappa)$ (Vachhani et al., 2006)

Initialization: $\check{x} \in \mathbb{R}^n$, $\Upsilon \in \mathbb{R}^{n \times 2n}$, $\theta \in \mathbb{R}^{2n}$

```

1 for  $i = 1, \dots, n$  do
2   if  $\hat{x}_i < a_i$  then
3      $\check{x}_i = a_i$ 
4   else if  $\hat{x}_i > b_i$  then
5      $\check{x}_i = b_i$ 
6   else
7      $\check{x}_i = \hat{x}_i$ 
8   end
9 end
10  $S = (\hat{P}^{\text{xx}})^{1/2}$  such that  $\hat{P}^{\text{xx}} = SS^\top$ 
11  $N = \begin{bmatrix} S & -S \end{bmatrix}$ 
12 for  $j = 1, \dots, 2n$  do
13   for  $i = 1, \dots, n$  do
14     if  $N_{(i,j)} == 0$  then
15        $\Upsilon_{(i,j)} = \sqrt{n + \kappa}$ 
16     else if  $N_{(i,j)} > 0$  then
17        $\Upsilon_{(i,j)} = \min\left(\sqrt{n + \kappa}, \frac{b_i - \hat{x}_i}{N_{(i,j)}}\right)$ 
18     else
19        $\Upsilon_{(i,j)} = \min\left(\sqrt{n + \kappa}, \frac{a_i - \hat{x}_i}{N_{(i,j)}}\right)$ 
20     end
21   end
22    $\theta_j = \min(\Upsilon_{:,j})$ 
23 end
24  $N_+ = \begin{bmatrix} \theta_1 N_{:,1} & \dots & \theta_n N_{:,n} \end{bmatrix}$ 
25  $N_- = \begin{bmatrix} \theta_{(n+1)} N_{:, (n+1)} & \dots & \theta_{2n} N_{:, 2n} \end{bmatrix}$ 
26  $X^{\text{sp}} = \check{x} 1_{1 \times (2n+1)} + \begin{bmatrix} 0_{n \times 1} & N_+ & N_- \end{bmatrix}$ 
27  $\varrho = \frac{2\kappa-1}{2(n+\kappa)(1_{1 \times 2n} \theta - (2n+1) \sqrt{n+\kappa})}$ 
28  $\beta = \frac{1}{2(n+\kappa)} - \varrho \sqrt{n + \kappa}$ 
29  $\gamma = \begin{bmatrix} 0 & \varrho \theta^\top \end{bmatrix}^\top + \beta 1_{(2n+1) \times 1}$ 
30  $\hat{y} = X^{\text{sp}} \gamma$ 
31  $\hat{P}^{\text{yy}} = ((X^{\text{sp}} - \hat{y} 1_{1 \times (2n+1)}) \odot 1_{n \times 1} \gamma^\top) (X^{\text{sp}} - \hat{y} 1_{1 \times (2n+1)})^\top$ , where  $\odot$  represents the elementwise
    product between matrices with same dimension
32  $\hat{Y} \sim \mathcal{N}(\hat{y}, \hat{P}^{\text{yy}})$ 

```

complexity order is given by

$$O(\text{TUKF}) = O(n^4 + q^3 + r^3 + m^3 + n^2(q + r) + n(q^2 + r^2) + (m^2 + mn)(n + r)).$$

This algorithm solves Problem 1.2.1 enforcing state interval constraints.

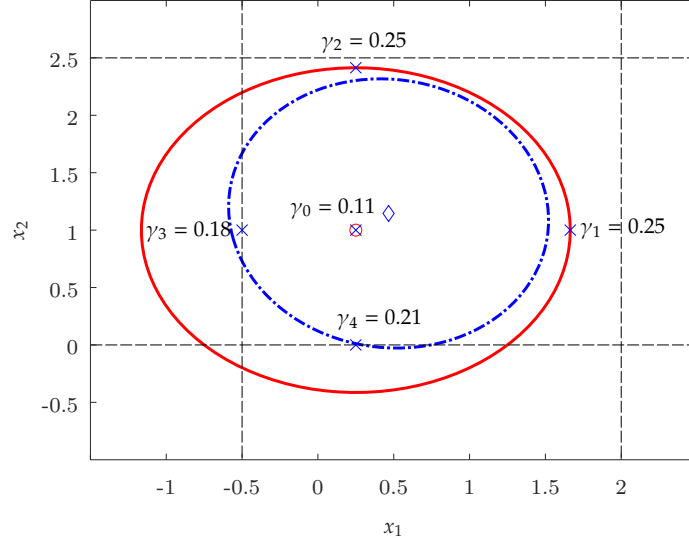


Figure A.3: An application of the ICUT algorithm together with $\kappa = 0$. Given a GRV $\hat{X}$ with mean $\hat{x} = [0.25 \ 1]^\top$ ($\circ$) and covariance $\hat{P}^{xx} = \mathbf{I}_2$ ($—$), the goal is to enforce the interval constraints $a \leq x \leq b$ ($--$), where $a = [-0.5 \ 0]^\top$ and $b = [2 \ 2.5]^\top$, using the ICUT given by Algorithm A.6. Since there exist exceeded bounds by $\hat{X}$, based on one standard deviation, some sigma points ($\times$) are projected onto such bounds and different weights (γ) are attributed to each sigma point. Thereby, the GRV $\hat{Y}$ is yielded with new mean $\diamond$ and covariance ($- \cdot -$).

Algorithm A.7: $\hat{X}_k^t = \text{TUKF}(\hat{X}_{k-1}^t, W_{k-1}, V_k, u_{k-1}, y_k, a_k, b_k, f, h)$ (Teixeira et al., 2008b)

- 1 $\hat{X}_k = \text{UKF}(\hat{X}_{k-1}^t, W_{k-1}, V_k, u_{k-1}, y_k, f, h)$, where UKF is given by Algorithm A.4
 - 2 $\hat{X}_k^t = \text{PT}(\hat{X}_k, a_k, b_k)$, where PT is given by Algorithm A.5
-

B

Set-based State Estimators: a Review

This appendix reviews the basis algorithms to develop state estimation throughout this work. Most of algorithms here presented are developed in the interval framework, in which a Kalman-like gain is designed to update the forecasts. Therefore, the reader is addressed to Appendix A to start this topic. Since this appendix receives most focus on zonotopes, some parallels can be traced to GRVs. For instance, center estimates remind mean estimates, and generator matrices remind the Cholesky decomposition of covariance matrices. Unlike the Kalman estimators, the set-based algorithms provide sets that include the true states. Further algorithms are here developed in the set-membership framework, where the measurement data are assimilated to the forecasts through intersection among sets. In order to better characterize these algorithms, we also derived their worst-case complexity order.

B.1 Linear Algorithms

Consider the discrete-time uncertain linear dynamical system given by

$$x_k = A_{k-1}x_{k-1} + B_{k-1}u_{k-1} + w_{k-1}, \quad (\text{B.1})$$

$$y_k = C_k x_k + v_k, \quad (\text{B.2})$$

where $A_{k-1} \in \mathbb{R}^{n \times n}$, $B_{k-1} \in \mathbb{R}^{n \times p}$, and $C_k \in \mathbb{R}^{m \times n}$ are the system matrices, $u_{k-1} \in \mathbb{R}^p$ is the input vector, $y_k \in \mathbb{R}^m$ is the output vector, $w_{k-1} \in \mathbb{R}^n$ is the process noise, $v_k \in \mathbb{R}^m$ is the

measurement noise, and $x_k \in \mathbb{R}^n$ is the state vector. We assume that the system matrices and the input and output vectors are known over $k \in \mathbb{Z}_+$. We also assume that the noise terms w_{k-1} and v_k , as well as the initial state term x_0 , are characterized by the set-based variables $\mathcal{W}_{k-1}$, $\mathcal{V}_k$, and $\hat{X}_0$, respectively.

For zonotopes, we have $\mathcal{W}_{k-1} = \{G_{k-1}^w, 0_{n \times 1}\}$, $\mathcal{V}_k = \{G_k^v, 0_{m \times 1}\}$, and $\hat{X}_0 = \{\hat{G}_0^x, \hat{x}_0\}$. Therefore, the goal is to solve Problems 1.2.2 and 1.2.3 to estimate the zonotope $\hat{X}_k = \{\hat{G}_k^x, \hat{x}_k\}$ based on different objective functions, such as Frobenius norm (1.6), volume (1.7), and H_∞ norm (2.67), for instance.

For CZs, we have $\mathcal{W}_{k-1} = \{G_{k-1}^w, 0_{n \times 1}, A_{k-1}^w, b_{k-1}^w\}$, $\mathcal{V}_k = \{G_k^v, 0_{m \times 1}, A_k^v, b_k^v\}$, and $\hat{X}_0 = \{\hat{G}_0^x, \hat{x}_0, \hat{A}_0^x, \hat{b}_0^x\}$. Then, the goal is to solve Problem 1.2.2 via generalized intersection (Property 2.2.8) to compute the CZ $\hat{X}_k = \{\hat{G}_k^x, \hat{x}_k, \hat{A}_k^x, \hat{b}_k^x\}$.

B.1.1 Zonotopic Kalman Predictor

B.1.1.1 Zonotope Reduction

The zonotope $\mathcal{X}$ is an affine transformation of the unitary box $\mathcal{B}^{n_g}$, where n_g corresponds to the number of generators and denotes the zonotope order. Algorithms based on zonotopes present an exponentially increasing computational burden, due to the order growth over operations. To circumvent this computational issue, order reduction algorithms must be employed. Such algorithms reduce the order of the input set guaranteeing the inclusion property. Therefore, the order n_g is reduced to the desired order φ_g adding wrapping effect. However, a tradeoff between precision and processing time should be sought, because it affects the performance of the algorithms. Zonotopes can be reduced to boxes (Kühn, 1998), parallelotopes (Alamo et al., 2008; Althoff et al., 2010), or simpler zonotopes (Combastel, 2003, 2005; Girard, 2005; Le et al., 2013b; Scott et al., 2016; Yang & Scott, 2018).

To reduce the order of the zonotope $\mathcal{X}$, we review the method proposed by Combastel (2003) in Algorithm B.1, whose complexity order is given by $O(\text{ZR}) = n_g (n + \log(n_g))$ (Yang & Scott, 2018). This technique splits the n_g generators of the zonotope $\mathcal{X}$ based on their 2-norm, to build two other zonotopes. One of them is chosen to be overestimated by its *interval hull*. At the end, this interval hull is added to the other zonotope, yielding the reduced-order zonotope $\mathcal{X}^\downarrow$ with the desired order φ_g .

Based on Kühn (1998), there is no systematic way to choose the order reduction φ_g , since this is a tradeoff between precision and processing time. A practical way to choose the order reduction φ_g is by means of trial-and-error simulations, where the Frobenius norm of G^x is evaluated over time for different reduction orders φ_g . Ultimately, we select an order that significantly minimizes the Frobenius norm. Another procedure is to choose the highest possible reduction order φ_g for which the processing time is compatible with the sampling time of the application.

Two ways to compute the interval hull of the zonotope $\mathcal{X} \subset \mathbb{R}^n$ are noted. The first is to apply the natural interval extension (Definition 2.2.15) to (2.7). In this case, the interval enclosure coincides with the interval hull, since there are no repeated variables. The second possibility is to define the smallest possible reduction order ($\varphi_g = n$), then apply the ZR to completely reduce the zonotope $\mathcal{X}$, that is, $\mathcal{X}^\downarrow = \text{ZR}(\mathcal{X}, n)$. In other words, all generators of G^x are evaluated at step 8.

Algorithm B.1: $\mathcal{X}^\downarrow = \text{ZR}(\mathcal{X}, \varphi_g)$ (Combastel, 2003)

```

1 if  $n_g \leq \varphi_g$  then
2    $G^{\downarrow x} = G^x$ 
3 else
4    $\psi = \sqrt{\text{Sum}(G^x \odot G^x, \text{row})}$ , where  $\odot$  represents the point-to-point product
      between matrices of same dimension,  $\text{Sum}(\cdot, \text{row})$  represents the summation
      of rows that yields a row, and  $\sqrt{\cdot}$  is the square root of each element
5    $G^{xs} = \text{Sort}(G^x, \psi)$ , where  $\text{Sort}(\cdot, \cdot)$  means that the columns of  $G^x$  are sorted such
      that  $\psi_j \geq \psi_{j+1}$ , for  $j = 1, \dots, n_g - 1$ 
6    $G^x_{>} = G^{xs}_{:, 1:(\varphi_g - n)}$ 
7    $G^x_{<} = G^{xs}_{: (\varphi_g - n + 1): n_g}$ 
8    $G^{\text{box}} = \text{diag}(|G^x_{<}| 1_{n_{g<} \times 1})$ , where  $|G^x_{<}|$  is the absolute value of each element of the
      matrix  $G^x_{<}$ ,  $1_{n_{g<} \times 1}$  is the vector of unitary elements, and  $\text{diag}(\cdot)$  is the returned
      diagonal matrix
9    $G^{\downarrow x} = \begin{bmatrix} G^x_{>} & G^{\text{box}} \end{bmatrix}$ 
10 end
11  $\mathcal{X}^\downarrow = \{G^{\downarrow x}, \hat{x}\}$ 

```

An example is proposed to illustrate different reductions over a given zonotope.

Example B.1.1. Consider the zonotope $\mathcal{X} = \left\{ \begin{bmatrix} 1 & 2 & 3 & 4 \\ 4 & 3 & 2 & 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \end{bmatrix} \right\}$, whose order is $n_g = 4$. In Figure B.1, the zonotope $\mathcal{X}$ (---) and its two order reductions are illustrated with the same center ($\diamond$). To reach $\varphi_g = 3$ (--) and $\varphi_g = 2$ (—) individually, the ZR given by Algorithm B.1 is applied to $\mathcal{X}$. Note that the zonotopes have the same domain in one dimension before and after the reduction. By reducing the order of the zonotope $\mathcal{X}$, the obtained sets become larger due to the guaranteed characteristic. Moreover, the smallest order reduction to be reached is $\varphi_g = 2$, where the interval hull of $\mathcal{X}$ is obtained. It is noteworthy that, in practice, a tradeoff between processing time and precision should be pursued.

B.1.1.2 Algorithm

The zonotopic Kalman predictor for LTV systems (ZKP) proposed in Combastel (2015) has the same observer structure of the KP. It returns *a posteriori* centers and generator matrices, whose parallels with respect to the KP can be traced as follows. Centers remind

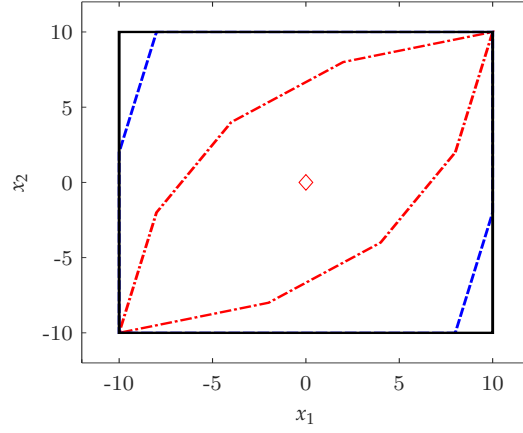


Figure B.1: Illustrative example of the order reduction over zonotopes. The zonotope $\mathcal{X}$ ($-\cdot-$) from Example B.1.1 has its order $n_g = 4$ reduced to $\varphi_g = 3$ ($--$) and $\varphi_g = 2$ ($---$). The employed reduction algorithm leads to other zonotopes with the same center ($\diamond$), but with different generator matrices. The smallest reduction order to be reached is $\varphi_g = 2$. In this case, the smallest box that contains $\mathcal{X}$ is obtained. The main idea is to enlarge $\mathcal{X}$ to reduce the order n_g . This is an example of the wrapping effect.

means, while generator matrices yield covariation matrices, which remind covariance matrices. Then, KP works on covariance matrices while ZKP works on generator matrices. Since a generator matrix can have more columns than rows, the square root of covariation matrices should be avoided. This is a significant difference between generator matrix and square root of covariance matrix. Another difference is the need to fix complexity in the ZKP, given that the amount of generators can increase indefinitely over operations as summation, for instance.

As for the KP (Algorithm A.1), we have the center estimates $\hat{x}_{k|k-1}$ in (A.3), $\hat{y}_{k-1}$ in (A.4), and $\hat{x}_k$ in (A.5), which leads to the estimation error $e_k \triangleq x_k - \hat{x}_k$ in (A.6). This error is characterized by the zonotope $\hat{\mathcal{X}}_k^z = \{\hat{G}_k^x, 0_{n \times 1}\}$. After replacing each realization of e_k by its corresponding zonotope, we obtain

$$\hat{\mathcal{X}}_k^z = (A_{k-1} - K_{k-1}C_{k-1})\hat{\mathcal{X}}_{k-1}^z \oplus \mathcal{W}_{k-1} \oplus (-K_{k-1}\mathcal{V}_{k-1}), \quad (\text{B.3})$$

whose generator matrix is given by

$$\hat{G}_k^x = \begin{bmatrix} (A_{k-1} - K_{k-1}C_{k-1})\hat{G}_{k-1}^{lx} & G_{k-1}^w & -K_{k-1}G_{k-1}^v \end{bmatrix}, \quad (\text{B.4})$$

with $\hat{G}_{k-1}^{lx}$ being the generator matrix of the reduced-order zonotope $\hat{\mathcal{X}}_{k-1}^l = \text{ZR}(\hat{\mathcal{X}}_{k-1}, \varphi_g)$.

In order to determine the optimal matrix $\check{K}_{k-1}$, we solve $\frac{\partial \text{tr}(\hat{G}_k^x (\hat{G}_k^x)^\top)}{\partial K_{k-1}} = 0$, yielding

$$\check{K}_{k-1} = \hat{C}_{k-1}^{xy} (\hat{C}_{k-1}^{yy})^{-1}, \quad (\text{B.5})$$

where

$$\hat{C}_{k-1}^{xx} = \hat{G}_{k-1}^{\downarrow x} (\hat{G}_{k-1}^{\downarrow x})^\top, \quad (\text{B.6})$$

$$\hat{C}_{k-1}^{xy} = A_{k-1} \hat{C}_{k-1}^{xx} C_{k-1}^\top, \quad (\text{B.7})$$

$$C_{k-1}^{vv} = G_{k-1}^v (G_{k-1}^v)^\top, \quad (\text{B.8})$$

$$\hat{C}_{k-1}^{yy} = C_{k-1} \hat{C}_{k-1}^{xx} C_{k-1}^\top + C_{k-1}^{vv}. \quad (\text{B.9})$$

Next, the ZKP is presented in Algorithm B.2, which solves Problem 1.2.3 disregarding state constraints. Its complexity order is given by

$$O(\text{ZKP}) = O(n_g(n + \log(n_g)) + n(p + mn_g^v) + n^2(n + m + \varphi_g) + m^2(n + m + n_g^v)).$$

Algorithm B.2: $\hat{\mathcal{X}}_k = \text{ZKP}(\hat{\mathcal{X}}_{k-1}, A_{k-1}, B_{k-1}, \mathcal{W}_{k-1}, C_{k-1}, \mathcal{V}_{k-1}, u_{k-1}, y_{k-1}, \varphi_g)$ (Combastel, 2015)

- 1 Compute $\hat{\mathcal{X}}_{k-1}^\downarrow = \text{ZR}(\hat{\mathcal{X}}_{k-1}, \varphi_g)$, where ZR is given by Algorithm B.1
 - 2 Compute the centers $\hat{x}_{k|k-1}$ in (A.3) and $\hat{y}_{k-1}$ in (A.4), and the covariation matrices $\hat{C}_{k-1}^{xx}$ in (B.6), $\hat{C}_{k-1}^{xy}$ in (B.7), C_{k-1}^{vv} in (B.8), and $\hat{C}_{k-1}^{yy}$ in (B.9)
 - 3 Compute the gain matrix (B.5), and thereby, the *a posteriori* generator matrix $\hat{G}_k^x$ and center $\hat{x}_k$ estimates given by (B.4) and (A.5), respectively, to obtain $\hat{\mathcal{X}}_k = \{\hat{G}_k^x, \hat{x}_k\}$
-

B.1.2 Minimum-Trace Zonotopic Filter for Descriptor Systems

The minimum-trace zonotopic filter for descriptor LTV systems (DTZF) is proposed in Tang et al. (2020). A descriptor system has additionally a known matrix $E_k \in \mathbb{R}^{n \times n}$ multiplying x_k in (B.1), that is,

$$E_k x_k = A_{k-1} x_{k-1} + B_{k-1} u_{k-1} + w_{k-1}. \quad (\text{B.10})$$

As $\text{rank}(E_k)$ may be incomplete, descriptor systems are also known like *singular systems*. This system is reduced to the state space when $E_k = I_n$. In order to design observers, an essential assumption is that

$$\text{rank} \left(\begin{bmatrix} E_k \\ C_k \end{bmatrix} \right) = n, \quad \forall k \geq 1.$$

According to Tang et al. (2020), such assumption ensures that x_k is uniquely determined from (B.2) and (B.10) when the involved quantities (representations of system and uncertainty) are known. The uniqueness of solution of the observer proposed by Tang et al. (2020) is also based on $G_{k-1}^w (G_{k-1}^w)^\top > 0_{n \times n}$, $G_{k-1}^v (G_{k-1}^v)^\top > 0_{m \times m}$, and $G_k^v (G_k^v)^\top > 0_{m \times m}$.

Therefore, there exists a full-rank matrix $\begin{bmatrix} T_k & N_k \end{bmatrix}$ such that

$$\begin{bmatrix} T_k & N_k \end{bmatrix} \begin{bmatrix} E_k \\ C_k \end{bmatrix} = \mathbf{I}_n. \quad (\text{B.11})$$

Combining (B.2) and (B.10) with (B.11) yields

$$\begin{aligned} \mathbf{x}_k &= (T_k E_k + N_k C_k) \mathbf{x}_k \\ &= T_k E_k \mathbf{x}_k + N_k (y_k - v_k) \\ &= T_k A_{k-1} \mathbf{x}_{k-1} + T_k B_{k-1} u_{k-1} + T_k w_{k-1} + N_k y_k - N_k v_k. \end{aligned} \quad (\text{B.12})$$

Therefore, the following state estimate is proposed:

$$\hat{\mathbf{x}}_k = T_k A_{k-1} \hat{\mathbf{x}}_{k-1} + T_k B_{k-1} u_{k-1} + N_k y_k + K_{k-1} (y_{k-1} - C_{k-1} \hat{\mathbf{x}}_{k-1}), \quad (\text{B.13})$$

whose estimation error is given by

$$e_k \triangleq (T_k A_{k-1} - K_{k-1} C_{k-1}) e_{k-1} + T_k w_{k-1} - K_{k-1} v_{k-1} - N_k v_k. \quad (\text{B.14})$$

Replacing each realization by its zonotope yields

$$\begin{aligned} \hat{\mathcal{X}}_k^z &= (T_k A_{k-1} - K_{k-1} C_{k-1}) \hat{\mathcal{X}}_{k-1}^{z\downarrow} \oplus T_k \mathcal{W}_{k-1} \oplus (-K_{k-1} \mathcal{V}_{k-1}) \oplus (-N_k \mathcal{V}_k) \\ &= \left\{ [(T_k A_{k-1} - K_{k-1} C_{k-1}) \hat{G}_{k-1}^{lx} \quad T_k G_{k-1}^w \quad -K_{k-1} G_{k-1}^v \quad -N_k G_k^v], 0_{n \times 1} \right\} \\ &= \left\{ \hat{G}_k^x, 0_{n \times 1} \right\}, \end{aligned} \quad (\text{B.15})$$

with $\hat{G}_{k-1}^{lx}$ being the generator matrix of the reduced-order zonotope $\hat{\mathcal{X}}_{k-1}^\downarrow = \text{ZR}(\hat{\mathcal{X}}_{k-1}, \varphi_g)$.

The optimization problem to be solved is the F -radius of $\hat{\mathcal{X}}_k^z$ subject to the equality constraints (B.11). In order to treat the constrained optimization problem as an unconstrained optimization problem, the Lagrangian function is used in Tang et al. (2020) to yield

$$J_k^{\text{eqfrad}}(T_k, N_k, K_{k-1}, \Lambda_k) = \text{tr} \left(\hat{G}_k^x (\hat{G}_k^x)^\top \right) + \text{tr} \left(\Lambda_k (E_k^\top T_k^\top + C_k^\top N_k^\top - \mathbf{I}_n) \right), \quad (\text{B.16})$$

where $\Lambda_k \in \mathbb{R}^{n \times n}$.

Calculating the partial derivative of J_k^{eqfrad} with respect to each matrix variable, we obtain

$$\frac{\partial J_k^{\text{eqfrad}}}{\partial T_k} = 2T_k \hat{C}_{k-1}^{xx} - 2K_{k-1} (\hat{C}_{k-1}^{xy})^\top + \Lambda_k E_k^\top, \quad (\text{B.17})$$

$$\frac{\partial J_k^{\text{eqfrad}}}{\partial N_k} = 2N_k C_k^{vv} + \Lambda_k C_k^\top, \quad (\text{B.18})$$

$$\frac{\partial J_k^{\text{eqfrad}}}{\partial K_{k-1}} = 2K_{k-1} \hat{C}_{k-1}^{yy} - 2T_k \hat{C}_{k-1}^{xy}, \quad (\text{B.19})$$

$$\frac{\partial J_k^{\text{eqfrac}}}{\partial \Lambda_k} = T_k E_k + N_k C_k - I_n, \quad (\text{B.20})$$

where $\hat{C}_{k-1}^{\text{xx}}$, $\hat{C}_{k-1}^{\text{xy}}$, C_{k-1}^{vv} , and $\hat{C}_{k-1}^{\text{yy}}$ are given by (B.6), (B.7), (B.8), and (B.9), respectively, and

$$C_{k-1}^{\text{ww}} = G_{k-1}^{\text{w}} (G_{k-1}^{\text{w}})^{\top}, \quad (\text{B.21})$$

$$C_k^{\text{vv}} = G_k^{\text{v}} (G_k^{\text{v}})^{\top}, \quad (\text{B.22})$$

$$\hat{C}_{k|k-1}^{\text{xx}} = A_{k-1} \hat{C}_{k-1}^{\text{xx}} A_{k-1}^{\top} + C_{k-1}^{\text{ww}}. \quad (\text{B.23})$$

Equating each partial derivative to zero yields the following result written as a linear equation system:

$$\Psi_k \Sigma_k + \Lambda_k \Pi_k^{\top} = 0_{n \times (n+2m)}, \quad (\text{B.24})$$

$$\Psi_k \Pi_k = I_n, \quad (\text{B.25})$$

where

$$\Psi_k = \begin{bmatrix} T_k & N_k & K_{k-1} \end{bmatrix}, \quad \Pi_k = \begin{bmatrix} E_k^{\top} & C_k^{\top} & 0_{n \times m} \end{bmatrix}^{\top}, \quad (\text{B.26})$$

$$\Sigma_k = 2 \begin{bmatrix} \hat{C}_{k|k-1}^{\text{xx}} & 0_{n \times m} & -\hat{C}_{k-1}^{\text{xy}} \\ 0_{m \times n} & C_k^{\text{vv}} & 0_{m \times m} \\ \left(-\hat{C}_{k-1}^{\text{xy}}\right)^{\top} & 0_{m \times m} & \hat{C}_{k-1}^{\text{yy}} \end{bmatrix}. \quad (\text{B.27})$$

For convenience, we can rewrite Σ_k as follows:

$$\Sigma_k = \Xi_k + \Gamma_k, \quad (\text{B.28})$$

where

$$\Xi_k = 2 \begin{bmatrix} C_{k-1}^{\text{ww}} & 0_{n \times m} & 0_{n \times m} \\ 0_{m \times n} & C_k^{\text{vv}} & 0_{m \times m} \\ 0_{m \times n} & 0_{m \times m} & C_{k-1}^{\text{vv}} \end{bmatrix}, \quad (\text{B.29})$$

$$\Gamma_k = 2 \begin{bmatrix} A_{k-1} \\ 0_{m \times n} \\ -C_{k-1} \end{bmatrix} \hat{C}_{k-1}^{\text{xx}} \begin{bmatrix} A_{k-1} \\ 0_{m \times n} \\ -C_{k-1} \end{bmatrix}^{\top}. \quad (\text{B.30})$$

On the one hand, since $C_{k-1}^{\text{ww}} > 0_{n \times n}$, $C_{k-1}^{\text{vv}} > 0_{m \times m}$, and $C_k^{\text{vv}} > 0_{m \times m}$, we have $\Xi_k > 0_{(n+2m) \times (n+2m)}$. On the other hand, $\Gamma_k \geq 0_{(n+2m) \times (n+2m)}$ since $0_{m \times n} \hat{C}_{k-1}^{\text{xx}} 0_{n \times m} = 0_{m \times m}$. Therefore, the positiveness of Ξ_k assures that $\Sigma_k > 0_{(n+2m) \times (n+2m)}$ is invertible. Thus, from (B.24), the optimal matrix $\check{\Psi}_k$ is uniquely given by

$$\check{\Psi}_k = -\Lambda_k \Pi_k^{\top} \Sigma_k^{-1}. \quad (\text{B.31})$$

After substituting this result in (B.25), we obtain

$$\Lambda_k(-\Pi_k^\top \Sigma_k^{-1} \Pi_k) = \mathbf{I}_n. \quad (\text{B.32})$$

Since Π_k has full rank, $-\Pi_k^\top \Sigma_k^{-1} \Pi_k$ is invertible. Then, the optimal matrix $\check{\Lambda}_k$ is uniquely determined by

$$\check{\Lambda}_k = -(\Pi_k^\top \Sigma_k^{-1} \Pi_k)^{-1}. \quad (\text{B.33})$$

Due to the fact that each matrix variable is uniquely determined, we guarantee the uniqueness of solution for the linear equation system (B.24)-(B.25). It implies that the original optimization problem is convex.

Next, we describe the DTZF in Algorithm B.3, which solves Problem 1.2.3 enforcing state linear equality constraints. Its complexity order is given by

$$O(\text{DTZF}) = O\left(n_s \left(n + \log(n_s)\right) + (n + m)^3 + n^2(p + \varphi_g + n_g^w) + (nm + m^2)(n_{g,k-1}^v + n_{g,k}^v)\right).$$

Algorithm B.3: $\hat{\mathcal{X}}_k = \text{DTZF}(\hat{\mathcal{X}}_{k-1}, E_k, A_{k-1}, B_{k-1}, \mathcal{W}_{k-1}, C_{k-1}, C_k, \mathcal{V}_{k-1}, \mathcal{V}_k, u_{k-1}, y_{k-1}, y_k, \varphi_g)$
(Tang et al., 2020)

- 1 Compute $\hat{\mathcal{X}}_{k-1}^\downarrow = \text{ZR}(\hat{\mathcal{X}}_{k-1}, \varphi_g)$, where ZR is given by Algorithm B.1
 - 2 Compute the matrices Π_k in (B.26) and Σ_k in (B.27) to yield $\check{\Lambda}_k$ in (B.33) and $\check{\Psi}_k$ in (B.31)
 - 3 Determine the optimal matrices $\check{T}_k$, $\check{N}_k$, and $\check{K}_{k-1}$ from $\check{\Psi}_k$ in (B.26)
 - 4 Compute the *a posteriori* generator matrix $\hat{G}_k^x$ and center $\hat{x}_k$ estimates given by (B.15) and (B.13), respectively, to obtain $\hat{\mathcal{X}}_k = \{\hat{G}_k^x, \hat{x}_k\}$
-

B.1.3 Minimum H_∞ Norm Zonotopic Filter

The minimum H_∞ norm zonotopic filter for LTI systems ($H_\infty\text{ZF}$) is proposed in Tang et al. (2019a). Consider a discrete-time linear time-invariant dynamical system represented by (B.1)-(B.2) for which now matrices A , B , G^w , C , and G^v are constant. Since the uncertainties are characterized by zonotopes, an interval observer with appealing properties can be designed for the *steady state*, in which the exact interval hull of the *a posteriori* zonotope $\hat{\mathcal{X}}_k$, $\square\hat{\mathcal{X}}_k$, is provided with the smallest processing time. Note that this technique is addressed to LTI systems whose transient is disregarded.

The $H_\infty\text{ZF}$ is derived from (B.11), where $E = \mathbf{I}_n$, and T and N are constant design matrices. Following steps similar to the DTZF (Algorithm B.3), the estimation error is given by

$$\begin{aligned} e_k &\triangleq (TA - KC)e_{k-1} + Tw_{k-1} - Kv_{k-1} - Nv_k \\ &= (TA - KC)e_{k-1} + TG^w \xi_{k-1}^w - KG^v \xi_{k-1}^v - NG^v \xi_k^v, \end{aligned} \quad (\text{B.34})$$

where K is a constant gain matrix, $\xi_{k-1}^w \in \mathcal{B}_g^w$, $\xi_{k-1}^v \in \mathcal{B}_g^v$, and $\xi_k^v \in \mathcal{B}_g^v$. Note that we made explicit the generator matrices $G^w \in \mathbb{R}^{n \times n_g^w}$ and $G^v \in \mathbb{R}^{m \times n_g^v}$, since they influence the H_∞ norm.

By convenience, the error e_k is rewritten as the error system

$$e_k = \tilde{A}e_{k-1} + \tilde{B}\tilde{w}_{k-1}, \quad (\text{B.35})$$

$$z_k = e_k, \quad (\text{B.36})$$

where $\tilde{w}_{k-1} = \left[(\xi_{k-1}^w)^\top \quad (\xi_{k-1}^v)^\top \quad (\xi_k^v)^\top \right]^\top$ and

$$\tilde{A} = TA - KC, \quad (\text{B.37})$$

$$\tilde{B} = \begin{bmatrix} TG^w & -KG^v & -NG^v \end{bmatrix}. \quad (\text{B.38})$$

Now, we formulate an optimization problem based on H_∞ norm to compute the design matrices. According to Lemma 2.5.3, the error system (B.35)-(B.36) leads to the following constrained optimization problem:

$$\begin{aligned} \min \quad & \gamma \\ \text{s.t.} \quad & \begin{bmatrix} \tilde{A}^\top P \tilde{A} + I_n - P & \tilde{A}^\top P \tilde{B} \\ \tilde{B}^\top P \tilde{A} & \tilde{B}^\top P \tilde{B} - \gamma I_\varrho \end{bmatrix} < 0_{(n+\varrho) \times (n+\varrho)}, \end{aligned}$$

where $\varrho = n_g^w + 2n_g^v$, and $P = P^\top > 0_{n \times n}$ is an unknown matrix related to the stability of (B.35)-(B.36). The prior inequality constraint can be rewritten as

$$\begin{bmatrix} I_n - P & 0_{n \times \varrho} \\ 0_{\varrho \times n} & -\gamma I_\varrho \end{bmatrix} + \begin{bmatrix} \tilde{A}^\top \\ \tilde{B}^\top \end{bmatrix} P \begin{bmatrix} \tilde{A} & \tilde{B} \end{bmatrix} < 0_{(n+\varrho) \times (n+\varrho)}.$$

Applying the Schur's complement, we obtain

$$\begin{bmatrix} \begin{bmatrix} I_n - P & 0_{n \times \varrho} \\ 0_{\varrho \times n} & -\gamma I_\varrho \end{bmatrix} & \begin{bmatrix} \tilde{A}^\top P \\ \tilde{B}^\top P \end{bmatrix} \\ \begin{bmatrix} P \tilde{A} & P \tilde{B} \end{bmatrix} & -P \end{bmatrix} < 0_{(2n+\varrho) \times (2n+\varrho)}.$$

Since T and N are related by (B.11), we make explicit such dependency using Lemma 2.5.1, yielding

$$T = \Theta^\dagger \alpha_1 + S\Psi \alpha_1, \quad (\text{B.39})$$

$$N = \Theta^\dagger \alpha_2 + S\Psi \alpha_2, \quad (\text{B.40})$$

where $S \in \mathbb{R}^{n \times (n+m)}$ is an arbitrary matrix and

$$\Theta = \begin{bmatrix} I_n \\ C \end{bmatrix}, \quad \alpha_1 = \begin{bmatrix} I_n \\ 0_{m \times n} \end{bmatrix}, \quad \alpha_2 = \begin{bmatrix} 0_{n \times m} \\ I_m \end{bmatrix}, \quad \Psi = I_{(n+m)} - \Theta \Theta^\dagger. \quad (\text{B.41})$$

As the products $P(TA - KC)$ and $P \begin{bmatrix} TG^w & -KG^v & -NG^v \end{bmatrix}$ are nonlinear, we define the linearizing transformations $Y_1 = PS$ and $Y_2 = PK$. Thereby, we obtain the following LMI:

$$\begin{bmatrix} I_n - P & 0_{n \times \varrho} & \Omega_1^\top \\ 0_{\varrho \times n} & -\gamma I_\varrho & \Omega_2^\top \\ \Omega_1 & \Omega_2 & -P \end{bmatrix} < 0_{(2n+\varrho) \times (2n+\varrho)}, \quad (\text{B.42})$$

where

$$\Omega_1 = P\Theta^\dagger \alpha_1 A + Y_1 \Psi \alpha_1 A - Y_2 C, \quad (\text{B.43})$$

$$\Omega_2 = \begin{bmatrix} (P\Theta^\dagger \alpha_1 + Y_1 \Psi \alpha_1) G^w & -Y_2 G^v & -(P\Theta^\dagger \alpha_2 + Y_1 \Psi \alpha_2) G^v \end{bmatrix}. \quad (\text{B.44})$$

Then, the constrained optimization problem to be solved is summarized as follows.

Theorem B.1.1 (Tang et al. (2019a)). Consider a parameter $\gamma > 0$ and matrices $P = P^\top > 0_{n \times n}$, $Y_1 \in \mathbb{R}^{n \times (n+m)}$, and $Y_2 \in \mathbb{R}^{n \times m}$. By solving the constrained optimization problem

$$\begin{aligned} \min \quad & \gamma \\ \text{s.t.} \quad & (\text{B.42}), \end{aligned} \quad (\text{B.45})$$

the following optimal matrices are obtained:

$$\check{T} = \Theta^\dagger \alpha_1 + \check{P}^{-1} \check{Y}_1 \Psi \alpha_1, \quad (\text{B.46})$$

$$\check{N} = \Theta^\dagger \alpha_2 + \check{P}^{-1} \check{Y}_1 \Psi \alpha_2, \quad (\text{B.47})$$

$$\check{K} = \check{P}^{-1} \check{Y}_2. \quad (\text{B.48})$$

Since the design matrices are given, we develop the procedure to timely yield the interval hull $\square \hat{X}_k = \hat{x}_k \oplus \square \hat{X}_k^z$. Retake the error parcel e_k (B.35), where e_0 is the initial term. From e_1 to e_k , the reachability analysis leads to

$$e_1 = \tilde{A}e_0 + Tw_0 - Kv_0 - Nv_1, \quad (\text{B.49})$$

$$e_2 = \tilde{A}^2 e_0 + \tilde{A}Tw_0 + Tw_1 - \tilde{A}Kv_0 - Kv_1 - \tilde{A}Nv_1 - Nv_2, \quad (\text{B.50})$$

$$\vdots \quad (\text{B.51})$$

$$e_k = \tilde{A}^k e_0 + \sum_{i=0}^{k-1} \tilde{A}^i Tw_{k-1-i} - \sum_{i=0}^{k-1} \tilde{A}^i Kv_{k-1-i} - \sum_{i=0}^{k-1} \tilde{A}^i Nv_{k-i}. \quad (\text{B.52})$$

Note that there exist common terms between v_{k-1-i} and v_{k-i} . After combining them, we reach the following recursive form:

$$e_k = \tilde{A}^k e_0 + \sum_{i=0}^{k-1} \tilde{A}^i T w_{k-1-i} - \tilde{A}^{k-1} K v_0 - N v_k - \sum_{i=0}^{k-2} \tilde{A}^i (\tilde{A}N + K) v_{k-1-i}, \quad \forall k \geq 2. \quad (\text{B.53})$$

Replacing all realizations by the corresponding zonotopes, we obtain

$$\hat{\mathcal{X}}_k^z = \tilde{A}^k \hat{\mathcal{X}}_0^z \oplus \bigoplus_{i=0}^{k-1} \tilde{A}^i T \mathcal{W} \oplus (-\tilde{A}^{k-1} K \mathcal{V}) \oplus (-N \mathcal{V}) \oplus \bigoplus_{i=0}^{k-2} (-\tilde{A}^i (\tilde{A}N + K) \mathcal{V}), \quad \forall k \geq 2. \quad (\text{B.54})$$

From Property 2.2.1, we have

$$\square \hat{\mathcal{X}}_k^z = \square \{\tilde{A}^k \hat{\mathcal{X}}_0^z\} \oplus \bigoplus_{i=0}^{k-1} \square \{\tilde{A}^i T \mathcal{W}\} \oplus \square \{-\tilde{A}^{k-1} K \mathcal{V}\} \oplus \square \{-N \mathcal{V}\} \oplus \bigoplus_{i=0}^{k-2} \square \{-\tilde{A}^i (\tilde{A}N + K) \mathcal{V}\}, \quad \forall k \geq 2. \quad (\text{B.55})$$

From the row-sum operator defined in (2.26), we have

$$\zeta_k = \text{rs}(\tilde{A}^k \hat{\mathcal{G}}_0^x) + \sum_{i=0}^{k-1} \text{rs}(\tilde{A}^i T \mathcal{G}^w) + \text{rs}(-\tilde{A}^{k-1} K \mathcal{G}^v) + \text{rs}(-N \mathcal{G}^v) + \sum_{i=0}^{k-2} \text{rs}(-\tilde{A}^i (\tilde{A}N + K) \mathcal{G}^v), \quad \forall k \geq 2. \quad (\text{B.56})$$

Finally, the interval hull is given by

$$\square \hat{\mathcal{X}}_k^z = [-\zeta_k, \zeta_k]. \quad (\text{B.57})$$

Next, the H_∞ ZF is presented in Algorithm B.4, which solves Problem 1.2.3 enforcing state linear equality constraints. Its complexity order is given by

$$O(H_\infty \text{ZF}) = O\left(n^2 \left(n + p + n_g + n_g^w + n_g^v + m\right) + n m n_g^v\right).$$

If the cost of the first two time steps are disregarded ($k = 1$ and $k = 2$), and $\hat{x}_k$ in (B.13) is computed with the fixed matrices $\tilde{A} = (TA - KC)$ and $\tilde{D} = TB$, then the computational cost of H_∞ ZF becomes $O\left(n^2 \left(n_g + n_g^w + n_g^v\right) + n(p + m)\right)$, which is the same value when $T = I_n$ and $N = 0_{n \times m}$.

B.1.4 Constrained Zonotopic Linear Filter

The set-membership linear filter based on CZs for LTV systems (CZLF) is proposed in Scott et al. (2016). This paper introduced the class of CZs to complete the link between zonotopes and convex polytopes using explicit calculus. On the one hand, the zonotope bundles proposed in Althoff & Krogh (2011) are already able to represent convex polytopes by using a list of zonotopes. However, zonotopes are not explicitly intersected in zonotope bundles, and therefore, this type of set has been restricted to computation of reachable

Algorithm B.4: $[\hat{x}_k, \mathcal{I}_k, \Pi_k^x, \Pi_k^w, \zeta_k^w, \Pi_k^{va}, \Pi_k^{vc}, \zeta_k^{vc}] = \text{H}_\infty \text{ZF}(\hat{x}_{k-1}, \Pi_{k-1}^x, \Pi_{k-1}^w, \zeta_{k-1}^w, \Pi_{k-1}^{va}, \zeta_{k-1}^{vb}, \Pi_{k-1}^{vc}, \zeta_{k-1}^{vc}, T, N, K, \tilde{A}, B, u_{k-1}, \mathcal{W}, \mathcal{V}, y_{k-1}, y_k, k)$ (Tang et al., 2019a)

```

1 if  $k = 1$  then
2   Input:  $\Pi_0^x = \hat{G}_0^x, \Pi_0^w = \Pi_0^{va} = \Pi_0^{vc} = \zeta_0^w = \zeta_0^{vc} = \emptyset, \zeta^{vb} = \text{rs}(-NG^v)$ , where  $\text{rs}$  is given
   by (2.26), and  $\emptyset$  represents an empty set
3   Update:  $\Pi_1^x = \tilde{A}\Pi_0^x, \Pi_1^w = TG^w, \Pi_1^{va} = -KG^v, \Pi_1^{vc} = \emptyset, \zeta_1^w = \text{rs}(\Pi_1^w), \zeta_1^{vc} = 0_{n \times 1}$ 
4 else if  $k = 2$  then
5    $\Pi_2^x = \tilde{A}\Pi_1^x, \Pi_2^w = \tilde{A}\Pi_1^w, \Pi_2^{va} = \tilde{A}\Pi_1^{va}, \Pi_2^{vc} = -(\tilde{A}N + K)G^v, \zeta_2^w = \text{rs}(\Pi_2^w) + \zeta_1^w,$ 
    $\zeta_2^{vc} = \text{rs}(\Pi_2^{vc}) + \zeta_1^{vc}$ 
6 else
7    $\Pi_k^x = \tilde{A}\Pi_{k-1}^x, \Pi_k^w = \tilde{A}\Pi_{k-1}^w, \Pi_k^{va} = \tilde{A}\Pi_{k-1}^{va}, \Pi_k^{vc} = \tilde{A}\Pi_{k-1}^{vc}, \zeta_k^w = \text{rs}(\Pi_k^w) + \zeta_{k-1}^w,$ 
    $\zeta_k^{vc} = \text{rs}(\Pi_k^{vc}) + \zeta_{k-1}^{vc}$ 
8 end
9  $\zeta_k = \text{rs}(\Pi_k^x) + \zeta_k^w + \text{rs}(\Pi_k^{va}) + \zeta^{vb} + \zeta_k^{vc}$ 
10 Compute the center  $\hat{x}_k$  in (B.13)
11  $\mathcal{I}_k = [\hat{x}_k - \zeta_k, \hat{x}_k + \zeta_k]$ 

```

sets.

Consider the discrete-time LTV uncertain system described in (B.1)-(B.2) whose uncertainties are characterized by CZs. The idea is to intersect the forecast set $\hat{\mathcal{X}}_{k|k-1}$ with the correction set $\hat{\mathcal{Y}}_{k|k-1}$ using Property 2.2.8 to yield the corrected set $\hat{\mathcal{X}}_k$. By employing Properties 2.2.6-2.2.7, we have

$$\begin{aligned}
\hat{\mathcal{X}}_{k|k-1} &= A_{k-1}\hat{\mathcal{X}}_{k-1} \oplus B_{k-1}u_{k-1} \oplus \mathcal{W}_{k-1} \\
&= \left\{ \left[A_{k-1}\hat{G}_{k-1}^x \quad G_{k-1}^w \right], (A_{k-1}\hat{x}_{k-1} + B_{k-1}u_{k-1}), \begin{bmatrix} \hat{A}_{k-1}^x & 0_{n_h \times n_g^w} \\ 0_{n_h^w \times n_g} & A_{k-1}^w \end{bmatrix}, \begin{bmatrix} \hat{b}_{k-1}^x \\ b_{k-1}^w \end{bmatrix} \right\} \\
&= \left\{ \hat{G}_{k|k-1}^x, \hat{x}_{k|k-1}, \hat{A}_{k|k-1}^x, \hat{b}_{k|k-1}^x \right\},
\end{aligned} \tag{B.58}$$

$$\begin{aligned}
\hat{\mathcal{Y}}_{k|k-1} &= y_k \oplus (-\mathcal{V}_k) \\
&= \{-G_k^v, y_k, A_k^v, b_k^v\}.
\end{aligned} \tag{B.59}$$

Since $\hat{\mathcal{Y}}_{k|k-1} = C_k\hat{\mathcal{X}}_{k|k-1}$, Property 2.2.8 implies

$$\check{\mathcal{X}}_k = \hat{\mathcal{X}}_{k|k-1} \cap_{C_k} \hat{\mathcal{Y}}_{k|k-1} \tag{B.60}$$

$$\begin{aligned}
&= \left\{ \left[\hat{G}_{k|k-1}^x \quad 0_{n \times n_g^v} \right], \hat{x}_{k|k-1}, \begin{bmatrix} \hat{A}_{k|k-1}^x & 0_{(n_h+n_h^w) \times n_g^v} \\ 0_{n_h^v \times (n_g+n_g^w)} & A_k^v \\ C_k\hat{G}_{k|k-1}^x & G_k^v \end{bmatrix}, \begin{bmatrix} \hat{b}_{k|k-1}^x \\ b_k^v \end{bmatrix} \right\} \\
&= \{\check{G}_k^x, \check{x}_k, \check{A}_k^x, \check{b}_k^x\}.
\end{aligned} \tag{B.61}$$

Finally, CZR in (2.31) is applied to $\check{\mathcal{X}}_k$ to reduce its order of constraint $\tilde{n}_h = (n_h + n_h^w + n_h^v + m)$ and generator $\tilde{n}_g = (n_g + n_g^w + n_g^v)$ to φ_c and φ_g , respectively, yielding the desired set $\hat{\mathcal{X}}_k$.

Next, the steps to execute one loop of CZLF is summarized in Algorithm B.5, whose complexity order is given by

$$\begin{aligned} O(\text{CZLF}) &= O\left(n^2 n_g + np + mn_g^v + nm(n_g + n_g^w)\right) \\ &\quad + k_c (\tilde{n}_h + \tilde{n}_g)^3 + k_c n \tilde{n}_g^2 + (n + \varphi_c)^2 (\tilde{n}_g - k_c) + k_g (\tilde{n}_g - k_c)(n + \varphi_c), \end{aligned}$$

where $k_c = \tilde{n}_h - \varphi_c$ and $k_g = \tilde{n}_g - k_c - \varphi_g$.

Algorithm B.5: $\hat{\mathcal{X}}_k = \text{CZLF}(\hat{\mathcal{X}}_{k-1}, A_{k-1}, B_{k-1}, \mathcal{W}_{k-1}, C_k, \mathcal{V}_k, u_{k-1}, y_k, \varphi_c, \varphi_g)$ (Scott et al., 2016)

- 1 Determine the CZ $\check{\mathcal{X}}_k$ using (B.58) and (B.60)
 - 2 Compute $\hat{\mathcal{X}}_k = \text{CZR}(\check{\mathcal{X}}_k, \varphi_c, \varphi_g)$, where CZR is given by (2.31)
-

B.2 Nonlinear Algorithms

Consider the discrete-time uncertain nonlinear dynamical system

$$x_k = f(x_{k-1}, u_{k-1}, w_{k-1}), \quad (\text{B.62})$$

$$y_k = h(x_k, v_k), \quad (\text{B.63})$$

where $f: \mathbb{R}^n \times \mathbb{R}^p \times \mathbb{R}^q \rightarrow \mathbb{R}^n$ and $h: \mathbb{R}^n \times \mathbb{R}^r \rightarrow \mathbb{R}^m$ are the known process and measurement models, respectively, and $x_k \in \mathbb{R}^n$ is the state vector to be estimated. The terms $w_{k-1} \in \mathbb{R}^q$ and $v_k \in \mathbb{R}^r$ represent process and measurement noise terms, respectively, whereas $x_0 \in \mathbb{R}^n$ represents the initial state; they are characterized by the sets $\mathcal{W}_{k-1} \subset \mathbb{R}^q$, $\mathcal{V}_k \subset \mathbb{R}^r$, and $\hat{\mathcal{X}}_0 \subset \mathbb{R}^n$, respectively. For $k \in \mathbb{Z}_+$, the measurement vector $y_k \in \mathbb{R}^m$, the input vector $u_{k-1} \in \mathbb{R}^p$, the functions f and h , and the aforementioned sets are assumed to be known. For the zonotopic algorithms, these sets correspond to the zonotopes $\mathcal{W}_{k-1} = \{G_{k-1}^w, 0_{q \times 1}\}$, $\mathcal{V}_k = \{G_k^v, 0_{r \times 1}\}$, and $\hat{\mathcal{X}}_0 = \{\hat{G}_0^x, \hat{x}_0\}$.

The goal is to solve Problems 1.2.2 and 1.2.3 to estimate the *a posteriori* zonotope $\hat{\mathcal{X}}_k = \{\hat{G}_k^x, \hat{x}_k\}$ based on some performance criterion, such as the volume (1.7) and *F-radius* (1.8).

B.2.1 Zonotopic Extended Kalman Predictor

The zonotopic extended Kalman predictor (ZEKP) is an approximated extension of the ZKP (Algorithm B.2) for nonlinear systems. It is similar to the EKF regarding the application of analytical linearization of the model. The main differences between them are: (i) uncertainties are characterized by sets rather than random vectors; and (ii) a predictor is proposed instead of filter. As the EKF, the ZEKP proposed by Wang & Puig

(2016) disregards the linearization remainder. In addition, the ZEKP considers additive noise terms, that is,

$$x_k = f(x_{k-1}, u_{k-1}) + w_{k-1}, \quad (\text{B.64})$$

$$y_k = h(x_k) + v_k. \quad (\text{B.65})$$

From (B.64)-(B.65), the following analytically linearized model is yielded:

$$x_k \approx f(\hat{x}_{k-1}, u_{k-1}) + F_{k-1}^x (x_{k-1} - \hat{x}_{k-1}) + w_{k-1}, \quad (\text{B.66})$$

$$y_{k-1} \approx h(\hat{x}_{k-1}) + H_{k-1}^x (x_{k-1} - \hat{x}_{k-1}) + v_{k-1}, \quad (\text{B.67})$$

where the center estimates $\hat{x}_{k|k-1}$ and $\hat{y}_{k-1}$ are given by, respectively,

$$\hat{x}_{k|k-1} = f(\hat{x}_{k-1}, u_{k-1}), \quad (\text{B.68})$$

$$\hat{y}_{k-1} = h(\hat{x}_{k-1}), \quad (\text{B.69})$$

with $F_{k-1}^x \in \mathbb{R}^{n \times n}$ and $H_k^x \in \mathbb{R}^{m \times n}$ being Jacobian matrices. The remaining steps to build the observer are equal to the ZKP ones, in which F_{k-1} and H_{k-1} are treated as system matrices. Then, the ZEKP can be seen as the most intuitive algorithm to locally apply the ZKP to nonlinear systems.

Next, the ZEKP is presented in Algorithm B.6, which solves Problem 1.2.3 disregarding state constraints. Assuming that the evaluation of functions $f(\cdot)$ and $h(\cdot)$ is $O(1)$, $O(\text{ZEKP}) \approx O(\text{ZKP})$.

Algorithm B.6: $\hat{\mathcal{X}}_k = \text{ZEKP}(\hat{\mathcal{X}}_{k-1}, \mathcal{W}_{k-1}, \mathcal{V}_{k-1}, u_{k-1}, y_{k-1}, f, h, \varphi_g)$ (Wang & Puig, 2016)

- 1 Compute $\hat{\mathcal{X}}_{k-1}^\downarrow = \text{ZR}(\hat{\mathcal{X}}_{k-1}, \varphi_g)$, where ZR is given by Algorithm B.1
 - 2 Determine the Jacobian matrices $F_{k-1} = \nabla_x f(\hat{x}_{k-1}, u_{k-1})$ and $H_{k-1} = \nabla_x h(\hat{x}_{k-1})$
 - 3 Compute the centers $\hat{x}_{k|k-1}$ in (B.68) and $\hat{y}_{k-1}$ in (B.69), and the covariation matrices $\hat{\mathcal{C}}_{k-1}^{xx}$ in (B.6), $\hat{\mathcal{C}}_{k-1}^{xy}$ in (B.7), $\mathcal{C}_{k-1}^{vv}$ in (B.8), and $\hat{\mathcal{C}}_{k-1}^{yy}$ in (B.9)
 - 4 Compute the gain matrix (B.5), and thereby, the *a posteriori* generator matrix $\hat{\mathcal{G}}_k^x$ and center $\hat{x}_k$ estimates given by (B.4) and (A.5), respectively, to obtain $\hat{\mathcal{X}}_k = \{\hat{\mathcal{G}}_k^x, \hat{x}_k\}$
-

B.2.2 Zonotopic Filter based on Mean Value Extension

B.2.2.1 Intersections among Strips and Zonotopes

We revisit three algorithms to compute the intersection $\bar{\mathcal{X}}$ based on the vector $\kappa \in \mathbb{R}^n$. Two of them are proposed by Alamo et al. (2005), while the latter is proposed by Bravo et al. (2006). First, we review the segment minimization (SM) approach (1.6) in Algorithm

B.7, whose optimal design vector is given by

$$\check{\kappa} = \arg \min_{\kappa} \|\bar{G}^x(\kappa)\|_F. \quad (\text{B.70})$$

Its complexity order is given by $O(\text{SM}) = O(n^2 n_g)$.

Algorithm B.7: $\bar{\mathcal{X}} = \text{SM}(\mathcal{X}, \mathcal{S})$ (Alamo et al., 2005)

- 1 Compute $\check{\kappa} = \frac{G^x(G^x)^\top c}{c^\top G^x(G^x)^\top c + \rho^2}$
 - 2 Compute the zonotope $\bar{\mathcal{X}} = \{\bar{G}^x, \bar{x}\}$, whose generator matrix $\bar{G}^x$ and center $\bar{x}$ are given by (2.24) and (2.25), respectively
-

Another possibility to perform the intersection is the volume minimization (VM) of the zonotope $\mathcal{X}_k$ in Algorithm B.8, whose optimal vector is given by

$$\check{\kappa} = \arg \min_{\kappa} \text{Vol}(\bar{\mathcal{X}}(\kappa)). \quad (\text{B.71})$$

In this case, a convex optimization problem must be online solved (Bravo et al., 2004), whose computational cost depends on the employed solver. To reduce the processing time of the VM1, Bravo et al. (2006) proposed an improvement in which the desired vector κ is parametrized. In doing so, the optimization problem is replaced by the direct computation of $(n_g + 1)$ candidate zonotopes $\bar{\mathcal{X}}(j) = \bar{G}(j)\mathcal{B}^{n_g} \oplus \bar{x}(j)$, for $j = 0, 1, \dots, n_g$. This improvement is presented in Algorithm B.9. Its complexity order is given by $O(\text{VM2}) = O(n_g(nn_g + N(n_g, n)n^3))$, where $N(n_g, n) = \frac{n_g!}{n!(n_g - n)!}$.

Algorithm B.8: $\bar{\mathcal{X}} = \text{VM1}(\mathcal{X}, \mathcal{S})$ (Alamo et al., 2005)

- 1 Compute $\check{\kappa} = \arg \min_{\kappa} \left\{ 2^n \sum_{i=1}^{N(n_g, n)} |\det[(I_n - \kappa c^\top) T_i^a]| + 2^n \sum_{i=1}^{N(n_g, n-1)} |\det[(I_n - \kappa c^\top) T_i^b \quad \rho \kappa]| \right\}$,
where $T_i^a \in \mathbb{R}^{n \times n}$ denotes all matrices that can be obtained taking n columns from the matrix G^x , and $T_i^b \in \mathbb{R}^{n \times (n-1)}$ denotes all matrices that can be obtained taking $(n-1)$ columns from the matrix G^x
 - 2 Compute the zonotope $\bar{\mathcal{X}} = \{\bar{G}^x, \bar{x}\}$, whose generator matrix $\bar{G}^x$ and center $\bar{x}$ are given by (2.24) and (2.25), respectively
-

In Subsection 3.2.1.2, we propose a modification for Algorithm B.8 to improve its efficiency. Among the three algorithms, the SM given by Algorithm B.7 is the fastest. However, it yields the most conservative zonotope. In addition, the VM1 given by Algorithm B.8 is the most precise. Conversely, it demands the largest computational burden. Recall that the VM2 given by Algorithm B.9 yields the zonotope $\bar{\mathcal{X}}$ whose order of generators is the same one of $\mathcal{X}$. This occurs because a specific structure to the vector κ is defined, in which a column of $\bar{G}^x$ is canceled. In doing so, $(n_g + 1)$ zonotopes are made

Algorithm B.9: $\bar{\mathcal{X}} = \text{VM2}(\mathcal{X}, \mathcal{S})$ (Bravo et al., 2006)

```

1  $\bar{G}^x = G^x$ 
2  $\bar{x} = \hat{x}$ 
3  $\psi = \text{Vol}(\bar{G}^x \mathcal{B}^{n_g} \oplus \bar{x})$ , where Vol is given by (1.7)
4 for  $j = 1, \dots, n_g$  do
5   if  $c^\top G_{:,j}^x \neq 0$  then
6      $A = G^x - \left(1_{n \times 1} \frac{c^\top G^x}{c^\top G_{:,j}^x}\right) \odot \left(G_{:,j}^x 1_{1 \times n_g}\right)$ , where  $\odot$  represents the elementwise product
       between matrices with same dimension
7      $A_{:,j} = \frac{\rho}{c^\top G_{:,j}^x} G_{:,j}^x$ 
8      $\bar{a} = \hat{x} + \frac{(d - c^\top \hat{x})}{c^\top G_{:,j}^x} G_{:,j}^x$ 
9      $\gamma = \text{Vol}(A \mathcal{B}^{n_g} \oplus \bar{a})$ 
10    if  $\gamma < \psi$  then
11       $\bar{G}^x = A$ 
12       $\bar{x} = \bar{a}$ 
13       $\psi = \gamma$ 
14    end
15  end
16 end
17  $\bar{\mathcal{X}} = \{\bar{G}^x, \bar{x}\}$ 

```

explicit and indexed by $j = 0, \dots, n_g$. Since we choose the best index j^* according to the smallest volume (1.7), the VM2 reduces the computational burden of the VM1.

B.2.2.2 Zonotope Nonlinear Transformation

From the mean value extension (Proposition 2.2.1) and zonotope inclusion (Proposition 2.2.2), it is possible to propagate the generator matrix G^x and the center $\hat{x}$ of the zonotope $\mathcal{X}$ through a nonlinear transformation. As in the unscented transformation (Subsection A.2.1.1), consider two variables $x_1 \in \mathbb{R}^n$ and $x_2 \in \mathbb{R}^r$, which represent the possible values of the zonotopes $\mathcal{X}_1 = G^{x_1} \mathcal{B}^{n_{x_1}} \oplus \hat{x}_1 \subset \mathbb{R}^n$ and $\mathcal{X}_2 = G^{x_2} \mathcal{B}^{n_{x_2}} \oplus \hat{x}_2 \subset \mathbb{R}^r$, respectively. In addition, consider a deterministic vector $d \in \mathbb{R}^p$. Next, we describe three procedures to compute the nonlinear transformation over zonotopes and identify one of them as the most precise.

For the nonlinear transformation $y = h(x, d)$, the mean value extension is performed for a given set $\mathcal{X}$. As in Combastel (2005), given the zonotopes $\mathcal{X}_1$ and $\mathcal{X}_2$, define the augmented zonotope:

$$\mathcal{X} \triangleq \mathcal{X}_1 \times \mathcal{X}_2 = \begin{bmatrix} G^{x_1} & 0_{n \times n_{x_2}} \\ 0_{r \times n_{x_1}} & G^{x_2} \end{bmatrix} \begin{bmatrix} \mathcal{B}^{n_{x_1}} \\ \mathcal{B}^{n_{x_2}} \end{bmatrix} \oplus \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix}. \quad (\text{B.72})$$

Given $\mathcal{X} \subseteq \mathbb{R}^{(n+r)}$ and the deterministic vector $d \in \mathbb{R}^p$, the mean value extension (Proposition

(2.2.1) is applied to $h(\mathcal{X}, d)$, yielding

$$h(\mathcal{X}, d) \subseteq h(\hat{x}, d) \oplus [M](\mathcal{X} - \hat{x}), \quad (\text{B.73})$$

where

$$[M] \triangleq \square \{ \nabla_x h(\mathcal{X}, d) \} \quad (\text{B.74})$$

is the interval matrix obtained by applying the natural interval extension (Definition 2.2.15). Next, the zonotope inclusion (Proposition 2.2.2) is applied to (B.73) to obtain the transformed zonotope $\mathcal{Y} = G^y \mathcal{B}^{n_y} \oplus \hat{y}$, such that

$$\mathcal{Y} = \diamond \{ [M] G^x \mathcal{B}^{n_x} \oplus h(\hat{x}, d) \}. \quad (\text{B.75})$$

This result is described in Algorithm B.10, whose complexity order is given by $O(\text{ZT1}) = O(m(n+r)(n_g^{x_1} + n_g^{x_2}))$.

Algorithm B.10: $\mathcal{Y} = \text{ZT1}(\mathcal{X}_1, \mathcal{X}_2, d, h)$

- 1 Given the zonotopes $\mathcal{X}_1 = G^{x_1} \mathcal{B}^{n_{x_1}} \oplus \hat{x}_1$ and $\mathcal{X}_2 = G^{x_2} \mathcal{B}^{n_{x_2}} \oplus \hat{x}_2$, define the augmented zonotope $\mathcal{X} = G^x \mathcal{B}^{n_x} \oplus \hat{x}$ in (B.72), where $n_x = n_{x_1} + n_{x_2}$
 - 2 Given the deterministic vector $d \in \mathbb{R}^p$, calculate the center $\hat{y} = h(\hat{x}, d)$
 - 3 Use interval arithmetic to compute the interval matrix $[M]$ in (B.74)
 - 4 Apply Proposition 2.2.2 to $[M] G^x \mathcal{B}^{n_x} \oplus \hat{y}$ to obtain the zonotope $\mathcal{Y} = G^y \mathcal{B}^{(m+n_x)} \oplus \hat{y}$ in (B.75)
-

Now, we review the second procedure for which the extension from $y = h(x, d)$ to $y = h(x_1, x_2, d)$ is used and the mean value extension is applied to approximate the function $h : \mathbb{R}^n \times \mathbb{R}^r \times \mathbb{R}^p \rightarrow \mathbb{R}^m$ with respect to one variable at time. Given the zonotopes $\mathcal{X}_1$ and $\mathcal{X}_2$ and the deterministic vector $d \in \mathbb{R}^p$, the mean value extension (Proposition 2.2.1) is applied to $h(\mathcal{X}_1, \mathcal{X}_2, d)$, resulting in

$$h(\mathcal{X}_1, \mathcal{X}_2, d) \subseteq h(\hat{x}_1, \mathcal{X}_2, d) \oplus [M_1](\mathcal{X}_1 - \hat{x}_1), \quad (\text{B.76})$$

where

$$[M_1] \triangleq \square \{ \nabla_{x_1} h(\mathcal{X}_1, \mathcal{X}_2, d) \} \quad (\text{B.77})$$

is the interval matrix given by the natural interval extension (Definition 2.2.15). Again, the mean value extension is applied to $h(\hat{x}_1, \mathcal{X}_2, d)$, which leads to

$$h(\hat{x}_1, \mathcal{X}_2, d) \subseteq h(\hat{x}_1, \hat{x}_2, d) \oplus [M_2](\mathcal{X}_2 - \hat{x}_2), \quad (\text{B.78})$$

where

$$[M_2] \triangleq \square \{ \nabla_{x_2} h(\hat{x}_1, \mathcal{X}_2, d) \}. \quad (\text{B.79})$$

Next, the following set is built as

$$h(\mathcal{X}_1, \mathcal{X}_2, d) \subseteq h(\hat{x}_1, \hat{x}_2, d) \oplus \begin{bmatrix} [M_1]G^{x_1} & [M_2]G^{x_2} \end{bmatrix} \begin{bmatrix} \mathcal{B}_s^{n_s^{x_1}} \\ \mathcal{B}_s^{n_s^{x_2}} \end{bmatrix}. \quad (\text{B.80})$$

Finally, the zonotope inclusion (Proposition 2.2.2) is applied to (B.80) to obtain the transformed zonotope $\mathcal{Y} = G^y \mathcal{B}_s^{n_s^y} \oplus \hat{y}$, such that

$$\mathcal{Y} = \diamond \left\{ \begin{bmatrix} [M_1]G^{x_1} & [M_2]G^{x_2} \end{bmatrix} \begin{bmatrix} \mathcal{B}_s^{n_s^{x_1}} \\ \mathcal{B}_s^{n_s^{x_2}} \end{bmatrix} \oplus h(\hat{x}_1, \hat{x}_2, d) \right\}. \quad (\text{B.81})$$

This result is described in Algorithm B.11, whose complexity order is given by $O(\text{ZT2}) = O(m(nn_s^{x_1} + rn_s^{x_2}))$.

Algorithm B.11: $\mathcal{Y} = \text{ZT2}(\mathcal{X}_1, \mathcal{X}_2, d, h)$ (Alamo et al., 2005)

- 1 Given the zonotopes $\mathcal{X}_1 = G^{x_1} \mathcal{B}_s^{n_s^{x_1}} \oplus \hat{x}_1$ and $\mathcal{X}_2 = G^{x_2} \mathcal{B}_s^{n_s^{x_2}} \oplus \hat{x}_2$, and the deterministic vector $d \in \mathbb{R}^p$, compute the center $\hat{y} = h(\hat{x}_1, \hat{x}_2, d)$
 - 2 Use interval arithmetic to calculate the interval matrices $[M_1]$ in (B.77) and $[M_2]$ in (B.79)
 - 3 Apply Proposition 2.2.2 to $[M_1]G^{x_1} \mathcal{B}_s^{n_s^{x_1}} \oplus [M_2]G^{x_2} \mathcal{B}_s^{n_s^{x_2}} \oplus \hat{y}$ to obtain the zonotope $\mathcal{Y} = G^y \mathcal{B}_s^{(m+n_s^{x_1}+n_s^{x_2})} \oplus \hat{y}$ in (B.81)
-

Finally, we review the third procedure, where the mean value extension is combined with the natural interval extension. Given the zonotopes $\mathcal{X}_1$ and $\mathcal{X}_2$ and the deterministic vector $d \in \mathbb{R}^p$, the mean value extension (Proposition 2.2.1) is applied to $h(\mathcal{X}_1, \mathcal{X}_2, d)$, resulting in (B.76). Next, the natural interval extension (Definition 2.2.15) is applied to $h(\hat{x}_1, \mathcal{X}_2, d)$ to obtain

$$h(\hat{x}_1, \mathcal{X}_2, d) \subseteq \hat{y} \oplus G^a \mathcal{B}^m, \quad (\text{B.82})$$

where

$$\hat{y} = \text{mid}(\square h(\hat{x}_1, \mathcal{X}_2, d)), \quad (\text{B.83})$$

$$G^a = \text{diag}(\text{rad}(\square h(\hat{x}_1, \mathcal{X}_2, d))). \quad (\text{B.84})$$

The following set is built as

$$h(\mathcal{X}_1, \mathcal{X}_2, d) \subseteq \hat{y} \oplus G^a \mathcal{B}^m \oplus [M]G^{x_1} \mathcal{B}_s^{n_s^{x_1}}, \quad (\text{B.85})$$

where $[M]$ is given by (B.77). Next, the zonotope inclusion (Proposition 2.2.2) is applied to $[M]G^{x_1} \mathcal{B}_s^{n_s^{x_1}} \oplus 0_{m \times 1}$ to obtain the zonotope $\mathcal{X}_3 = G^{x_3} \mathcal{B}_s^{(m+n_s^{x_1})} \oplus 0_{m \times 1}$, such that

$$\mathcal{X}_3 = \diamond \{ [M]G^{x_1} \mathcal{B}_s^{n_s^{x_1}} \oplus 0_{m \times 1} \}. \quad (\text{B.86})$$

Finally, the transformed zonotope $\mathcal{Y} = G^y \mathcal{B}_s^{n_y} \oplus \hat{y}$ is obtained, such that

$$\mathcal{Y} = \begin{bmatrix} G^a & G^{x_3} \end{bmatrix} \mathcal{B}^{(2m+n_s^{x_1})} \oplus \hat{y}. \quad (\text{B.87})$$

This result is described in Algorithm B.12, whose complexity order is given by $O(\text{ZT3}) = O(mnn_s^{x_1})$.

Algorithm B.12: $\mathcal{Y} = \text{ZT3}(\mathcal{X}_1, \mathcal{X}_2, d, h)$ (Alamo et al., 2005)

- 1 Given the zonotopes $\mathcal{X}_1 = G^{x_1} \mathcal{B}_s^{n_{x_1}} \oplus \hat{x}_1$ and $\mathcal{X}_2 = G^{x_2} \mathcal{B}_s^{n_{x_2}} \oplus \hat{x}_2$, and the deterministic vector $d \in \mathbb{R}^p$, use interval arithmetic to compute the generator matrix G^a in (B.84) and the center $\hat{y}$ in (B.83)
 - 2 Use interval arithmetic to compute the interval matrix $[M]$ in (B.77)
 - 3 Apply Proposition 2.2.2 to $[M]G^{x_1} \mathcal{B}_s^{n_{x_1}} \oplus 0_{m \times 1}$ to obtain the zonotope $\mathcal{X}_3$ in (B.86)
 - 4 Obtain the zonotope $\mathcal{Y} = G^y \mathcal{B}_s^{n_y} \oplus \hat{y}$ in (B.87)
-

From Algorithm B.10 to Algorithm B.12, the quantity of interval operations to yield the interval Jacobian matrix related to the variables x_1 and x_2 decreases. Therefore, the wrapping and dependency effects become less expressive. Thus, Algorithm B.12 is the most likely to transform zonotopes. Note that if $\mathcal{X}_2$ is an additive set in Algorithm B.12, then it does not need to be transformed to an interval, as in (B.74) and (B.79). This procedure improves the precision of $\mathcal{Y}$. An example is proposed to illustrate the nonlinear transformation of zonotopes using different methods.

Example B.2.1. Consider the nonlinear transformation from polar to Cartesian coordinates

$$y = h(x_1, x_2) = \begin{bmatrix} x_1 \cos(x_2) & x_1 \sin(x_2) \end{bmatrix}^\top, \quad (\text{B.88})$$

where x_1 is the radius and x_2 is the angle. Consider also the zonotopes $\mathcal{X}_1 = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \mathcal{B}^3 \oplus 3$ and $\mathcal{X}_2 = \begin{bmatrix} 0.5 & 0.5 & 2 \end{bmatrix} \mathcal{B}^3 \oplus 0$, which yield the augmented zonotope $\mathcal{X} = G^x \mathcal{B}^6 \oplus \hat{x}$, where $G^x = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.5 & 0.5 & 2 \end{bmatrix}$ and $\hat{x} = \begin{bmatrix} 3 & 0 \end{bmatrix}^\top$. Algorithms B.10, B.11, and B.12 are employed to transform the zonotopes $\mathcal{X}_1$ and $\mathcal{X}_2$ over (B.88).

In Figure B.2, the zonotopes are illustrated before and after their transformation. In (a), the initial zonotopes (black solid line) are illustrated with the grid of 3,600 points (red points). In (b), the corresponding transformations are presented. The exact image (red points) is reached by transforming each point of the grid. Note that, in the function h , there exist only products between different variables, and the zonotopes $\mathcal{X}_1$ and $\mathcal{X}_2$ are, in fact, intervals. That means that there are no wrapping and dependency effects in the Jacobian matrices. The unique existing overestimation is caused by zonotope inclusion and the natural interval extension, where the wrapping effect is added. As expected, the ZT1 given by Algorithm B.10 leads to the most conservative set $(-\cdot-)$, since it has the

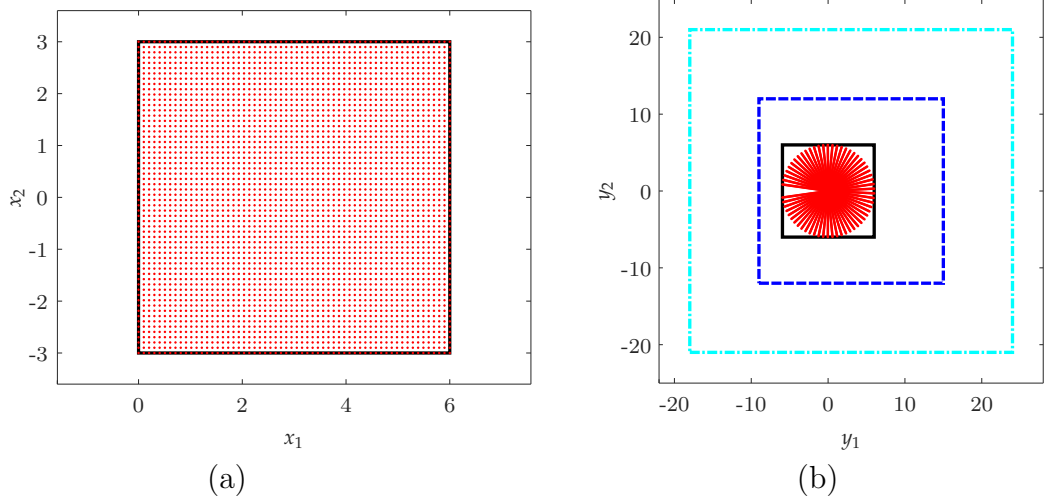


Figure B.2: One-step nonlinear transformation of zonotopes. Let $y = h(x) = [x_1 \cos(x_2) \ x_1 \sin(x_2)]^\top$ be the nonlinear transformation. In addition, consider the zonotopes $\mathcal{X}_1 = [1 \ 1 \ 1] \mathcal{B}^3 \oplus 3$ and $\mathcal{X}_2 = [0.5 \ 0.5 \ 2] \mathcal{B}^3 \oplus 0$. In (a), the zonotopes (black solid line) are illustrated with the grid of 3600 points (red points). In (b), the corresponding ZTs are illustrated. The reference image is generated by transforming each point of the grid (red points), leading to a nonconvex set. The largest set ($-\cdot-$) is computed by the ZT1 given by Algorithm B.10, where the mean value extension is applied only once to h with the augmented zonotope $\mathcal{X}$. The intermediate precision set ($--$) is computed with the ZT2 given by Algorithm B.11, where the mean value extension is twice used with the zonotopes $\mathcal{X}_1$ and $\mathcal{X}_2$ split. The smallest set ($—$) is obtained with the ZT3 given by Algorithm B.12, where the mean value extension is combined with the natural interval extension.

largest amount of intervals in its Jacobian matrix. The ZT2 given by Algorithm B.11 yields an intermediate set ($--$). In turn, the smallest box containing the exact set is provided by the ZT3, given by Algorithm B.12, since it has the smallest amount of intervals in its Jacobian matrix. As the zonotopes $\mathcal{X}_1$ and $\mathcal{X}_2$ are one-dimensional, the natural interval extension of $h(\mathcal{X}_1, \mathcal{X}_2)$ also leads to the smallest overestimated box. In this case, it only introduces wrapping effect, given that the actual set (red points) is overestimated by a box.

B.2.2.3 Algorithm

The zonotopic filter based on the mean value extension (ZMV) proposed in Alamo et al. (2005) is a set-membership observer. By simplicity, it can be summarized in three steps. In the first one, the *a priori* (or forecast) set is estimated by a zonotope $\hat{\mathcal{X}}_{k|k-1}$ using the ZT; see Subsection B.2.2.2 for more details. After, the measurement set is estimated by a strip $\hat{\mathcal{Y}}_{k|k-1}$ as in Definition 2.2.7. In the second step, an intersection is carried out between $\hat{\mathcal{X}}_{k|k-1}$ and $\hat{\mathcal{Y}}_{k|k-1}$ to indirectly extract the parcel of $\hat{\mathcal{X}}_{k|k-1}$ related to $\hat{\mathcal{Y}}_{k|k-1}$, yielding $\bar{\mathcal{X}}$. This zonotope is parametrized by a design vector κ , whose solution is given by the SM or VM; see Subsection B.2.2.1 for more details. In the third step, an order reduction is executed to fix the number of generators of $\bar{\mathcal{X}}$ to a desired amount φ_g .

Next, we present the ZMV in Algorithm B.13, which solves Problem 1.2.2 disregarding state constraints. Assuming that the evaluation of elementary and inclusion functions is $O(1)$, the complexity order of ZMV with SM is given by

$$O(\text{ZMV}) = (n + n_g + m)(mn^2 + \log(n + n_g + m)).$$

Algorithm B.13: $\hat{\mathcal{X}}_k = \text{ZMV}(\hat{\mathcal{X}}_{k-1}, \mathcal{W}_{k-1}, \mathcal{V}_k, u_{k-1}, y_k, \varphi_g, f, h)$ (Alamo et al., 2005)

- 1 $\hat{\mathcal{X}}_{k|k-1} = \text{ZT3}(\hat{\mathcal{X}}_{k-1}, \mathcal{W}_{k-1}, u_{k-1}, f)$, where ZT3 is given by Algorithm B.12
 - 2 $\bar{\mathcal{X}}_{1,k} = \hat{\mathcal{X}}_{k|k-1}$ is the initialization zonotope
 - 3 **for** $i = 1, \dots, m$ **do**
 - 4 $c_i = \text{mid}\left(\square\left\{\nabla_x h_i(\bar{\mathcal{X}}_{i,k}, \mathcal{V}_k)^\top\right\}\right)$ is the direction of a strip
 - 5 $[\Lambda] = \square\left\{c_i^\top \bar{\mathcal{X}}_{i,k} \oplus (-h_i(\bar{\mathcal{X}}_{i,k}, \mathcal{V}_k))\right\}$ is a consistency interval
 - 6 $s_i = \text{mid}([\Lambda])$ is the middle point of $[\Lambda]$
 - 7 $\rho_i = \text{rad}([\Lambda])$ is the width of a strip
 - 8 $d_i = y_{i,k} + s_i$ is the shift of a strip
 - 9 $\hat{\mathcal{Y}}_{i,k|k-1} = \{x \in \mathbb{R}^n : |c_i^\top x - d_i| \leq \rho_i\}$ is the i th strip
 - 10 $\bar{\mathcal{X}}_{i+1,k} = \bar{\mathcal{X}}_{i,k} \cap \hat{\mathcal{Y}}_{i,k|k-1}$, whose performance criterion can be SM or VM discussed in Subsection B.2.2.1
 - 11 **end**
 - 12 $\bar{\mathcal{X}}_k = \bar{\mathcal{X}}_{m+1,k}$ is the output zonotope
 - 13 $\hat{\mathcal{X}}_k = \text{ZR}(\bar{\mathcal{X}}_k, \varphi_g)$, where ZR is given by Algorithm B.1
-



State Estimators based on Mixed Uncertainties: a Review

In this appendix, we review two algorithms based on mixed uncertainties. The first algorithm is for linear systems, whereas the other one is based on nonlinear systems. In both cases, zonotopes are merged with GRVs into a predictor.

C.1 Zonotopic and Gaussian Kalman Predictor

The zonotopic and Gaussian Kalman predictor for LTV systems (ZGKP) is proposed in [Combastel \(2015\)](#). Consider the discrete-time LTV uncertain dynamical system

$$\mathbf{x}_k = A_{k-1}\mathbf{x}_{k-1} + B_{k-1}\mathbf{u}_{k-1} + \mathbf{z}_{k-1}^w + \mathbf{g}_{k-1}^w, \quad (\text{C.1})$$

$$\mathbf{y}_k = C_k\mathbf{x}_k + \mathbf{z}_k^v + \mathbf{g}_k^v, \quad (\text{C.2})$$

where $A_{k-1} \in \mathbb{R}^{n \times n}$, $B_{k-1} \in \mathbb{R}^{n \times p}$, and $C_k \in \mathbb{R}^{m \times n}$ are the system matrices, $\mathbf{u}_{k-1} \in \mathbb{R}^p$ is the input vector, $\mathbf{y}_k \in \mathbb{R}^m$ is the output vector, and $\mathbf{x}_k \in \mathbb{R}^n$ is the state vector to be estimated. We assume that the system matrices and the input and output vectors are known over $k \in \mathbb{Z}_+$.

On the one hand, the set-bounded terms $\mathbf{z}_{k-1}^w \in \mathbb{R}^n$ and $\mathbf{z}_k^v \in \mathbb{R}^m$ belong to the zonotopes $\mathcal{W}_{k-1} = \{G_{k-1}^w, 0_{n \times 1}\}$ and $\mathcal{V}_k = \{G_k^v, 0_{m \times 1}\}$, respectively. On the other hand, the random terms $\mathbf{g}_{k-1}^w \in \mathbb{R}^n$ and $\mathbf{g}_k^v \in \mathbb{R}^m$ are characterized by the GRVs $W_{k-1} \sim \mathcal{N}(0_{n \times 1}, Q_{k-1})$ and $V_k \sim \mathcal{N}(0_{m \times 1}, R_k)$, respectively. We assume that W_{k-1} and V_k define independent stochastic

processes that satisfy $E[W_k W_k^\top] = 0_{n \times n}$ and $E[V_k V_k^\top] = 0_{m \times m}$ for all $k \neq j$. In turn, the term x_0 is assumed to be mixed as in Definition (2.45), that is, $x_0 = c_0^x + z_0^x + g_0^x$. Since the mixed center c_0^x and the exact characterization of both z_0^x and g_0^x are unknown, they are approximated by $\hat{c}_0^x, \hat{z}_0^x \in \hat{X}_0 = \{\hat{G}_0^x, 0_{n \times 1}\}$, and $\hat{g}_0^x$ with representation $\hat{X}_0 \sim \mathcal{N}(0_{n \times 1}, \hat{P}_0^{xx})$. We assume that $\hat{X}_0$ is uncorrelated to W_{k-1} and V_k .

The goal is to solve Problem 1.2.4 disregarding state constraints to obtain the *a posteriori* estimates $\hat{c}_k^x$, $\hat{X}_k$, and $\hat{X}_k$ according to the minimum-variance criterion J_k^{zg} in (1.9). To achieve that, the gain matrix $K_k \in \mathbb{R}^{n \times m}$ must be determined taking both uncertainties into account.

The ZGKP merges the KP (Algorithm A.1) with the ZKP (Algorithm B.2) into a unique algorithm. Due to the superposition principle, the same observer structure presented in (A.5) is considered:

$$\hat{c}_k^x = (A_{k-1} - K_{k-1}C_{k-1})\hat{c}_{k-1}^x + B_{k-1}u_{k-1} + K_{k-1}y_{k-1}, \quad (C.3)$$

whose estimation error $e_k \triangleq x_k - \hat{c}_k^x = \hat{z}_k^x + \hat{g}_k^x$ is given by the parcels

$$\hat{z}_k^x = (A_{k-1} - K_{k-1}C_{k-1})\hat{z}_{k-1}^x + z_{k-1}^w - K_{k-1}z_{k-1}^v, \quad (C.4)$$

$$\hat{g}_k^x = (A_{k-1} - K_{k-1}C_{k-1})\hat{g}_{k-1}^x + g_{k-1}^w - K_{k-1}g_{k-1}^v. \quad (C.5)$$

Replacing each realization by its uncertainty characterization, we have

$$\hat{X}_k = (A_{k-1} - K_{k-1}C_{k-1})\hat{X}_{k-1}^\downarrow \oplus \mathcal{W}_{k-1} \oplus (-K_{k-1})\mathcal{V}_{k-1} = \{\hat{G}_k^x, 0_{n \times 1}\}, \quad (C.6)$$

$$\hat{X}_k = (A_{k-1} - K_{k-1}C_{k-1})\hat{X}_{k-1} + W_{k-1} - K_{k-1}V_{k-1} \sim \mathcal{N}(0_{n \times 1}, \hat{P}_k^{xx}), \quad (C.7)$$

with $\hat{G}_{k-1}^{\downarrow x}$ being the generator matrix of the reduced-order zonotope $\hat{X}_{k-1}^\downarrow = \text{ZR}(\hat{X}_{k-1}, \varphi_g)$. The *a posteriori* error estimates are given by

$$\hat{G}_k^x = [(A_{k-1} - K_{k-1}C_{k-1})\hat{G}_{k-1}^{\downarrow x} \quad G_{k-1}^w \quad -K_{k-1}G_{k-1}^v], \quad (C.8)$$

$$\hat{P}_k^{xx} = (A_{k-1} - K_{k-1}C_{k-1})\hat{P}_{k-1}^{xx}(A_{k-1} - K_{k-1}C_{k-1})^\top + Q_{k-1} + K_{k-1}R_{k-1}K_{k-1}^\top. \quad (C.9)$$

The minimum-variance criterion J_k^{zg} (1.9) merges the *F-radius* of $\hat{G}_k^x$, $\|\hat{G}_k^x\|_F^2$, with the $\text{tr}(\hat{P}_k^{xx})$ through the convex linear combination performed by the parameter $\eta \in [0, 1]$. Therefore, the biobjective optimization problem is transformed in a monobjective optimization problem. To find the optimal gain $\check{K}_{k-1}$, we carry out $\frac{\partial \text{tr}(J_k^{zg}(K_{k-1}))}{\partial K_{k-1}} = 0$, yielding

$$\check{K}_{k-1} = A_{k-1}P_{k-1}^{zg}C_{k-1}^\top (C_{k-1}P_{k-1}^{zg}C_{k-1}^\top + R_{k-1}^{zg})^{-1}, \quad (C.10)$$

where

$$P_{k-1}^{zg} = (1 - \eta) \left(\hat{G}_{k-1}^{lx} \left(\hat{G}_{k-1}^{lx} \right)^\top \right) + \eta \hat{P}_{k-1}^{xx}, \quad (\text{C.11})$$

$$R_{k-1}^{zg} = (1 - \eta) \left(G_{k-1}^v \left(G_{k-1}^v \right)^\top \right) + \eta R_{k-1}. \quad (\text{C.12})$$

Given the *a posteriori* estimates, the confidence box (2.49) is finally computed. The aforementioned steps are presented in Algorithm C.1. Its complexity order is given by

$$O(\text{ZGKP}) = O\left(n_g \left(n + \log(n_g)\right) + n \left(p + mn_g^v + n_g^w\right) + n^2 \left(n + \varphi_g + m\right) + m^2 \left(m + n + n_g^v\right)\right).$$

Algorithm C.1: $[\hat{c}_k^x, \hat{X}_k, \hat{X}_k, \mathcal{I}_k^a] = \text{ZGKP}(\hat{c}_{k-1}^x, \hat{X}_{k-1}, \hat{X}_{k-1}, A_{k-1}, B_{k-1}, C_{k-1}, \mathcal{W}_{k-1}, \mathcal{V}_{k-1}, W_{k-1}, V_{k-1}, u_{k-1}, y_{k-1}, \varphi_g, \eta)$ (Combastel, 2015)

- 1 Compute $\hat{X}_{k-1}^\downarrow = \text{ZR}(\hat{X}_{k-1}, \varphi_g)$, where ZR is given by Algorithm B.1
 - 2 Compute the covariance matrices P_{k-1}^{zg} in (C.11) and R_{k-1}^{zg} in (C.12) to obtain the optimal gain $\check{K}_{k-1}$ in (C.10)
 - 3 Compute the *a posteriori* estimates $\hat{c}_k^x$ in (C.3), $\hat{G}_k^x$ in (C.8), and $\hat{P}_k^{xx}$ in (C.9), yielding $\hat{X}_k = \{\hat{G}_k^x, 0_{n \times 1}\}$ and $\hat{X}_k \sim \mathcal{N}(0_{n \times 1}, \hat{P}_k^{xx})$, and thereby, the confidence box $\mathcal{I}_k^a$ given by (2.49)
-

C.2 Extended Zonotopic and Gaussian Kalman Predictor

The extended zonotopic and Gaussian Kalman predictor (EZGKP) is proposed in Combastel (2016). Consider the discrete-time uncertain nonlinear dynamical system

$$x_k = f(x_{k-1}, u_{k-1}) + w_{k-1}, \quad (\text{C.13})$$

$$y_k = h(x_k) + v_k \quad (\text{C.14})$$

where $f : \mathbb{R}^n \times \mathbb{R}^p \rightarrow \mathbb{R}^n$ and $h : \mathbb{R}^n \rightarrow \mathbb{R}^m$ are the process and measurement models, respectively, and $x_k \in \mathbb{R}^n$ is the state vector to be estimated. We assume that the functions f and h , the input vector $u_{k-1} \in \mathbb{R}^p$, and the measurement vector $y_k \in \mathbb{R}^m$ are known over $k \in \mathbb{Z}_+$.

The terms $w_{k-1} = z_{k-1}^w + g_{k-1}^w$ and $v_k = z_k^v + g_k^v$ represent process and measurement noises, respectively. The terms $z_{k-1}^w \in \mathbb{R}^n$ and $z_k^v \in \mathbb{R}^m$ are characterized by the zonotopes $\mathcal{W}_{k-1}$ and $\mathcal{V}_k$. The terms $g_{k-1}^w \in \mathbb{R}^n$ and $g_k^v \in \mathbb{R}^m$ are assumed to belong to independent stochastic processes and characterized by the random vectors W_{k-1} and V_k with zero mean and covariance $Q_{k-1} \in \mathbb{R}^{n \times n}$ and $R_k \in \mathbb{R}^{m \times m}$, respectively. Moreover, we assume that W_{k-1} and V_k are such that $\mathbb{E}[W_k W_j^\top] = 0_{n \times n}$ and $\mathbb{E}[V_k V_j^\top] = 0_{m \times m}$ for all $k \neq j$. The initial state x_0 is assumed mixed, where $\hat{X}_0$ and $\hat{X}_0$ represent the bounded and stochastic parcels, respectively, with

$\hat{X}_0$ being uncorrelated to W_{k-1} and V_k . As in the linear case, the goal is to solve Problem 1.2.4 disregarding state constraints to estimate the *a posteriori* center $\hat{c}_k^x$, zonotope $\hat{X}_k$, and GRV $\hat{X}_k$ for all $k \geq 1$.

According to the quasi-LPV approach, the nonlinear dynamical system (C.13)-(C.14) is transformed into a quasi-LPV system as follows (Combastel, 2016):

$$x_k = A_{k-1}x_{k-1} + B_{k-1}u_{k-1} + z_{k-1}^{\tilde{w}} + g_{k-1}^{\tilde{w}}, \quad (C.15)$$

$$y_{k-1} = C_{k-1}x_{k-1} + z_{k-1}^{\tilde{v}} + g_{k-1}^{\tilde{v}}, \quad (C.16)$$

where $\tilde{w}_{k-1}$ and $\tilde{v}_k$ are the noises w_{k-1} and v_k augmented to incorporate the corresponding set-bounded and stochastic errors,

$$A_{k-1} = A_{0,k-1} + \Delta_{k-1}, \quad \Delta_{k-1} = \sum_{i=1}^{n_\delta} A_{i,k-1} \delta_{i,k-1}, \quad \delta_{i,k-1} \in [-1, 1],$$

and δ_{k-1} is an unknown, bounded and time-varying parameter, but constant for all possible outcomes at time $k-1$. The matrices B_{k-1} and C_{k-1} are deterministic, while $A_{0,k-1}$, $A_{1,k-1}$, $\dots$, $A_{n_\delta,k-1}$ are known by means of A_{k-1} . The EZGKP was proposed considering correlated process and measurement noises. However, in this work, for simplicity, they are assumed to be uncorrelated.

Since A_{k-1} is uncertain, Combastel (2016) modifies the optimal structure of the ZGKP to incorporate all possible occurrences of the parameter $\delta_{k-1} \in [-1, 1]^{n_\delta}$. In short, the modification is to separate the contribution of Δ_{k-1} from the contribution of $A_{0,k-1}$ in both the center $\hat{c}_k^x$ and gain matrix $\check{K}_{k-1}$ given by (C.3) and (C.10), respectively. This procedure allows for developing a suboptimal state estimator where both explicit calculus and punctual center estimate are preserved.

Firstly, the optimal gain (C.10) is rewritten as

$$\check{K}_{k-1} = A_{k-1}T, \quad T = P_{k-1}^{zg} C_{k-1}^\top (C_{k-1} P_{k-1}^{zg} C_{k-1}^\top + R_{k-1}^{zg})^{-1}, \quad (C.17)$$

where P_{k-1}^{zg} and R_{k-1}^{zg} are given by (C.11) and (C.12), respectively. For

$$N^x \triangleq I_n - TC_{k-1}, \quad M^x \triangleq A_{0,k-1}N^x, \quad (C.18)$$

the suboptimal observer structure is given by (Combastel, 2016)

$$\hat{c}_k^x = M^x \hat{c}_{k-1}^x + B_{k-1}u_{k-1} + A_{0,k-1}Ty_{k-1}, \quad (C.19)$$

$$\hat{z}_k^x = M^x \hat{z}_{k-1}^x + z_{k-1}^{\tilde{w}} - A_{0,k-1}Tz_{k-1}^{\tilde{v}} + \Delta(N^x \hat{c}_{k-1}^x + Ty_{k-1} + N^x \hat{z}_{k-1}^x - Tz_{k-1}^{\tilde{v}}), \quad (C.20)$$

$$\hat{g}_k^x = M^x \hat{g}_{k-1}^x + g_{k-1}^{\tilde{w}} - A_{0,k-1}Tg_{k-1}^{\tilde{v}} + \Delta(N^x \hat{g}_{k-1}^x - Tg_{k-1}^{\tilde{v}}). \quad (C.21)$$

In order to estimate the zonotope $\hat{X}_k$ and the GRV $\hat{X}_k$, Combastel (2016) proposes the

following results.

Proposition C.2.1 (Combastel (2016)). In (C.20), $\hat{z}_k^x \in \hat{X}_k = \{\hat{G}_k^x, 0_{n \times 1}\}$ with

$$\Xi \triangleq \begin{bmatrix} n_\delta A_{i,k-1} \left[(N^x \hat{c}_{k-1}^x + T y_{k-1}) & N^x \hat{G}_{k-1}^{lx} & -T G_{k-1}^v \right] \end{bmatrix}, \quad (C.22)$$

$$\hat{G}_k^x = \begin{bmatrix} M^x \hat{G}_{k-1}^{lx} & G_{k-1}^w & -A_{0,k-1} T G_{k-1}^v & \Xi \end{bmatrix}, \quad (C.23)$$

where $[\cdot]_{i=1}^{n_\delta}$ is an iterative concatenation operator indexed by i in which each inner matrix $[\cdot]_{i=1}^{n_\delta}$ is included in Ξ as new columns from left to right.

Proposition C.2.2 (Combastel (2016)). In (C.21), the realizations $\hat{g}_k^x$ are represented by $\hat{X}_k \sim \mathcal{N}(0_{n \times 1}, \hat{P}_k^{xx})$ with

$$\hat{P}_k^{xx} = H_0 + \sum_{i=1}^{n_\delta} (\Omega_{1,i} + \Omega_{2,i}) + \sum_{i < j} \Omega_{2,ij}, \quad (C.24)$$

where

$$T^r = T \tilde{R}_{k-1} T^\top, \quad (C.25)$$

$$H_0 = M^x \hat{P}_{k-1}^{xx} (M^x)^\top + \tilde{Q}_{k-1} + A_{0,k-1} T^r A_{0,k-1}^\top, \quad (C.26)$$

$$H_1 = N^x \hat{P}_{k-1}^{xx} (M^x)^\top + T^r A_{0,k-1}^\top, \quad (C.27)$$

$$H_2 = N^x \hat{P}_{k-1}^{xx} (N^x)^\top + T^r, \quad (C.28)$$

$$\Omega_{1,i} = A_{i,k-1} H_1 + H_1^\top A_{i,k-1}^\top, \quad (C.29)$$

$$\Omega_{2,i} = A_{i,k-1} H_2 A_{i,k-1}^\top, \quad (C.30)$$

$$\Omega_{2,ij} = A_{i,k-1} H_2 A_{j,k-1}^\top + A_{j,k-1} H_2 A_{i,k-1}^\top. \quad (C.31)$$

Thereby, the mixed state vector x_k is finally characterized. Next, the EZGKP is described in Algorithm C.2, whose complexity order is given by

$$O(\text{EZGKP}) = O(n_g \log(n_g) + n(n_g + p + n_g^w + mn_\delta n_g^v) + n^2(m + n_\delta \varphi_g + nn_\delta^2 + n_\delta n_g^v) + m^2(n + m + n_g^v)).$$

Remark C.2.1. For the cases in which $\tilde{w}_k$ and $\tilde{v}_k$ are correlated, we do $n_g^w = n_g^v$ to match matrix dimensions. This is the unique modification occurred in the total worst-case complexity order here derived for the EZGKP.

Algorithm C.2: $[\hat{c}_k^x, \hat{\mathcal{X}}_k, \hat{X}_k, \mathcal{I}_k^a] = \text{EZGKP}(\hat{c}_{k-1}^x, \hat{\mathcal{X}}_{k-1}, \hat{X}_{k-1}, A_{0,k-1}, A_{1,k-1}, \dots, A_{n_\delta, k-1}, B_{k-1}, C_{k-1}, \tilde{\mathcal{W}}_{k-1}, \tilde{\mathcal{V}}_{k-1}, \tilde{W}_{k-1}, \tilde{V}_{k-1}, u_{k-1}, y_{k-1}, \varphi_g, \eta, \alpha)$ (Combastel, 2016)

- 1 Compute $\hat{\mathcal{X}}_{k-1}^\downarrow = \text{ZR}(\hat{\mathcal{X}}_{k-1}, \varphi_g)$, where ZR is given by Algorithm B.1
 - 2 Compute the covariance matrices P_{k-1}^{zg} in (C.11) and R_{k-1}^{zg} in (C.12) to obtain the matrices T in (C.17), N^x and M^x in (C.18)
 - 3 Compute the center $\hat{c}_k^x$, generator matrix $\hat{G}_k^x$, and covariance matrix $\hat{P}_k^{\text{xx}}$ estimates given by (C.19), (C.23), and (C.24), respectively, yielding $\hat{\mathcal{X}}_k = \{\hat{G}_k^x, 0_{n \times 1}\}$ and $\hat{X}_k \sim \mathcal{N}(0_{n \times 1}, \hat{P}_k^{\text{xx}})$
 - 4 Compute the confidence box $\mathcal{I}_k^a$ given by (2.49)
-

D

Practical Aspects to Implement Zonotopic Filters

In this appendix, we discuss some challenges to implement set-based state estimators in continuous-time nonlinear dynamical systems with sampled measurement data. This topic is devoted to the ZMV (Algorithm B.13), but the ideas and methods discussed to overcome challenges are also applicable to other zonotopic filters. This content was published in [de Paula et al. \(2022c\)](#).

The ZMV algorithm is reviewed in Subsection B.2.2. Its forecast step is based on interval arithmetic to compute interval matrices. As explained in Subsection 2.2.2, the usage of such an arithmetic over nonlinear operations introduces conservative results. Here, we discuss some challenges to implement the ZMV when the process model is nonlinear and represented in continuous time, or the measurement model is nonlinear. In this case, the process model must be approximated and discretized. To approximate nonlinear models, we focus on the analytical linearization and mean value extension (Proposition 2.2.1) approaches. The first one disregards the linearization remainder, and thereby requires tuning the linearization error at the end. Such a strategy has been employed by the EKF ([Särkkä, 2013](#)) and ZMV ([Rego & Raffo, 2019](#)). The second approach directly considers the approximation error.

For nonlinear measurement models, step 9 of the ZMV algorithm (called the correction set) may yield the strip $\hat{\mathcal{Y}}_{k|k-1}$ with a large width due to the usage of interval arithmetic. This work revisits the procedure used in [Rego & Raffo \(2019\)](#) to approximate nonlinear

models for the ZMV and mitigate the enlargement of strips.

D.1 Continuous-Time Nonlinear Process Models

Consider the sampled-data system whose process and measurement models are given by

$$\dot{x}(t) = f(x(t), u(t)) + w(t), \quad (\text{D.1})$$

$$y(kT_s) = C(kT_s)x(kT_s) + v(kT_s), \quad (\text{D.2})$$

where $x(t) \in \mathbb{R}^n$ is the state vector, $u(t) \in \mathbb{R}^p$ is the control input, $w(t) \in \mathbb{R}^n$ is the process noise, $y(kT_s) \in \mathbb{R}^m$ is the sampled output vector, $C(kT_s) \in \mathbb{R}^{m \times n}$ is the output matrix, and $v(kT_s) \in \mathbb{R}^m$ is the measurement noise. For the sake of simplicity, the notation $x_k \triangleq x(kT_s)$ is used to relate the sampled data at continuous time t to the discrete time k , given the sampling time T_s . The other assumptions of Problem 1.2.2 are also applicable.

The goal is to approximate (D.1) in discrete time to use the ZMV. The discretization method must be such that the wrapping and dependency effects are mitigated. In this work, the integration methods employ a step size equal to T_s . For applications where a step size smaller than T_s is considered, the ZMV should perform forecasts while the measurements are not acquired, as in Rego & Raffo (2019).

There exist several approaches for discretizing a continuous-time model, such as zero-order holder (ZOH), Euler, and Runge-Kutta. Both Euler and Runge-Kutta methods directly allow for discretizing deterministic, uncertain, nonlinear, LTI, LTV, and LPV models. The quality of discretization depends on both step size and discretization technique. Here, we are interested in analyzing the simplest discretization methods with step size T_s : ZOH and Euler. Such simplicity is an appealing advantage for the ZMV, since sets must be used over operations rather than elements. Thus, the smallest number of variable repetitions is required. In the ZOH method, the input variables are assumed to be constant during T_s between samples. To compute the state and input matrices (F^{xd} and F^{ud}), we must solve matrix exponentiation and integration as follows:

$$\begin{aligned} F^{\text{xd}} &= \exp(F^{\text{xc}}T_s), \\ F^{\text{ud}} &= \left(\int_0^{T_s} \exp(F^{\text{xc}}\tau) d\tau \right) F^{\text{uc}}, \end{aligned}$$

with F^{xc} and F^{uc} being the state and input matrices of the continuous-time linearized model. If they are solved using second-order truncated Taylor series expansion, as in Chen (1999), then the discretization method reaches second-order precision. This approximation reduces the processing time over iterations. The Taylor series should be truncated according to each problem. In the Euler method, the discretization of the continuous-time

nonlinear model is given by

$$x_k = x_{k-1} + T_s f(x[(k-1)T_s], u[(k-1)T_s]) + T_s w[(k-1)T_s].$$

Though accurate for punctual evaluations, the Runge-Kutta method may yield a discretized function containing repeated variables, which cannot be simplified. From the set perspective, this is the crucial drawback of the Runge-Kutta method, because it introduces (or increases) the dependency effect after applying the mean value extension. Since large sets may be obtained due to the dependency effect, the ZMV may diverge.

D.2 Approximation and Discretization Procedures

Consider the forecast step of the ZMV, in which the forecast zonotope $\hat{\mathcal{X}}_{k|k-1}$ is computed using the discrete-time dynamical model (1.1). Thus, the mean value extension, given by Proposition 2.2.1, directly provides the desired zonotope only for discrete-time functions. If the mean value extension is applied to discrete-time models, then no error tuning is required, since the approximation error is already included into the interval Jacobian matrix. On the other hand, if the mean value extension is applied to discretized models, then the discretization error must be tuned.

Next, two approximated results are described, denoted as *linearization-discretization* (LD) and *discretization-approximation* (DA). The term *approximation* is used to indicate that the yielded result is not exact. Figure D.1 illustrates the procedures regarding LD and DA for better understanding. The former aims at providing the forecast zonotope $\hat{\mathcal{X}}_{k|k-1}$ with no influence of the wrapping and dependency effects. In this case, the continuous-time nonlinear model (D.1) is approximated and discretized using the ZOH method. This approximation yields a Jacobian matrix evaluated at punctual values; the approximation and discretization errors are tuned at the end. The second approach aims at providing the forecast zonotope $\hat{\mathcal{X}}_{k|k-1}$ with the smallest influence of the wrapping and dependency effects. In this case, the continuous-time nonlinear model (D.1) is discretized via the Euler approach and approximated using the mean value extension. This approximation yields an interval matrix that compensates the approximation error, such that the discretization error is tuned at the end.

The LD approach is now described. Assume that a control input $u(kT_s)$, with reference $u^{\text{tr}}(kT_s)$ (where *tr* means trajectory), is applied to the system (D.1), and thereby, the states $x(kT_s)$ vary around the reference trajectory $x^{\text{tr}}(kT_s)$. From the estimator perspective, the linearization procedure is commonly performed around the most current state estimate $\hat{x}[(k-1)T_s]$, assuming that this known value is close enough to the reference trajectory $x^{\text{tr}}[(k-1)T_s]$. Therefore, applying the analytical linearization around $\hat{x}[(k-1)T_s]$ and $u^{\text{tr}}[(k-$

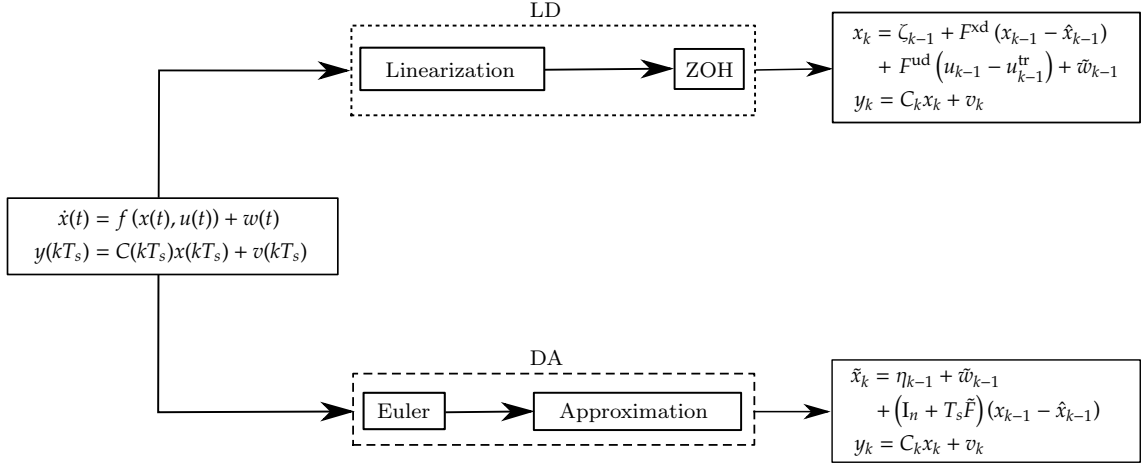


Figure D.1: Diagram to illustrate the linearization-discretization (LD), given by Algorithm D.1, and discretization-approximation (DA), given by Algorithm D.2. For both cases, the sampled-data system with additive noise (D.1)-(D.2) is assumed. On the one hand, the LD applies the analytical linearization around a reference trajectory to obtain the continuous-time linearized model (D.3). Next, this model is discretized using the zero-order-holder (ZOH) equivalence, yielding the discretized linearized model (D.6). On the other hand, the DA discretizes the initial model using the Euler method to obtain the discretized nonlinear model (D.13). After, this model is approximated using the mean value extension (Proposition 2.2.1), yielding the discretized approximated model (D.14).

1) $T_s]$ to (D.1) yields the following continuous-time linearized model:

$$\begin{aligned} \dot{x}(t) &= f(\hat{x}[(k-1)T_s], u^{\text{tr}}[(k-1)T_s]) + F^{\text{xc}}(x(t) - \hat{x}[(k-1)T_s]) \\ &\quad + F^{\text{uc}}(u(t) - u^{\text{tr}}[(k-1)T_s]) + w(t) + \tilde{w}(t), \end{aligned} \quad (\text{D.3})$$

where

$$F^{\text{xc}} \triangleq \nabla_x f(\hat{x}[(k-1)T_s], u^{\text{tr}}[(k-1)T_s]), \quad (\text{D.4})$$

$$F^{\text{uc}} \triangleq \nabla_u f(\hat{x}[(k-1)T_s], u^{\text{tr}}[(k-1)T_s]) \quad (\text{D.5})$$

are the Jacobian matrices, and $\tilde{w}(t)$ corresponds to the linearization error. Given the sampling time T_s , the linearized model (D.3) is discretized according to the second-order ZOH method, yielding the discretized linearized model

$$x_k = \zeta_{k-1} + F^{\text{xd}}(x_{k-1} - \hat{x}_{k-1}) + F^{\text{ud}}(u_{k-1} - u_{k-1}^{\text{tr}}) + \tilde{w}_{k-1}, \quad (\text{D.6})$$

where ζ_{k-1} is an offset term, which is obtained via numerical integration of the parcel $f(\hat{x}[(k-1)T_s], u^{\text{tr}}[(k-1)T_s])$. The term $\tilde{w}_{k-1} = F^{\text{wd}}w_{k-1} + \bar{w}_{k-1}$ is the augmented process noise that models both the discretized parcel $F^{\text{wd}}w_{k-1}$, for $w_{k-1} \triangleq w[(k-1)T_s]$, and the total approximation error $\bar{w}_{k-1}$. That is, $\bar{w}_{k-1}$ encloses the discretized term $F^{\text{wd}}\tilde{w}_{k-1}$ (where $F^{\text{wd}} =$

F^{wd}) and the truncation error of the following matrices:

$$F^{\text{xd}} \approx I_n + F^{\text{xc}}T_s + \frac{1}{2!}(F^{\text{xc}}T_s)^2, \quad (\text{D.7})$$

$$F^{\text{ud}} \approx \left(T_s I_n + \frac{T_s^2}{2!}F^{\text{xc}} + \frac{T_s^3}{3!}(F^{\text{xc}})^2 \right) F^{\text{uc}}, \quad (\text{D.8})$$

$$F^{\text{wd}} \approx T_s I_n + \frac{T_s^2}{2!}F^{\text{xc}} + \frac{T_s^3}{3!}(F^{\text{xc}})^2. \quad (\text{D.9})$$

Next, the zonotope $\bar{\mathcal{W}}_{k-1} = G_{k-1}^{\bar{\mathbf{w}}} \mathcal{B}_s^{\bar{\mathbf{w}}} \oplus c_{k-1}^{\bar{\mathbf{w}}}$ is defined as small as possible, such that $\bar{w}_{k-1} \in \bar{\mathcal{W}}_{k-1}$. Finally, each uncertain variable is replaced by its zonotopic set in (D.6), that is,

$$\hat{\mathcal{X}}_{k|k-1} = \zeta_{k-1} \oplus F^{\text{xd}}(\hat{\mathcal{X}}_{k-1} - \hat{x}_{k-1}) \oplus F^{\text{ud}}(u_{k-1} - u_{k-1}^{\text{tr}}) \oplus \tilde{\mathcal{W}}_{k-1}, \quad (\text{D.10})$$

where $\hat{\mathcal{X}}_{k-1} = \hat{G}_{k-1}^{\mathbf{x}} \mathcal{B}_s^{\mathbf{x}} \oplus \hat{x}_{k-1}$, $\tilde{\mathcal{W}}_{k-1} = F^{\text{wd}}\mathcal{W}_{k-1} \oplus \bar{\mathcal{W}}_{k-1}$, and $\mathcal{W}_{k-1} = G_{k-1}^{\mathbf{w}} \mathcal{B}_s^{\mathbf{w}} \oplus c_{k-1}^{\mathbf{w}}$. In doing so, the zonotope $\hat{\mathcal{X}}_{k|k-1} = \hat{G}_{k|k-1}^{\mathbf{x}} \mathcal{B}_s^{\mathbf{x}} \oplus \hat{x}_{k|k-1}$ is obtained, such that

$$\hat{G}_{k|k-1}^{\mathbf{x}} = \begin{bmatrix} F^{\text{xd}}\hat{G}_{k-1}^{\mathbf{x}} & F^{\text{wd}}G_{k-1}^{\mathbf{w}} & G_{k-1}^{\bar{\mathbf{w}}} \end{bmatrix}, \quad (\text{D.11})$$

$$\hat{x}_{k|k-1} = \zeta_{k-1} + F^{\text{ud}}(u_{k-1} - u_{k-1}^{\text{tr}}) + F^{\text{wd}}c_{k-1}^{\mathbf{w}} + c_{k-1}^{\bar{\mathbf{w}}}. \quad (\text{D.12})$$

This result is described in Algorithm D.1, whose complexity order is given by

$$O(\text{LD}) = O\left(n^2(n + p + n_s^{\mathbf{x}} + n_s^{\mathbf{w}})\right).$$

Algorithm D.1: $\hat{\mathcal{X}}_{k|k-1} = \text{LD}(\hat{\mathcal{X}}_{k-1}, \mathcal{W}_{k-1}, \bar{\mathcal{W}}_{k-1}, u_{k-1}, u_{k-1}^{\text{tr}}, T_s, f)$ (Adapted from [Rego & Raffo \(2019\)](#))

- 1 Given the zonotope $\hat{\mathcal{X}}_{k-1} = \hat{G}_{k-1}^{\mathbf{x}} \mathcal{B}_s^{\mathbf{x}} \oplus \hat{x}_{k-1}$ and the reference input u_{k-1}^{tr} , compute the Jacobian matrices F^{xd} in (D.7), F^{ud} in (D.8), and F^{wd} in (D.9)
 - 2 Compute the vector ζ_{k-1} related to (D.6)
 - 3 Given the zonotopes $\mathcal{W}_{k-1} = G_{k-1}^{\mathbf{w}} \mathcal{B}_s^{\mathbf{w}} \oplus c_{k-1}^{\mathbf{w}}$ and $\bar{\mathcal{W}}_{k-1} = G_{k-1}^{\bar{\mathbf{w}}} \mathcal{B}_s^{\bar{\mathbf{w}}} \oplus c_{k-1}^{\bar{\mathbf{w}}}$, and the control input $u_{k-1} \in \mathbb{R}^p$, obtain the forecast zonotope $\hat{\mathcal{X}}_{k|k-1} = \hat{G}_{k|k-1}^{\mathbf{x}} \mathcal{B}_s^{\mathbf{x}} \oplus \hat{x}_{k|k-1}$ with (D.11)-(D.12)
-

In (D.12), a numerical integration must be solved to find the vector ζ_{k-1} . Here, either the forward Euler method or the Runge-Kutta method is used. This choice is established according to the method used to discretize the continuous dynamics (D.1) during the simulation of the true states x_k .

The DA approach is now presented. Given the sampling time T_s , the continuous-time nonlinear model (D.1) is discretized via the Euler method to obtain

$$x_k = x_{k-1} + T_s f(x[(k-1)T_s], u[(k-1)T_s]) + \tilde{w}_{k-1}, \quad (\text{D.13})$$

where $\tilde{w}_{k-1} = T_s w_{k-1} + \bar{w}_{k-1}$ is the augmented process noise that models the noise w_{k-1} and

the discretization error $\bar{w}_{k-1}$. Next, the mean value theorem (Kühn, 1998) is applied to (D.13) to obtain the discretized approximated model

$$\tilde{x}_k = \eta_{k-1} + (I_n + T_s \tilde{F})(x_{k-1} - \hat{x}_{k-1}) + \tilde{w}_{k-1}, \quad (\text{D.14})$$

where

$$\eta_{k-1} = \hat{x}_{k-1} + T_s f(\hat{x}[(k-1)T_s], u[(k-1)T_s]) \quad (\text{D.15})$$

and $\tilde{F} = \nabla_x f(x[(k-1)T_s], u[(k-1)T_s])$. After, the zonotope $\bar{\mathcal{W}}_{k-1} = G_{k-1}^{\bar{w}} \mathcal{B}_{\bar{s}}^{n_{\bar{w}}} \oplus c_{k-1}^{\bar{w}}$ is tuned as small as possible, such that $\bar{w}_{k-1} \in \bar{\mathcal{W}}_{k-1}$. Next, each uncertain variable is replaced by its zonotopic set in (D.14), that is,

$$\hat{\mathcal{X}}_{k|k-1} = \eta_{k-1} \oplus (I_n \oplus T_s \tilde{F})(\hat{\mathcal{X}}_{k-1} - \hat{x}_{k-1}) \oplus \tilde{\mathcal{W}}_{k-1}, \quad (\text{D.16})$$

where $\hat{\mathcal{X}}_{k-1} = \hat{G}_{k-1}^x \mathcal{B}_{\hat{s}}^{n_{\hat{x}}} \oplus \hat{x}_{k-1}$, $\tilde{F} = \nabla_x f(\mathcal{X}[(k-1)T_s], u[(k-1)T_s])$, $\mathcal{W}_{k-1} = G_{k-1}^w \mathcal{B}_{\bar{s}}^{n_{\bar{w}}} \oplus c_{k-1}^w$, and $\tilde{\mathcal{W}}_{k-1} = T_s \mathcal{W}_{k-1} \oplus \bar{\mathcal{W}}_{k-1}$. Applying interval arithmetic to the parcel $I_n \oplus T_s \tilde{F}$ yields the interval matrix $[M]$, that is,

$$[M] = \square \{I_n \oplus T_s \tilde{F}\}. \quad (\text{D.17})$$

After, the zonotope inclusion is applied to the zonotope $[M]\hat{G}_{k-1}^x \mathcal{B}_{\hat{s}}^{n_{\hat{x}}} \oplus 0_{n \times 1}$ to obtain $\mathcal{X}^a = G^a \mathcal{B}^{(n+n_{\hat{x}})} \oplus 0_{n \times 1}$ from

$$\mathcal{X}^a = \diamond \{[M]\hat{G}_{k-1}^x \mathcal{B}_{\hat{s}}^{n_{\hat{x}}} \oplus 0_{n \times 1}\}. \quad (\text{D.18})$$

Finally, the forecast zonotope $\hat{\mathcal{X}}_{k|k-1} = \hat{G}_{k|k-1}^x \mathcal{B}_{\hat{s}}^{(n_{\hat{x}}^w + n_{\hat{s}}^{\bar{w}} + n + n_{\hat{s}}^x)} \oplus \hat{x}_{k|k-1}$ is such that

$$\hat{G}_{k|k-1}^x = \begin{bmatrix} T_s G_{k-1}^w & G_{k-1}^{\bar{w}} & G^a \end{bmatrix}, \quad (\text{D.19})$$

$$\hat{x}_{k|k-1} = \eta_{k-1} + T_s c_{k-1}^w + c_{k-1}^{\bar{w}}. \quad (\text{D.20})$$

This result is described in Algorithm D.2, whose complexity order is given by

$$O(\text{DA}) = O(n^2 n_{\hat{s}}^x + n n_{\hat{s}}^w).$$

Algorithm D.2: $\hat{\mathcal{X}}_{k|k-1} = \text{DA}(\hat{\mathcal{X}}_{k-1}, \mathcal{W}_{k-1}, \bar{\mathcal{W}}_{k-1}, u_{k-1}, T_s, f)$

- 1 Given the zonotope $\hat{\mathcal{X}}_{k-1} = \hat{G}_{k-1}^x \mathcal{B}_{\hat{s}}^{n_{\hat{x}}} \oplus \hat{x}_{k-1}$ and the control input $u_{k-1} \in \mathbb{R}^p$, use interval arithmetic to compute the interval matrix $[M]$ in (D.17)
 - 2 Compute the vector η_{k-1} in (D.15)
 - 3 Apply the zonotope inclusion (Proposition 2.2.2) to the zonotope $[M]\hat{G}_{k-1}^x \mathcal{B}_{\hat{s}}^{n_{\hat{x}}} \oplus 0_{n \times 1}$ to obtain the zonotope $\mathcal{X}^a = G^a \mathcal{B}^{(n+n_{\hat{x}})} \oplus 0_{n \times 1}$ in (D.18)
 - 4 Given the zonotopes $\mathcal{W}_{k-1} = G_{k-1}^w \mathcal{B}_{\bar{s}}^{n_{\bar{w}}} \oplus c_{k-1}^w$ and $\bar{\mathcal{W}}_{k-1} = G_{k-1}^{\bar{w}} \mathcal{B}_{\bar{s}}^{n_{\bar{w}}} \oplus c_{k-1}^{\bar{w}}$, obtain the forecast zonotope $\hat{\mathcal{X}}_{k|k-1} = \hat{G}_{k|k-1}^x \mathcal{B}_{\hat{s}}^{(n_{\hat{x}}^w + n_{\hat{s}}^{\bar{w}} + n + n_{\hat{s}}^x)} \oplus \hat{x}_{k|k-1}$ with (D.19)-(D.20)
-

Recall that, in both LD and DA, the zonotope $\bar{\mathcal{W}}_{k-1}$ must be previously tuned to

characterize the linearization and/or approximation errors over $k \in \mathbb{Z}_+$. To achieve that, offline simulations can be firstly executed with the known models (D.1) and (D.2). Next, the tuned set can be employed in online experiments, justifying the practical application of LD and DA.

D.2.1 Numerical Examples

Three numerical nonlinear examples are discussed next. The main goal is to apply the previous algorithms to different scenarios of increased complexity and, thereby, to assess performance. For each example, the ZMV is used with both segment and volume minimizations, and with the same reduction order $\varphi_g = 20$. To generate the true states x_k for the continuous-time processes, the fixed-step, fourth-order Runge-Kutta method is employed.

Example D.2.1. This example considers a discrete-time dynamical system whose process model is nonlinear. Only the approximation procedures used in the LD and DA (Algorithms D.1 and D.2 respectively) are investigated. That is, the corresponding discretization methods are initially disregarded. The goal is to illustrate how the wrapping and dependency effects can become the ZMV conservative, since the mean value extension is used in the DA. In this case, the measurements are not enough to mitigate the enlargement of corrected zonotopes over the time steps, and then the ZMV diverges. On the other hand, the analytical linearization used in the LD can circumvent such divergence.

Consider the discrete-time system given by (3.128)-(3.129), where $|w_{1,k-1}| \leq 0.1$ and $|w_{2,k-1}| \leq 0.1$ are the process noises, and $|v_k| \leq 0.2$ is the measurement noise. To simulate the system (3.128)-(3.129), the initial states are set as $x_0 = 1_{2 \times 1}$. The realizations w_{k-1} and v_k are taken from uniform distributions.

To estimate states, we set $\hat{X}_0 = 3I_2 \mathcal{B}^2 \oplus 0_{2 \times 1}$, $\mathcal{W}_{k-1} = 0.1I_2 \mathcal{B}^2 \oplus 0_{2 \times 1}$, and $\mathcal{V}_k = 0.2\mathcal{B} \oplus 0$. No process error tuning was applied to the LD. Therefore, $x_0 \in \hat{X}_0$, $w_{k-1} \in \mathcal{W}_{k-1}$, $v_k \in \mathcal{V}_k$.

The Jacobian matrix related to (3.128) is given by

$$F = \begin{bmatrix} 0.1 \exp(x_{1,k-1}) + 0.1x_{2,k-1} & -0.7 + 0.1x_{1,k-1} + 0.2x_{2,k-1} \\ 1 - 0.2x_{1,k-1} + 0.2x_{2,k-1} & 1 + 0.2x_{1,k-1} \end{bmatrix}.$$

With respect to the evaluation of this matrix, the analytical linearization replaces x_{k-1} by the center $\hat{x}_{k-1}$, whereas the mean value extension replaces x_{k-1} by the zonotope $\hat{X}_{k-1}$. Since there exist nonlinear operations, such a zonotope is replaced by its interval hull $\square \hat{X}_{k-1}$ to apply known interval operations. These operations guarantee the inclusion $F(X) \subseteq \square F(X)$, where $\square F$ is the natural interval extension of F given in Definition 2.2.15.

The state estimation is illustrated in Figure D.2. From Figures D.2a and D.2b, it is verified that the mean value extension leads the ZMV to diverge, because the wrapping and

dependency effects occur in the forecast step. The wrapping effect starts by overestimating $\hat{\mathcal{X}}_{k-1}$ by $\square\hat{\mathcal{X}}_{k-1}$. Since there exist operations between $x_{1,k-1}$ and $x_{2,k-1}$ at the same matrix position in F , some points into $\square\hat{\mathcal{X}}_{k-1}$, but outside $\hat{\mathcal{X}}_{k-1}$, are included. In turn, the dependency effect appears due to $\square\{F\}\hat{\mathcal{G}}_{k-1}^x$ in (D.18). Although there are no repeated variables at the same matrix position in F , $\square\{F\}\hat{\mathcal{G}}_{k-1}^x$ introduces summations among repeated variables. Such a summation appears because the generator matrix $\hat{\mathcal{G}}_{k-1}^x$ mixes the variables within F after multiplication. That is, if a given row of the interval Jacobian matrix has at least a variable appearing at two different positions, then the dependency effect may arise. Furthermore, since the zonotope inclusion must guarantee $\square(I_n \oplus F)\hat{\mathcal{G}}_{k-1}^x \mathcal{B}_s^x \subset \diamond \{\square(I_n \oplus F)\hat{\mathcal{G}}_{k-1}^x \mathcal{B}_s^x\}$ at the end, the interval generator matrix $\square(I_n \oplus F)\hat{\mathcal{G}}_{k-1}^x$ is replaced by an element generator matrix (2.22). This procedure adds wrapping effect.

For open-loop stable systems as in (3.128)-(3.129), it is desired that the eigenvalues of the Jacobian matrix F have their amplitude within the unit circle. This issue is relevant to limit the reachability of zonotopes over the state estimation, since the measurements are incorporated in an approximate way and may not be effective to refine the large forecasts originated from unstable processes. This is the explanation why the analytical linearization led the ZMV to converge, while the mean value extension led the ZMV to diverge.

Two different analysis can be obtained from Figures D.2a and D.2b. For the LD, the segment minimization returns the smallest values of Frobenius norm (1.6), while the volume minimization returns the smallest values of volume (1.7). In principle, these results are expected, because each approach is interested in minimizing the corresponding performance criterion. The opposite is observed for the DA. Note that the segment minimization leads the ZMV to diverge faster. In other words, the volume minimization is directly related to reduce uncertainty. This is the reason why the volume minimization returns the smallest intervals for the LD in Figures D.2c and D.2d.

The next example considers a continuous-time nonlinear process. Here, we start by analyzing the performance of the ZMV when all steps of the LD and DA are employed.

Example D.2.2. Consider the batch reactor system described by (5.63)-(5.64), whose measurement noise is characterized by $|v_k| \leq 0.3$. To simulate the system, the Runge-Kutta method is used to discretize the continuous dynamics with sampling time $T_s = 3$ s and initial state $x(0) = [3 \ 1]^T$. No process error is added to the states. Moreover, the realizations v_k are taken from a uniform distribution.

To estimate states, we set $\hat{\mathcal{X}}_0 = 1.2\mathbf{I}_2\mathcal{B}^2 \oplus [2.5 \ 1.5]^T$, and $\mathcal{V}_k = 0.3\mathcal{B} \oplus 0$. These values are based on de Paula et al. (2018). To improve the stability of the ZMV with Algorithm D.1, the process error $\bar{\mathcal{W}}_{k-1} = 1 \times 10^{-5}\mathbf{I}_2\mathcal{B}^2 \oplus 0_{2 \times 1}$ was added. No process error was added to the DA.

The Jacobian matrix related to (5.63) is given by $F = \begin{bmatrix} -4k_1x_1(t) & 2k_2 \\ 2k_1x_1(t) & -k_2 \end{bmatrix}$. Since there exists

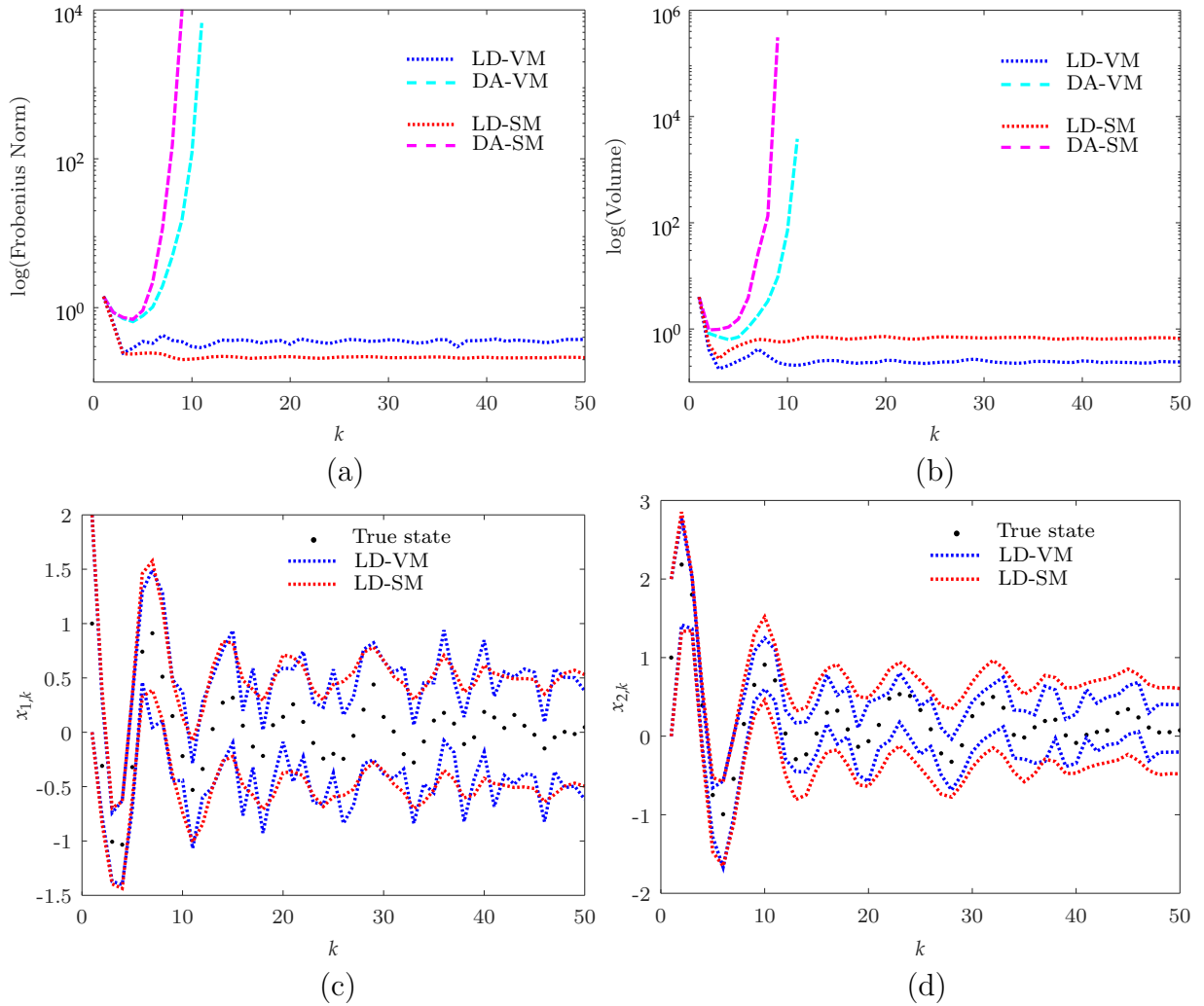


Figure D.2: State estimation of the system (3.128)-(3.129) using the ZMV. In (a) and (b), the convergence of the ZMV with either LD given by Algorithm D.1 ($\cdots$) or DA given by Algorithm D.2 ($--$) is evaluated in terms of performance criteria. Each algorithm is individually performed with the segment (red or magenta line) and volume (blue or cyan line) minimizations. In (c) and (d), the true states (black points) are illustrated in the intervals yielded by the ZMV with the LD.

only x_1 in F (and it appears only once at the same row), the mean value extension does not cause the wrapping and dependency effects. However, due to the zonotope inclusion, the wrapping effect is introduced in the DA, because $\square\{F\}\hat{G}_{k-1}^x$ is an interval matrix.

In Figure D.3, the results yielded by the ZMV are presented. From Figure D.3(a), the DA (dashed line) can lead the ZMV to two different scenarios. In the first, the segment minimization yields divergent estimates. In the second, the volume minimization yields convergent estimates. The explanation is related to the goal of each performance criterion. In the segment minimization, we aim at minimizing the variation of uncertainty around the center estimate, since the segments (or generators) are evaluated in a quadratic function (square Frobenius norm). This is the reason why the smoothest results are achieved by the segment minimization in Figures D.2a and D.2b, for instance. However, such a goal

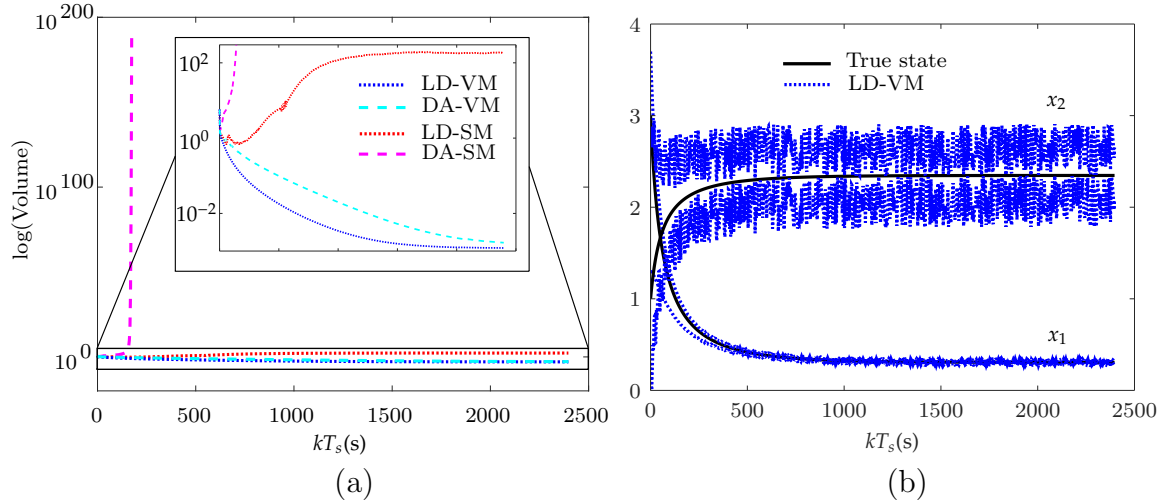


Figure D.3: State estimation of the two-state batch reactor. Graph (a) illustrates the time evolution of volume when the ZMV is performed with either LD given by Algorithm D.1 ($\cdots$) or DA given by Algorithm D.2 ($--$). Each algorithm is individually executed with the segment (red or magenta line) and volume (blue or cyan line) minimizations. In (b), the intervals estimated by the ZMV with the LD via volume minimization are presented. For any time instant, the true states ($--$) are included in the intervals.

can disregard smaller sets when their variation is larger. In turn, the volume minimization returns the smallest set, although its variation can be the largest. This is the reason why expressive variations are achieved by the volume minimization in Figures D.2a and D.2b, for instance.

In this example, the volume minimization yields similar solutions for both LD and DA, approximately $kT_s = 2500$ s. When the wrapping and/or dependency effects are not significant for the estimator, DA is the most indicated to be used with the ZMV, since no error tuning is necessary.

Example D.2.3. Consider the Lorenz system (Wolf et al., 1985)

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_3(t) \end{bmatrix} = \begin{bmatrix} \alpha(x_2(t) - x_1(t)) \\ x_1(t)(\beta - x_3(t)) - x_2(t) \\ x_1(t)x_2(t) - \gamma x_3(t) \end{bmatrix}, \quad (\text{D.21})$$

whose parameters $\alpha = 16$, $\beta = 45.92$, and $\gamma = 4$ characterize chaotic dynamics. The measurement model is given by

$$y_k = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} x_k + v_k, \quad (\text{D.22})$$

where v_k corresponds to the measurement noise. To simulate the system, the Runge-Kutta integration is employed with sampling time $T_s = 0.01$ s and initial state $x(0) = [0.1 \ 0.1 \ 0.1]^T$. No process error is added to the simulation. The realizations v_k are taken from a uniform distribution defined in the interval $[-3\sigma_v, 3\sigma_v]$, where σ_v is the standard

deviation, such that the signal-to-noise ratio is 20 dB.

To estimate states, we set $\hat{\mathcal{X}}_0 = 0.75\mathbf{I}_3\mathcal{B}^3 \oplus \mathbf{0}_{3 \times 1}$, and $\mathcal{V}_k = 3\sigma_v\mathcal{B} \oplus \mathbf{0}$. The process error $\bar{\mathcal{W}}_{k-1} = 0.03\mathbf{I}_3\mathcal{B}^3 \oplus \mathbf{0}_{3 \times 1}$ was added only to the LD.

The Jacobian matrix of the model (D.21) is given by $F = \begin{bmatrix} -\alpha & \alpha & 0 \\ \beta - x_3(t) & -1 & -x_1(t) \\ x_2(t) & x_1(t) & -\gamma \end{bmatrix}$. By

applying the mean value extension in the DA, there is no dependency effect in the interval Jacobian matrix $\square F$. However, the wrapping effect appears over the states x_2 and x_3 . This occurrence is discussed in the first example. Moreover, the wrapping effect is added to the forecast zonotope $\hat{\mathcal{X}}_{k|k-1}$ after applying the zonotope inclusion, since there exist intervals into the Jacobian matrix $\square F$.

In Figure D.4, convergent estimates are achieved for all algorithms. Recall that the segment minimization is computationally the fastest approach, while the volume minimization is the slowest one. Unlike Example D.2.2, the mean value extension introduces wrapping effect in the forecast zonotope. The performance difference between the LD and DA then becomes significant. Therefore, in terms of precision, the fastest results of the ZMV (yielded by the LD) are similar to the slowest results of the ZMV (yielded by the DA). Thus, when the wrapping and/or dependency effects compromise the convergence or the precision of the estimator, the LD should be used with the ZMV.

D.3 Nonlinear Measurement Models

Consider the sampled-data system whose process model is given by (D.1), and whose nonlinear measurement model is represented by

$$y_k = h(x_k) + v_k, \quad (\text{D.23})$$

where $x_k \in \mathbb{R}^n$ is the state vector, and $v_k \in \mathbb{R}^m$ is the measurement noise. The noise term v_k is characterized by the zonotope $\mathcal{V}_k = G_k^v \mathcal{B}_s^m \oplus c_k^v$.

Addressing nonlinearity is a difficult task for ZMV, since interval arithmetic may introduce the dependency effect in the correction set $\hat{\mathcal{Y}}_{k|k-1}$. To circumvent this inconvenience, the linearization procedure used in Rego & Raffo (2019) is revisited. Thus, the goals are: (i) to understand why problems may occur in ZMV when the measurement model (D.23) is nonlinear; and (ii) to provide a strategy to mitigate such problems.

First, retake the linear measurement model (D.2), for which step 5 of Algorithm B.13 that is given by

$$[\Lambda] = \square \left\{ c_i^\top \hat{\mathcal{X}}_{k|k-1} \oplus \left(-h_i \left(\hat{\mathcal{X}}_{k|k-1}, \mathcal{V}_k \right) \right) \right\} \quad (\text{D.24})$$

is reduced to $[\Lambda] = \square \{-\mathcal{V}_{i,k}\}$, for $i = 1, \dots, m$. Therefore, interval arithmetic introduces no overestimation to compute the i th strip $\hat{\mathcal{Y}}_{i,k|k-1}$.

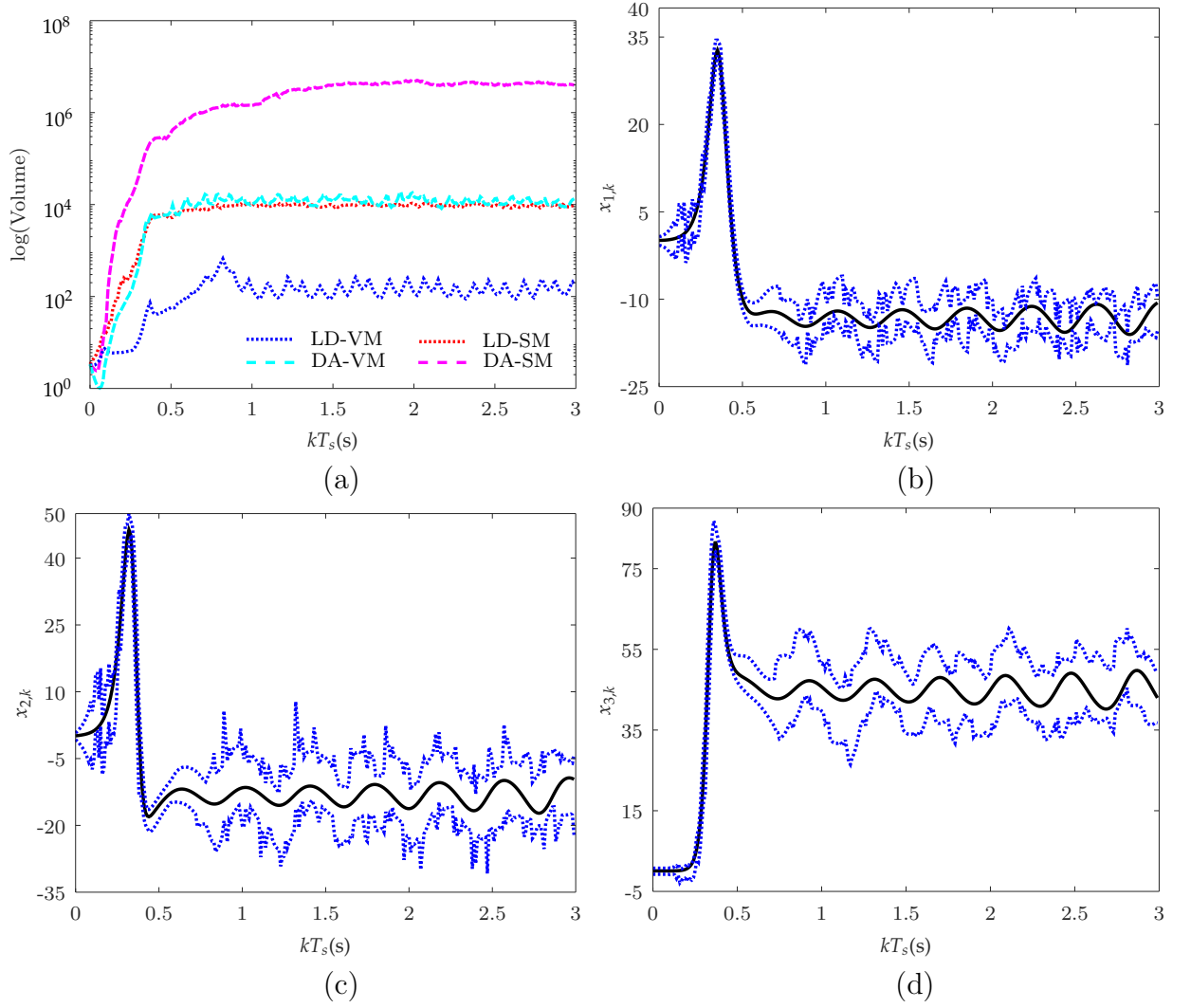


Figure D.4: Graph (a) presents the time evolution of the ZMV with either LD given by Algorithm D.1 ($\cdots$) or DA given by Algorithm D.2 ($--$). The ZMV is individually executed with the segment (red or magenta line) and volume (blue or cyan line) minimizations. In (b)-(d), the true states ($--$) are involved by intervals estimated by the ZMV with the LD via volume minimization.

Nonetheless, if the measurement model is nonlinear, then the dependency and wrapping effects may affect the computation of the correction set $\hat{\mathcal{Y}}_{k|k-1}$. This occurs due to interval arithmetic applied to (D.24). The dependency effect may arise because $\hat{\mathcal{X}}_{k|k-1}$ appears at least twice. The zonotope $\mathcal{V}_k$ can also introduce the dependency effect if it has repeated variables, such as in $\mathbf{y}_k = \mathbf{x}_{1,k}\mathbf{v}_k + \mathbf{x}_{2,k}\sin(\mathbf{v}_k)$, where $\mathbf{v}_k$ appears twice. The wrapping effect may arise when the natural interval extension is applied to (D.24), while the m dimensions of the transformation (D.23) are individually analyzed by (D.24). In this case, the multidimensional mapping of $\hat{\mathcal{X}}_{k|k-1} \subset \mathbb{R}^n$ and $\mathcal{V}_k \subset \mathbb{R}^r$ may be the cause, since their interval hulls $\square\hat{\mathcal{X}}_{k|k-1}$ and $\square\mathcal{V}_k$ introduce extra points to be operated. See Subsection 2.2.2 for more details.

Consequently, (D.23) is approximated to an analytically linearized model to mitigate

the wrapping and dependency effects in the ZMV during step 9, called the correction set, where the i th strip $\hat{\mathcal{Y}}_{i,k|k-1}$ is computed. Next, the obtained strip whose result is denoted as *function linearization* (FL) is demonstrated. Given the forecast zonotope $\hat{\mathcal{X}}_{k|k-1} = \hat{G}_{k|k-1}^x \mathcal{B}_s^{n_s^x} \oplus \hat{x}_{k|k-1}$ and the discrete-time nonlinear model (D.23), the Jacobian matrix is computed as

$$H = \nabla_x h(\hat{x}_{k|k-1}) \quad (\text{D.25})$$

to obtain the discrete-time linearized model

$$y_k = \bar{h}(x_k, \bar{v}_k) = h(\hat{x}_{k|k-1}) + H(x_k - \hat{x}_{k|k-1}) + \bar{v}_k, \quad (\text{D.26})$$

where the augmented measurement noise $\bar{v}_k = v_k + \bar{v}_k$ models the noise v_k and the approximation error $\bar{v}_k$. Next, the zonotope $\bar{\mathcal{V}}_k = G_k^v \mathcal{B}_s^{n_s^v} \oplus c_k^v$ is tuned as small as possible, such that $\bar{v}_k \in \bar{\mathcal{V}}_k$. After replacing each uncertain variable by its zonotopic set in (D.26), the vector $c_i \in \mathbb{R}^n$ is computed as

$$\begin{aligned} c_i &= \text{mid} \left(\square \left\{ \nabla_x \bar{h}_i \left(\hat{\mathcal{X}}_{k|k-1}, \bar{\mathcal{V}}_k \right)^\top \right\} \right) \\ &= H_{i,:}^\top. \end{aligned} \quad (\text{D.27})$$

Thereby, the approximated model (D.26) and the vector c_i (D.27) are replaced in (D.24) to obtain the interval

$$[\Lambda] = \square \left\{ (H_{i,:} \hat{x}_{k|k-1} - h_i(\hat{x}_{k|k-1})) \oplus (-\mathcal{V}_{i,k}) \oplus (-\bar{\mathcal{V}}_{i,k}) \right\}. \quad (\text{D.28})$$

The scalars $s_i, \rho_i \in \mathbb{R}$ are computed as

$$\begin{aligned} s_i &= \text{mid}([\Lambda]) \\ &= H_{i,:} \hat{x}_{k|k-1} - h_i(\hat{x}_{k|k-1}) - c_{i,k}^v - c_{i,k}^{\bar{v}}, \end{aligned} \quad (\text{D.29})$$

$$\begin{aligned} \rho_i &= \text{rad}([\Lambda]) \\ &= \text{rs} \left(\left[(-G_k^v)_{i,:} \quad (-G_k^{\bar{v}})_{i,:} \right] \right), \end{aligned} \quad (\text{D.30})$$

where rs is given by (2.26). Finally, given the i th measurement $y_{i,k}$, the i th strip $\hat{\mathcal{Y}}_{i,k|k-1}$ is obtained. This result is described in Algorithm D.3, whose complexity order is given by

$$O(\text{FL}) = O(n + n_s^v + n_s^{\bar{v}}).$$

Recall that FL requires the previous knowledge of the zonotope $\bar{\mathcal{V}}_k$ over $k \in \mathbb{Z}_+$. As discussed in Subsection D.2, the tuning of $\bar{\mathcal{V}}_k$ can be offline executed with the known models (D.1) and (D.23). After, the tuned set can be applied to the real experiment.

Algorithm D.3: $\hat{\mathcal{Y}}_{i,k|k-1} = \text{FL}(\hat{\mathcal{X}}_{k|k-1}, \mathcal{V}_k, \bar{\mathcal{V}}_k, y_{i,k}, h_i)$ (Adapted from [Rego & Raffo \(2019\)](#))

- 1 Given the nonlinear function h_i , determine $H_{i,:}$ using (D.25)
 - 2 Given the forecast zonotope $\hat{\mathcal{X}}_{k|k-1} = \hat{\mathcal{G}}_{k|k-1}^x \mathcal{B}_s^x \oplus \hat{x}_{k|k-1}$, compute the vector c_i in (D.27)
 - 3 Given the zonotopes $\mathcal{V}_k = G_k^v \mathcal{B}_s^v \oplus c_k^v$ and $\bar{\mathcal{V}}_k = G_k^v \mathcal{B}_s^{\bar{v}} \oplus c_k^{\bar{v}}$, compute the interval $[\Lambda]$ in (D.28)
 - 4 Compute the scalars s_i in (D.29) and ρ_i in (D.30)
 - 5 $d_i = y_{i,k} + s_i$
 - 6 $\hat{\mathcal{Y}}_{i,k|k-1} = \{x \in \mathbb{R}^n : |c_i^\top x - d_i| \leq \rho_i\}$
-

D.3.1 Numerical Example

This example illustrates how the nonlinear measurement model is challenging for ZMV, since the dependency effect may arise during the correction set step. Therefore, the LD and DA are individually combined with the FL, given by Algorithm D.3, to perform the forecast and correction set steps of the ZMV. To verify the individual influence of the nonlinear measurement model in the ZMV, the ZMV was executed with a third combination: LD to compute forecast zonotopes and (D.24) to compute strips. In this case, the guaranteed procedure is used, rather than the FL.

Consider the continuous-time nonlinear system ([Rengaswamy et al., 2013](#))

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} r_1 x_1^2(t) x_2^5(t) - r_2 x_1^3(t) x_2^4(t) \\ -r_1 x_1^2(t) x_2^5(t) + r_2 x_1^3(t) x_2^4(t) \end{bmatrix}, \quad (\text{D.31})$$

where $r_1 = 2.5$ and $r_2 = 0.01$. The measurement model is given by

$$y_k = x_{1,k}^4 x_{2,k} + v_k, \quad (\text{D.32})$$

where $|v_k| \leq 1$ is the measurement noise. To simulate the system, the continuous dynamics are discretized via the Runge-Kutta method with sampling time $T_s = 0.1$ ms and initial state $x(0) = [3 \ 1]^\top$. No process error is added to the simulation. The realizations v_k are taken from a uniform distribution.

To estimate states, we set $\hat{\mathcal{X}}_0 = \mathbf{I}_2 \mathcal{B}^2 \oplus [2 \ 2]^\top$, $\bar{\mathcal{W}}_{k-1} = 0.1 \mathbf{I}_2 \mathcal{B}^2 \oplus 0_{2 \times 1}$, $\mathcal{V}_k = \mathbf{1} \mathcal{B} \oplus 0$, $\bar{\mathcal{V}}_k = \mathbf{1} \mathcal{B} \oplus 0$, and $\varphi_s = 20$. The error tuning was performed only for the LD. The ZMV is individually executed with the segment and volume minimizations. The Jacobian matrices related to (D.31) and (D.32) are, respectively, given by

$$\begin{aligned} F &= \begin{bmatrix} 2r_1 x_1(t) x_2^5(t) - 3r_2 x_1^2(t) x_2^4(t) & 5r_1 x_1^2(t) x_2^4(t) - 4r_2 x_1^3(t) x_2^3(t) \\ -2r_1 x_1(t) x_2^5(t) + 3r_2 x_1^2(t) x_2^4(t) & -5r_1 x_1^2(t) x_2^4(t) + 4r_2 x_1^3(t) x_2^3(t) \end{bmatrix}, \\ H &= \begin{bmatrix} 4x_{1,k}^3 x_{2,k} & x_{1,k}^4 \end{bmatrix}. \end{aligned}$$

Analogous to Example D.2.1, after applying the mean value extension in the DA, both wrapping and dependency effects appear within the interval Jacobian matrix $\square F$. Moreover, the zonotope inclusion adds wrapping to yield the forecast zonotope. This overestimation may be enough to lead the ZMV to diverge.

The state estimation is illustrated in Figure D.5. Although the FL is employed together with the DA, the ZMV do not reach convergence. Likewise, divergence is verified when the guaranteed approach (D.24) is combined with the LD using the segment minimization. In these cases, the forecast zonotopes and the correction sets are significantly large, affecting the convergence of the corrected zonotopes over the state estimation. On the other hand, the prior combination reaches convergence when the segment minimization is replaced by the volume minimization. When the FL is employed together with the LD, the ZMV reaches convergence for both segment and volume minimizations.

Note that the volume minimization (blue dotted line) converges faster than the segment minimization (cyan dotted line) does for both performance criteria illustrated in (a) and (b). Therefore, this example highlights the convergence rate as a relevant criterion to be investigated during applications of the ZMV.

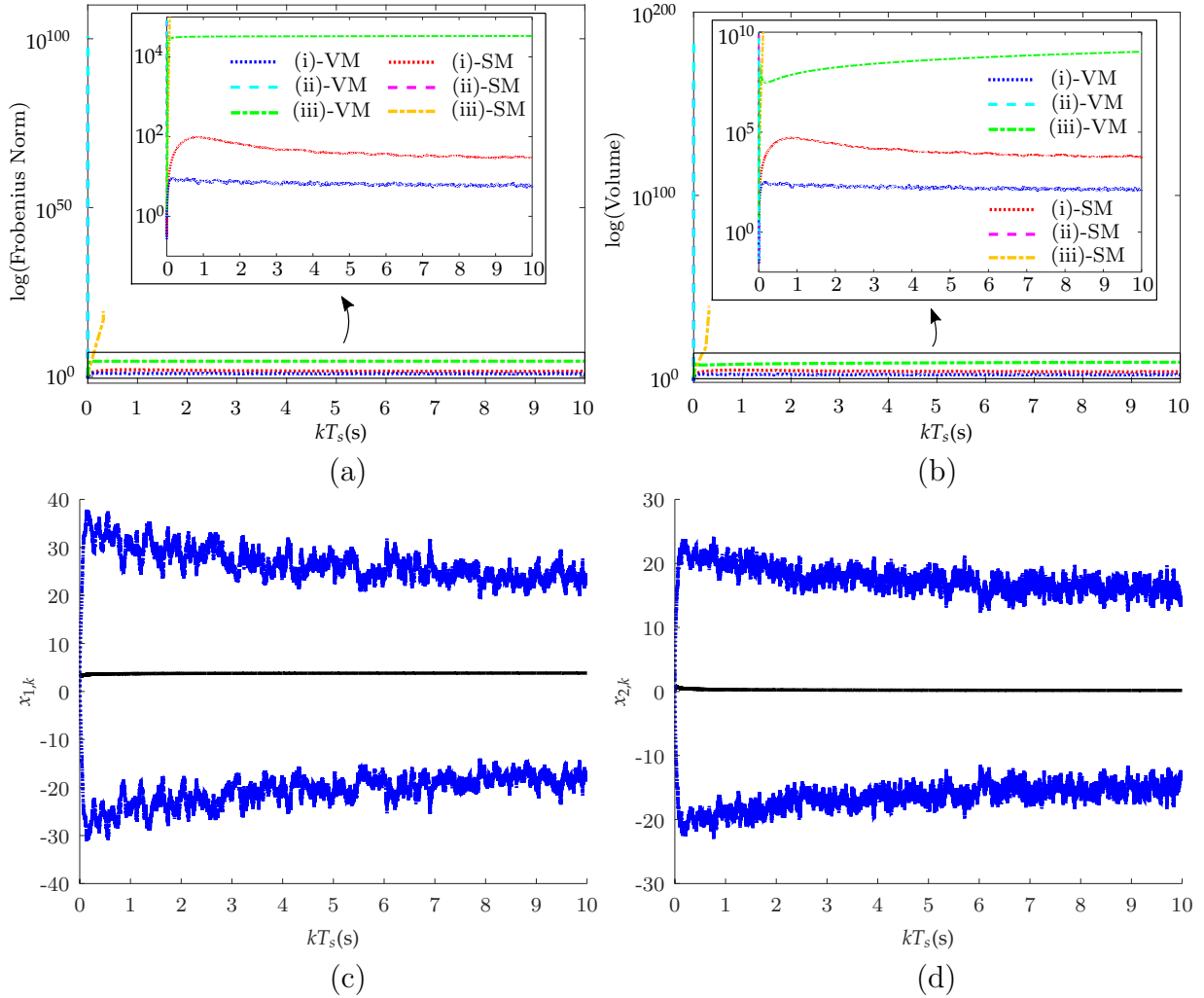


Figure D.5: State estimation for the system (D.31)-(D.32). In (a) and (b), the convergence of the ZMV is evaluated in terms of performance criteria. Such an algorithm is executed under three different combinations: (i) LD given by Algorithm D.1 with FL given by Algorithm D.3 ($\cdots$), (ii) DA given by Algorithm D.2 with FL ($--$), and (iii) LD with the guaranteed approach (D.24) ($-\cdot-$). Both LD and DA are used to compute forecast zonotopes, while both FL and the guaranteed approach (D.24) are used to compute strips. Each combination of the ZMV is individually performed with the segment (red, magenta, or yellow line) and volume (blue, cyan, or green line) minimizations. In (c) and (d), the true states ($--$) are illustrated in the intervals yielded by the ZMV with the combination (i).

E

Worst-Case Complexity

In this appendix, we show how each complexity order informed throughout the thesis was derived.

E.1 Basic Operations

General

Assumptions

Evaluation of nonlinear functions (such as real-valued functions and interval extensions) and matrices (such as Jacobian and Hessian matrices) has complexity $O(1)$. Moreover, matrix operations such as transpose, stacking, block diagonalization, vectorization, and reshape have complexity $O(1)$.

Summation of Matrices

Let $M \in \mathbb{R}^{a \times b}$ and $N \in \mathbb{R}^{a \times b}$. Then, $O(M + N) = O(ab)$.

Product of Matrices

Let $M \in \mathbb{R}^{a \times b}$ and $N \in \mathbb{R}^{b \times c}$. Then, $O(MN) = O(abc)$.

Cholesky Factorization

Let $P = P^\top > 0_{n \times n}$ be a symmetric and positive-definite matrix. The computational cost to execute the Cholesky factorization $P = SS^\top$ is $O(n^3)$ (Boyd et al., 2004), with $S \in \mathbb{R}^{n \times n}$ being a lower triangular matrix.

Matrix Inversion

The computational cost to invert the non-singular matrix $P \in \mathbb{R}^{n \times n}$ is $O(n^3)$ (Tang et al., 2016).

Jordan Decomposition

The computational cost to decompose a matrix $P \in \mathbb{R}^{n \times n}$ in the Jordan canonical form is $O(n^3)$ (Mastronardi & Dooren, 2017). This decomposition can be used to calculate the determinant of square matrix, and thereby, the same complexity is attributed.

Matrix Trace

Let $P \in \mathbb{R}^{n \times n}$ be a square matrix. Then,

$$O(\text{tr}(P)) = O(P_{1,1} + P_{2,2} + \cdots + P_{n,n}) = O(n).$$

Singular Value Decomposition

The computational cost to decompose a matrix $M \in \mathbb{R}^{a \times b}$ in terms of its singular values is given by $O(ab^2)$ (Golub & Van Loan, 2013), with $a \geq b$. This factorization can be used to find an orthonormal basis for the null space of M , and thereby, the same complexity is attributed.

F-radius

Consider a matrix $M \in \mathbb{R}^{a \times b}$, with $a \leq b$, whose F -radius is computed as $\|M\|_F^2 = \text{tr}(MM^\top)$. Then,

$$O(\text{tr}(MM^\top)) = O(a^2b) + O(a) = O(a^2b).$$

Solution of Linear Program

The computational cost to solve a specific LP containing N_d decision variables and N_c constraints (including equality and inequality) is $O(N_d N_c^3)$ (Kelner & Spielman, 2006).

Gaussian Random Vectors

Affine transformation (Property 2.1.1)

Let $X \sim \mathcal{N}(\hat{x}, P^{\text{xx}})$, $u \in \mathbb{R}^m$, and $L \in \mathbb{R}^{m \times n}$, where $\hat{x} \in \mathbb{R}^n$ and $P^{\text{xx}} \in \mathbb{R}^{n \times n}$. Consider the affine transformation $Y = u + LX \sim \mathcal{N}(u + L\hat{x}, LP^{\text{xx}}L^\top)$. Then,

$$O(Y) = O(u + L\hat{x}) + O(LP^{\text{xx}}L^\top) = O(mn + m) + O(n^2m + nm^2) = O(n^2m + nm^2).$$

Summation (Property 2.1.2)

Let $X \sim \mathcal{N}(\hat{x}, P^{\text{xx}})$ and $W \sim \mathcal{N}(\hat{w}, P^{\text{ww}})$, where $\hat{x} \in \mathbb{R}^n$, $P^{\text{xx}} \in \mathbb{R}^{n \times n}$, $\hat{w} \in \mathbb{R}^n$, and $P^{\text{ww}} \in \mathbb{R}^{n \times n}$. Consider the summation $X + W \sim \mathcal{N}(\hat{x} + \hat{w}, P^{\text{xx}} + P^{\text{ww}})$. Then,

$$O(X + W) = O(\hat{x} + \hat{w}) + O(P^{\text{xx}} + P^{\text{ww}}) = O(n) + O(n^2) = O(n^2).$$

Cartesian Product (Property 2.1.3)

Let $X \sim \mathcal{N}(\hat{x}, P^{\text{xx}})$ and $W \sim \mathcal{N}(\hat{w}, P^{\text{ww}})$, where $\hat{x} \in \mathbb{R}^n$, $P^{\text{xx}} \in \mathbb{R}^{n \times n}$, $\hat{w} \in \mathbb{R}^w$, and $P^{\text{ww}} \in \mathbb{R}^{w \times w}$. Consider the Cartesian product

$$X \times W \sim \mathcal{N}\left(\begin{bmatrix} \hat{x} \\ \hat{w} \end{bmatrix}, \begin{bmatrix} P^{\text{xx}} & 0_{n \times w} \\ 0_{w \times n} & P^{\text{ww}} \end{bmatrix}\right).$$

Then,

$$O(X \times W) = O(1).$$

Zonotopes

Affine transformation (Property 2.2.2)

Let $\mathcal{X} = \{G^x, c^x\} \subset \mathbb{R}^n$, $u \in \mathbb{R}^m$, and $L \in \mathbb{R}^{m \times n}$, where $G^x \in \mathbb{R}^{n \times n_g}$ and $c^x \in \mathbb{R}^n$. Consider the affine transformation $L\mathcal{X} \oplus u = \{LG^x, Lc^x + u\}$. Then,

$$O(L\mathcal{X} \oplus u) = O(Lc^x + u) + O(LG^x) = O(mn + m) + O(mnn_g) = O(mnn_g).$$

Minkowski Sum (Property 2.2.3)

Let $\mathcal{X} = \{G^x, c^x\} \subset \mathbb{R}^n$ and $\mathcal{W} = \{G^w, c^w\} \subset \mathbb{R}^n$, where $G^x \in \mathbb{R}^{n \times n_g}$, $c^x \in \mathbb{R}^n$, $G^w \in \mathbb{R}^{n \times n_g^w}$, and $c^w \in \mathbb{R}^n$. Consider the Minkowski sum

$$\mathcal{X} \oplus \mathcal{W} = \left\{ \begin{bmatrix} G^x & G^w \end{bmatrix}, c^x + c^w \right\}.$$

Then,

$$O(\mathcal{X} \oplus \mathcal{W}) = O(c^x + c^w) = O(n).$$

Cartesian Product (Property 2.2.4)

Let $\mathcal{X} = \{G^x, c^x\} \subset \mathbb{R}^n$ and $\mathcal{W} = \{G^w, c^w\} \subset \mathbb{R}^w$, where $G^x \in \mathbb{R}^{n \times n_g}$, $c^x \in \mathbb{R}^n$, $G^w \in \mathbb{R}^{n^w \times n_g^w}$, and $c^w \in \mathbb{R}^{n^w}$. Consider the Cartesian product

$$\mathcal{X} \times \mathcal{W} = \left\{ \begin{bmatrix} G^x & 0_{n \times n_g^w} \\ 0_{n^w \times n_g} & G^w \end{bmatrix}, \begin{bmatrix} c^x \\ c^w \end{bmatrix} \right\}.$$

Then,

$$O(\mathcal{X} \times \mathcal{W}) = O(1).$$

Zonotope Reduction (ZR, Algorithm B.1)

Let $\mathcal{X} = \{G^x, c^x\} \subset \mathbb{R}^n$, with $G^x \in \mathbb{R}^{n \times n_g}$ and $c^x \in \mathbb{R}^n$. Let also $\varphi_g \geq n$ be the maximum number of generators to be fixed from $\mathcal{X}$, yielding the reduced-order zonotope $\mathcal{X}^\downarrow \supset \mathcal{X}$. By employing the generator-reduction algorithm proposed by [Combastel \(2003\)](#), the computational complexity to obtain $\mathcal{X}^\downarrow$ is given by ([Yang & Scott, 2018](#))

$$O(\mathcal{X}^\downarrow) = n_g (n + \log(n_g)).$$

Interval Hull

Let $\mathcal{X} = \{G^x, c^x\} \subset \mathbb{R}^n$, with $G^x \in \mathbb{R}^{n \times n_g}$ and $c^x \in \mathbb{R}^n$. Let $\square\mathcal{X} = [c^x - r, c^x + r]$ be the interval hull of $\mathcal{X}$, where $r = |G^x| 1_{n_g \times 1}$. Then,

$$\begin{aligned} O(\square\mathcal{X}) &= O(c^x - r) + O(c^x + r) + O(|G^x| 1_{n_g \times 1}) \\ &= O(n) + O(n) + O(nn_g) \\ &= O(nn_g). \end{aligned}$$

Zonotope Inclusion (Proposition 2.2.2)

Let $[\mathcal{X}] = [G^x] \mathcal{B}^{n_g} \oplus c^x$, where $[G^x] \subset \mathbb{R}^{n \times n_g}$ and $c^x \in \mathbb{R}^n$. Let $\diamond\mathcal{X} \supseteq [\mathcal{X}]$ be the zonotope inclusion given by

$$\diamond\mathcal{X} = \left[\text{mid}([G^x]) \quad L \right] \begin{bmatrix} \mathcal{B}^{n_g} \\ \mathcal{B}^n \end{bmatrix} \oplus c^x,$$

where $L \in \mathbb{R}^{n \times n}$ is a diagonal matrix, whose elements are computed as

$$L_{i,i} = \sum_{j=1}^{n_g} \text{rad}_{i,j}([G^x]), \quad i = 1, \dots, n. \quad (\text{E.1})$$

Then,

$$O(\diamond\mathcal{X}) = O(\text{mid}([G^x])) + O(L)$$

$$\begin{aligned}
&= O(nn_g) + O(nn_g) \\
&= O(nn_g).
\end{aligned}$$

Volume of Zonotope

Let $\mathcal{X} = \{G^x, c^x\}$, where $G^x \in \mathbb{R}^{n \times n_g}$ and $c^x \in \mathbb{R}^n$. Consider the volume of zonotope given by

$$\text{Vol}(\mathcal{X}) = 2^n \sum_{i=1}^{N(n_g, n)} |\det(T_i)|,$$

where $N(n_g, n)$ is the mathematical combination that returns all possible ways to choose n elements from a set with size $n_g \geq n$, and $T_i \in \mathbb{R}^{n \times n}$ denotes all matrices that can be obtained by taking n columns from the matrix G^x . Then,

$$\begin{aligned}
O(\text{Vol}(\mathcal{X})) &= N(n_g, n) (O(\det(\cdot)) + O(|\cdot|)) + O(\text{product with } 2^n) \\
&= N(n_g, n) O((n^3) + O(1)) + O(1) \\
&= O(n^3 N(n_g, n)).
\end{aligned}$$

Intersection between Zonotope and Strip (Property 2.2.5)

Let $\mathcal{X} = \{G^x, \bar{x}\}$, $\mathcal{S} = \{x \in \mathbb{R}^n : |c^\top x - d| \leq \rho\}$, and $\kappa \in \mathbb{R}^n$, where $G^x \in \mathbb{R}^{n \times n_g}$, $\bar{x} \in \mathbb{R}^n$, $c \in \mathbb{R}^n$, $d \in \mathbb{R}$, and $\rho \geq 0$. Assume that κ is known and let $\check{\mathcal{X}} = \{G^{\check{x}}, \check{x}\} \supseteq \mathcal{X} \cap \mathcal{S}$, where

$$\begin{aligned}
G^{\check{x}} &= \begin{bmatrix} (I_n - \kappa c^\top) G^x & \rho \kappa \end{bmatrix}, \\
\check{x} &= \bar{x} + \kappa (d - c^\top \bar{x}).
\end{aligned}$$

Then,

$$\begin{aligned}
O(\check{\mathcal{X}}) &= O(G^{\check{x}}) + O(\check{x}) \\
&= O(n^2 n_g) + O(n) \\
&= O(n^2 n_g).
\end{aligned}$$

Constrained Zonotopes

Affine transformation (Property 2.2.6)

Let $\mathcal{X} = \{G^x, c^x, A^x, b^x\} \subset \mathbb{R}^n$, $u \in \mathbb{R}^m$, and $L \in \mathbb{R}^{m \times n}$, where $G^x \in \mathbb{R}^{n \times n_g}$, $c^x \in \mathbb{R}^n$, $A^x \in \mathbb{R}^{n_h \times n_g}$, and $b^x \in \mathbb{R}^{n_h}$. Consider the affine transformation $L\mathcal{X} \oplus u = \{LG^x, Lc^x + u, A^x, b^x\}$. Then,

$$O(L\mathcal{X} \oplus u) = O(Lc^x + u) + O(LG^x) = O(mn + m) + O(mn n_g) = O(mn n_g).$$

Minkowski Sum (Property 2.2.7)

Let $\mathcal{X} = \{G^x, c^x, A^x, b^x\} \subset \mathbb{R}^n$ and $\mathcal{W} = \{G^w, c^w, A^w, b^w\} \subset \mathbb{R}^n$, where $G^x \in \mathbb{R}^{n \times n_g}$, $c^x \in \mathbb{R}^n$, $A^x \in \mathbb{R}^{n_h \times n_g}$, $b^x \in \mathbb{R}^{n_h}$, $G^w \in \mathbb{R}^{n \times n_g^w}$, $c^w \in \mathbb{R}^n$, $A^w \in \mathbb{R}^{n_h^w \times n_g^w}$, and $b^w \in \mathbb{R}^{n_h^w}$. Consider the Minkowski sum

$$\mathcal{X} \oplus \mathcal{W} = \left\{ \begin{bmatrix} G^x & G^w \end{bmatrix}, c^w + c^x, \begin{bmatrix} A^x & 0_{n_h \times n_g^w} \\ 0_{n_h^w \times n_g} & A^w \end{bmatrix}, \begin{bmatrix} b^x \\ b^w \end{bmatrix} \right\}.$$

Then,

$$O(\mathcal{X} \oplus \mathcal{W}) = O(c^w + c^x) = O(n).$$

Generalized Intersection (Property 2.2.8)

Let $\mathcal{X} = \{G^x, c^x, A^x, b^x\} \subset \mathbb{R}^n$, $\mathcal{W} = \{G^w, c^w, A^w, b^w\} \subset \mathbb{R}^{n^w}$, and $M \in \mathbb{R}^{n^w \times n}$, where $G^x \in \mathbb{R}^{n \times n_g}$, $c^x \in \mathbb{R}^n$, $A^x \in \mathbb{R}^{n_h \times n_g}$, $b^x \in \mathbb{R}^{n_h}$, $G^w \in \mathbb{R}^{n^w \times n_g^w}$, $c^w \in \mathbb{R}^{n^w}$, $A^w \in \mathbb{R}^{n_h^w \times n_g^w}$, and $b^w \in \mathbb{R}^{n_h^w}$. Consider the generalized intersection

$$\mathcal{X} \cap_M \mathcal{W} = \left\{ \begin{bmatrix} G^x & 0_{n \times n_g^w} \end{bmatrix}, c^x, \begin{bmatrix} A^x & 0_{n_h \times n_g^w} \\ 0_{n_h^w \times n_g} & A^w \\ MG^x & -G^w \end{bmatrix}, \begin{bmatrix} b^x \\ b^w \\ c^w - Mc^x \end{bmatrix} \right\}.$$

Then,

$$\begin{aligned} O(\mathcal{X} \cap_M \mathcal{W}) &= O(MG^x) + O(-G^w) + O(c^w - Mc^x) \\ &= O(n^w n n_g) + O(n^w n_g^w) + O(n^w n + n^w) \\ &= O(n^w n n_g + n^w n_g^w). \end{aligned}$$

Cartesian Product (Property 2.2.9)

Let $\mathcal{X} = \{G^x, c^x, A^x, b^x\} \subset \mathbb{R}^n$ and $\mathcal{W} = \{G^w, c^w, A^w, b^w\} \subset \mathbb{R}^{n^w}$, where $G^x \in \mathbb{R}^{n \times n_g}$, $c^x \in \mathbb{R}^n$, $A^x \in \mathbb{R}^{n_h \times n_g}$, $b^x \in \mathbb{R}^{n_h}$, $G^w \in \mathbb{R}^{n^w \times n_g^w}$, $c^w \in \mathbb{R}^{n^w}$, $A^w \in \mathbb{R}^{n_h^w \times n_g^w}$, and $b^w \in \mathbb{R}^{n_h^w}$. Consider the Cartesian product

$$\mathcal{X} \times \mathcal{W} = \left\{ \begin{bmatrix} G^x & 0_{n \times n_g^w} \\ 0_{n^w \times n_g} & G^w \end{bmatrix}, \begin{bmatrix} c^x \\ c^w \end{bmatrix}, \begin{bmatrix} A^x & 0_{n_h \times n_g^w} \\ 0_{n_h^w \times n_g} & A^w \end{bmatrix}, \begin{bmatrix} b^x \\ b^w \end{bmatrix} \right\}.$$

Then,

$$O(\mathcal{X} \times \mathcal{W}) = O(1).$$

Order Reduction (CZR in (2.31))

Let $\mathcal{X} = \{G^x, c^x, A^x, b^x\} \subset \mathbb{R}^n$, with $G^x \in \mathbb{R}^{n \times n_g}$, $c^x \in \mathbb{R}^n$, $A^x \in \mathbb{R}^{n_h \times n_g}$, and $b^x \in \mathbb{R}^{n_h}$. Let also $\varphi_c \geq 0$ and $\varphi_g \geq (n + \varphi_c)$ be the maximum number of constraints and generators, respectively, to be fixed from $\mathcal{X}$, yielding the reduced-order CZ $\mathcal{X}^l \supset \mathcal{X}$. By applying

the order-reduction algorithm proposed in [Scott et al. \(2016\)](#), the total computational complexity to obtain $\mathcal{X}^\downarrow$ is given by ([Rego et al., 2020a](#)):

$$O(\mathcal{X}^\downarrow) = O\left(k_c(n_h + n_g)^3 + k_c n n_g^2\right) + O\left((n + \varphi_c)^2 n_g + k_g n_g(n + \varphi_c)\right),$$

where $k_c = (n_h - \varphi_c) \geq 0$, $k_g = (n_g - \varphi_g) \geq 0$, and $n_g = (n_g - k_c)$.

Interval Hull (Proposition 2.2.3)

Let $\mathcal{X} = \{G^x, c^x, A^x, b^x\} \subset \mathbb{R}^n$, with $G^x \in \mathbb{R}^{n \times n_g}$, $c^x \in \mathbb{R}^n$, $A^x \in \mathbb{R}^{n_h \times n_g}$, and $b^x \in \mathbb{R}^{n_h}$. Consider the interval hull of $\mathcal{X}$ as $\square\mathcal{X} = [\underline{\zeta}, \bar{\zeta}]$, where

$$\begin{aligned} \underline{\zeta}_i &= \min_{\xi} \{G_i^x \xi + c_i^x : \xi \in \mathcal{B}(A^x, b^x)\}, \quad i = 1, \dots, n, \\ \bar{\zeta}_i &= \max_{\xi} \{G_i^x \xi + c_i^x : \xi \in \mathcal{B}(A^x, b^x)\}, \quad i = 1, \dots, n. \end{aligned}$$

Define $N_d = n_g$ and $N_c = (n_h + 2n_g)$, where n_h and $2n_g$ correspond to the number of equality constraints and inequality constraints on ξ , respectively. Then,

$$O(\square\mathcal{X}) = 2nO\left(n_g(n_h + 2n_g)^3\right) = O\left(n n_g(n_h + n_g)^3\right).$$

Interval Enclosure (Remark 2.4.3)

Let $\mathcal{X} = \{G^x, c^x, A^x, b^x\} \subset \mathbb{R}^n$, with $G^x \in \mathbb{R}^{n \times n_g}$, $c^x \in \mathbb{R}^n$, $A^x \in \mathbb{R}^{n_h \times n_g}$, and $b^x \in \mathbb{R}^{n_h}$. Assume that all constraints n_h of $\mathcal{X}$ are reduced using the pre-conditioning, rescaling, and elimination steps proposed in [Scott et al. \(2016\)](#), whose computational complexity is $O\left(n_h(n_h + n_g)^3 + n_h n n_g^2\right)$ ([Rego et al., 2020a](#)). This procedure yields the zonotope $\tilde{\mathcal{X}} = \{G^{\tilde{x}}, c^{\tilde{x}}\}$, with $G^{\tilde{x}} \in \mathbb{R}^{n \times (n_g - n_h)}$ and $c^{\tilde{x}} \in \mathbb{R}^n$, whose interval hull $\square\tilde{\mathcal{X}}$ is analytically computed as $\square\tilde{\mathcal{X}} = [c^{\tilde{x}} - r, c^{\tilde{x}} + r]$, where $r = |G^{\tilde{x}}| 1_{(n_g - n_h) \times 1}$, with computational burden

$$O(\square\tilde{\mathcal{X}}) = O(n(n_g - n_h)).$$

Then, $\square\mathcal{X}$ can be outer-approximated by $\square\tilde{\mathcal{X}}$ with the following computational cost:

$$\begin{aligned} O(\square\mathcal{X}) &= O\left(n_h(n_h + n_g)^3 + n_h n n_g^2\right) + O(n(n_g - n_h)) \\ &= O\left(n_h(n_h + n_g)^3 + n_h n n_g^2\right). \end{aligned}$$

Outer approximation of CZs by Zonotopes via Interval Hull

Let $\mathcal{X} = \{G^x, c^x, A^x, b^x\} \subset \mathbb{R}^n$, with $G^x \in \mathbb{R}^{n \times n_g}$, $c^x \in \mathbb{R}^n$, $A^x \in \mathbb{R}^{n_h \times n_g}$, and $b^x \in \mathbb{R}^{n_h}$. Consider the zonotope $\tilde{\mathcal{X}} = \{G^{\tilde{x}}, c^{\tilde{x}}\} = G^x [\underline{\zeta}, \bar{\zeta}] \oplus c^x$, where $c^{\tilde{x}} = (c^x + \frac{1}{2}G^x(\underline{\zeta} + \bar{\zeta}))$ and $G^{\tilde{x}} = \frac{1}{2}G^x \text{diag}(\bar{\zeta} - \underline{\zeta})$, that outerapproximates $\mathcal{X}$ using the interval hull of $\mathcal{B}(A^x, b^x)$. Such interval hull can be

computed via LPs or the iterative method proposed in [Scott et al. \(2016\)](#). By employing LPs, the following computational cost is required:

$$\begin{aligned}
 O(\tilde{\mathcal{X}}) &= O(c^{\bar{x}}) + O(G^{\bar{x}}) + O(\square \mathcal{B}(A^x, b^x)) \\
 &= O(nn_g) + O(nn_g^2) + 2n_g O(n_g(n_h + 2n_g)^3) \\
 &= O(nn_g^2 + n_g^2(n_h + n_g)^3).
 \end{aligned}$$

On the other hand, the iterative algorithm results in a smaller computational complexity:

$$\begin{aligned}
 O(\tilde{\mathcal{X}}) &= O(c^{\bar{x}}) + O(G^{\bar{x}}) + O(\square \mathcal{B}(A^x, b^x)) \\
 &= O(nn_g) + O(nn_g^2) + O(n_h^2 n_g + n_h n_g^2) \\
 &= O(nn_g^2 + n_h^2 n_g + n_h n_g^2).
 \end{aligned}$$

Outer approximation of CZs by Parallelotopes (Proposition 2.3.2)

Let $\mathcal{X} = \{G^x, c^x, A^x, b^x\} \subset \mathbb{R}^n$, with $G^x \in \mathbb{R}^{n \times n_g}$, $c^x \in \mathbb{R}^n$, $A^x \in \mathbb{R}^{n_h \times n_g}$, and $b^x \in \mathbb{R}^{n_h}$. Let $\mathcal{C} = \{G^c, c^c\}$ be the parallelotopic outer approximation of $\mathcal{X}$ obtained by Remark 2.3.1, where $G^c \in \mathbb{R}^{n \times n}$ and $c^c \in \mathbb{R}^n$. Let also $\mathcal{T} = \{G^t, c^t\}$ be the parallelotope obtained by applying Proposition 2.3.2 to the parallelotope $\mathcal{C}$, with $G^t \in \mathbb{R}^{n \times n}$ and $c^t \in \mathbb{R}^n$. Then,

$$\begin{aligned}
 O(\mathcal{T}) &= O(\mathcal{C}) + O(\text{Proposition 2.3.2}) \\
 &= O(\text{CZR}(\mathcal{X}, 0, n)) + O(\text{LPs}) + O(G^t) + O(c^t) \\
 &= O(n_h(n_h + n_g)^3 + n_h nn_g^2 + n^2(n_g - n_h) + (n_g - n_h - n)(n_g - n_h)n \\
 &\quad + 2nO((n + n_g)(2n + 2n_g + n_h + 1)^3) + n^3 + n^2) \\
 &= O(n_h(n_h + n_g)^3 + n_h nn_g^2 + n^2(n_g - n_h) + (n_g - n_h - n)(n_g - n_h)n \\
 &\quad + n(n + n_g)(n + n_h + n_g)^3) \\
 &= O(n_h(n_h + n_g)^3 + n(n + n_g)(n + n_h + n_g)^3).
 \end{aligned}$$

Closest Point (1-norm)

Let $\mathcal{X} = \{G^x, c^x, A^x, b^x\} \subset \mathbb{R}^n$ and $z \in \mathbb{R}^n$, with $G^x \in \mathbb{R}^{n \times n_g}$, $c^x \in \mathbb{R}^n$, $A^x \in \mathbb{R}^{n_h \times n_g}$, and $b^x \in \mathbb{R}^{n_h}$. Consider the optimization problem $\min_{x \in \mathcal{X}} \|x - z\|_1$ to determine the optimal solution ξ^* , and thereby, the punctual estimate $x^* = (G^x \xi^* + c^x) \in \mathcal{X}$. Using LPs with $N_d = (n + n_g)$ and $N_c = (2n + 2n_g + n_h)$, the following computational complexity is required:

$$\begin{aligned}
 O(\text{closest}) &= O(G^x \xi^* + c^x) + O(N_d N_c^3) \\
 &= O(nn_g + n) + O((n + n_g)(2n + 2n_g + n_h)^3)
 \end{aligned}$$

$$= O\left(\left(n + n_g\right)\left(n + n_g + n_h\right)^3\right).$$

Enumeration of Vertices

Let $\mathcal{Z} = \{G^z, c^z\} \subset \mathbb{R}^n$ be either a box or a parallelotope, with $G^z \in \mathbb{R}^{n \times n}$ and $c^z \in \mathbb{R}^n$. Let the enumeration of vertices for $\mathcal{Z}$ be given by

$$\text{vert}(\mathcal{Z}) = G^z \text{vert}(\mathcal{B}^n) + \bar{z} \mathbf{1}_{1 \times 2^n},$$

where $\text{vert}(\mathcal{B}^n)$ is the set of vertices for the unitary box. Then,

$$\begin{aligned} O(\text{vert}(\mathcal{Z})) &= O(G^z \text{vert}(\mathcal{B}^n)) + O(\bar{z} \mathbf{1}_{1 \times 2^n}) + O(\text{sum}) \\ &= O(n^2 2^n) + O(n 2^n) + O(n 2^n) \\ &= O(n^2 2^n). \end{aligned}$$

Lower Bound of Eigenvalue (Algorithm 2.1)

Let $[H_1], \dots, [H_m]$ be $(n \times n)$ -dimensional interval Hessian matrices, $\text{vert}(\mathcal{B}^n)$ be a set of 2^n vertices, and $S \in \mathbb{R}^{n \times N}$ the first $N = 2^{n-1}$ non-symmetric vertices. In the following, the complexity of LBE (Algorithm 2.1) is derived.

$$\begin{aligned} O(\text{Steps 1-2}) &= O(1), \\ O(\text{Steps 3-5}) &= O(mn^2), \\ O(\text{Steps 9-19}) &= O(m), \\ O(\text{Steps 8-20}) &= O((n-2)m), \\ O(\text{Steps 7-21}) &= O((n-1)(n-2)m), \\ O(\text{Steps 22-27}) &= O(mn^3), \\ O(\text{Steps 6-28}) &= O(N((n-1)(n-2)m + mn^3)). \end{aligned}$$

Then, $O(\text{LBE}) = O(Nmn^3)$.

Enclosure for DC Programming (Lemma 2.3.1)

Let $\varrho(z, u) = (\varrho^a(z, u) - \varrho^b(z, u)) \in \mathbb{R}^m$ be DC on $\mathcal{Z} = \{G^z, c^z, A^z, b^z\} \subset \mathbb{R}^n$, with $z \in \mathcal{Z}$, $u \in \mathbb{R}^p$ a deterministic input, $G^z \in \mathbb{R}^{n \times n_g}$, $c^z \in \mathbb{R}^n$, $A^z \in \mathbb{R}^{n_h \times n_g}$, and $b^z \in \mathbb{R}^{n_h}$. Let $\mathcal{T} = \mathcal{P} \supset \mathcal{Z}$ be the parallelotope given by Proposition 2.3.2. Let also $\bar{z} \in \mathcal{T}$ be any interior point. Assume that ϱ^a , ϱ^b , and $\bar{z}$ are known. Then, the computational cost to obtain the linearization enclosure $\mathcal{R} = \{G^e, c^e\} \subset \mathbb{R}^m$ in Lemma 2.3.1 is given by

$$O(\mathcal{R}) = O(\mathcal{T}) + O(\text{vert}(\mathcal{T})) + O((2.40), (2.41)) + O(G^e) + O(c^e)$$

$$\begin{aligned}
&= O\left(n_h(n_h + n_g)^3 + n_h n n_g^2 + n^2(n_g - n_h) + (n_g - n_h - n)(n_g - n_h)n\right. \\
&\quad \left.+ n(n + n_g)(n + n_h + n_g)^3\right) + O(m2^n) + O(n^2 2^n) + O(m) + O(m) \\
&= O\left(n_h(n_h + n_g)^3 + n(n + n_g)(n + n_h + n_g)^3 + (m + n^2)2^n\right).
\end{aligned}$$

Enclosure for First-Order Taylor Expansion

Let $\mathcal{Z} \subset \mathbb{R}^n$ be a CZ with n_g generators and n_h constraints, and $\varrho: \mathbb{R}^n \times \mathbb{R}^p \rightarrow \mathbb{R}^m$ be a function with deterministic input $u \in \mathbb{R}^p$. According to [Rego et al. \(2020a\)](#) and [Rego et al. \(2021\)](#), the linearization remainder $\mathcal{R}$ has $0.5n_g^2 + 1.5n_g + 2m$ generators and $0.5n_h^2 + 1.5n_h$ constraints. Moreover,

$$O(\mathcal{R}) = O\left(m(n^2 n_g + n n_g^2) + (n n_g + n_h)(n_g + n_h)^3\right).$$

Mixed-Uncertainty Vectors

Confidence Constrained Zonotope (Definition 2.4.3)

Let $c^x \in \mathbb{R}^n$, $\mathcal{X} = \{G^x, 0_{n \times 1}, A^x, b^x\} \subset \mathbb{R}^n$, $X \sim \mathcal{N}(0_{n \times 1}, P^{xx})$, and $\alpha \in [0, 1]$, with $G^x \in \mathbb{R}^{n \times n_g}$, $A^x \in \mathbb{R}^{n_h \times n_g}$, $b^x \in \mathbb{R}^{n_h}$, and $P^{xx} \in \mathbb{R}^{n \times n}$. Consider the confidence CZ $\mathcal{X}^\alpha = (c^x \oplus \mathcal{X} \oplus \mathcal{P})$, where $\mathcal{P} = \{G^e, 0_{n \times 1}\}$, $G^e = (\varsigma P^{xx})^{1/2}$, and $\varsigma \geq 0$ is a chi-square realization that accords with α and X . Then,

$$\begin{aligned}
O(\mathcal{X}^\alpha) &= O(\text{sum}) + O(\text{Cholesky}) \\
&= O(n) + O(n^3) \\
&= O(n^3).
\end{aligned}$$

Confidence Box (Definition 2.4.4)

Let $c^x \in \mathbb{R}^n$, $\mathcal{X} = \{G^x, 0_{n \times 1}, A^x, b^x\} \subset \mathbb{R}^n$, $X \sim \mathcal{N}(0_{n \times 1}, P^{xx})$, and $\alpha \in [0, 1]$, with $G^x \in \mathbb{R}^{n \times n_g}$, $A^x \in \mathbb{R}^{n_h \times n_g}$, $b^x \in \mathbb{R}^{n_h}$, and $P^{xx} \in \mathbb{R}^{n \times n}$. Consider the confidence box $\mathcal{I}^\alpha = (c^x \oplus \square \mathcal{X} \oplus \square \mathcal{E})$, where $\mathcal{E}$ is the confidence ellipsoid related to α and X . Then,

$$\begin{aligned}
O(\mathcal{I}^\alpha) &= O(\text{sum}) + O(\square \mathcal{X}) + O(\square \mathcal{E}) \\
&= O(n) + O\left(n n_g (n_h + n_g)^3\right) + O(n) \\
&= O\left(n n_g (n_h + n_g)^3\right).
\end{aligned}$$

E.2 Stochastic Approaches

Auxiliary Algorithms

UT (Algorithm A.3)

Consider the GRVs $X_1 \sim \mathcal{N}(\hat{x}_1, P^{x_1 x_1})$ and $X_2 \sim \mathcal{N}(\hat{x}_2, P^{x_2 x_2})$, with $\hat{x}_1 \in \mathbb{R}^n$, $P^{x_1 x_1} \in \mathbb{R}^{n \times n}$, $\hat{x}_2 \in \mathbb{R}^r$, and $P^{x_2 x_2} \in \mathbb{R}^{r \times r}$, the deterministic vector $d \in \mathbb{R}^p$, and the nonlinear transformation $h : \mathbb{R}^n \times \mathbb{R}^r \times \mathbb{R}^p \rightarrow \mathbb{R}^m$.

$$\begin{aligned}
 O(\text{Step 1}) &= O(1), \\
 O(\text{Step 2}) &= O(1), \\
 O(\text{Step 3}) &= O((n+r)^3), \\
 O(\text{Step 4}) &= O(n+r), \\
 O(\text{Step 5}) &= O(1), \\
 O(\text{Step 6}) &= O(m(n+r)), \\
 O(\text{Step 7}) &= O(m^2(n+r)), \\
 O(\text{Step 8}) &= O(mn(n+r)).
 \end{aligned}$$

Then,

$$O(\text{UT}) = O((n+r)^3 + m^2(n+r) + mn(n+r)).$$

ICUT (Algorithm A.6)

Consider the GRV $X \sim \mathcal{N}(\hat{x}, P^{xx})$, with $\hat{x} \in \mathbb{R}^n$ and $P^{xx} \in \mathbb{R}^{n \times n}$, the interval constraints $a \leq b$, with $a, b \in \mathbb{R}^n$, and spreading parameter $\kappa > -n$.

$$\begin{aligned}
 O(\text{Steps 1–9}) &= O(n), \\
 O(\text{Step 10}) &= O(n^3), \\
 O(\text{Step 11}) &= O(n^2), \\
 O(\text{Steps 12–23}) &= O(n^2), \\
 O(\text{Step 24}) &= O(n^2), \\
 O(\text{Step 25}) &= O(n^2), \\
 O(\text{Step 26}) &= O(n^2), \\
 O(\text{Step 27}) &= O(n), \\
 O(\text{Step 28}) &= O(1), \\
 O(\text{Step 29}) &= O(n), \\
 O(\text{Step 30}) &= O(n^2),
 \end{aligned}$$

$$O(\text{Step 31}) = O(n^3),$$

$$O(\text{Step 32}) = O(1).$$

Then, $O(\text{ICUT}) = O(n^3)$.

PT (Algorithm A.5)

Consider the GRV $X \sim \mathcal{N}(\hat{x}, P^{\text{xx}})$, with $\hat{x} \in \mathbb{R}^n$ and $P^{\text{xx}} \in \mathbb{R}^{n \times n}$, and the interval constraints $a \leq b$, with $a, b \in \mathbb{R}^n$.

$$O(\text{Steps 1–2}) = O(1),$$

$$O(\text{Step 4}) = O(n^3),$$

$$O(\text{Step 5}) = O(n^2),$$

$$O(\text{Step 7}) = O(1),$$

$$O(\text{Steps 8–10}) = O((j-1)n),$$

$$O(\text{Step 11}) = O(n),$$

$$O(\text{Steps 12–18}) = O((j-1)n),$$

$$O(\text{Step 19}) = O(n),$$

$$O(\text{Steps 6–20}) = O(n^3),$$

$$O(\text{Step 21–27}) = O(1),$$

$$O(\text{Step 28}) = O(n^3),$$

$$O(\text{Step 29}) = O(n^3),$$

$$O(\text{Step 3–30}) = O(n^4),$$

$$O(\text{Step 31–33}) = O(1).$$

Then,

$$O(\text{PT}) = O(n^4).$$

KP (Algorithm A.1)

Step 1

$$O(\hat{x}_{k|k-1}) = O(n^2 + np),$$

$$O(\hat{y}_{k-1}) = O(mn),$$

$$O(\hat{P}_{k|k-1}^{\text{xx}}) = O(n^3),$$

$$O(\hat{P}_{k-1}^{\text{xy}}) = O(n^3 + n^2m),$$

$$\begin{aligned} O(\hat{P}_{k-1}^{yy}) &= O(n^2m + nm^2), \\ O(\check{K}_{k-1}) &= O(m^3 + nm^2). \end{aligned}$$

Then,

$$O(\text{Step 1}) = O(n^3 + m^3 + np + n^2m + nm^2).$$

Step 2

$$\begin{aligned} O(\hat{x}_k) &= O(nm), \\ O(\hat{P}_k^{xx}) &= O(n^3 + n^2m + nm^2). \end{aligned}$$

Then,

$$O(\text{Step 2}) = O(n^3 + n^2m + nm^2).$$

All Steps

$$O(\text{KP}) = O(n^3 + m^3 + np + n^2m + nm^2).$$

SKP

$$\begin{aligned} O(\text{SKP}) &= O(\hat{x}_k) + O(\hat{P}_k^{xx}) \\ &= O(n^3 + np + n^2m + nm^2). \end{aligned}$$

KF (Algorithm A.2)

Step 1

$$\begin{aligned} O(\hat{x}_{k|k-1}) &= O(n^2 + np), \\ O(\hat{y}_{k|k-1}) &= O(mn), \\ O(\hat{P}_{k|k-1}^{xx}) &= O(n^3), \\ O(\hat{P}_{k|k-1}^{xy}) &= O(n^2m), \\ O(\hat{P}_{k|k-1}^{yy}) &= O(n^2m + nm^2), \\ O(\check{K}_k) &= O(m^3 + nm^2). \end{aligned}$$

Then,

$$O(\text{Step 1}) = O(n^3 + m^3 + np + n^2m + nm^2).$$

Step 2

$$\begin{aligned} O(\hat{x}_k) &= O(nm), \\ O(\hat{P}_k^{\text{xx}}) &= O(n^3 + n^2m + nm^2). \end{aligned}$$

Then,

$$O(\text{Step 2}) = O(n^3 + n^2m + nm^2).$$

All Steps

$$O(\text{KF}) = O(n^3 + m^3 + np + n^2m + nm^2).$$

SKF

$$\begin{aligned} O(\text{SKF}) &= O(\hat{x}_k) + O(\hat{P}_k^{\text{xx}}) \\ &= O(n^3 + np + n^2m + nm^2). \end{aligned}$$

DGF (Algorithm 3.1)

Step 1

$$\begin{aligned} O(\hat{P}_{k|k-1}^{\text{xx}}) &= O(n^3), \\ O(\hat{P}_{k-1}^{\text{xy}}) &= O(n^3 + n^2m), \\ O(\hat{P}_{k-1}^{\text{yy}}) &= O(n^2m + nm^2), \\ O(\Pi_k) &= O(1), \\ O(\Sigma_k) &= O((n + 2m)^2) = O((n + m)^2), \\ O(\check{\Lambda}_k) &= O(n^2 + n^3 + (n + 2m)^3 + n(n + 2m)^2 + n^2(n + 2m)) = O((n + m)^3), \\ O(\check{\Psi}_k) &= O(n(n + 2m) + n^2(n + 2m) + n(n + 2m)^2) = O(n(n + m)^2). \end{aligned}$$

Then,

$$O(\text{Step 1}) = O((n + m)^3).$$

Step 2

$$O(\check{T}_k, \check{N}_k, \check{K}_{k-1}) = O(1).$$

Step 3

$$\begin{aligned} O(\hat{x}_k) &= O(n^3 + n^2p + n^2m), \\ O(\hat{P}_k^{\text{xx}}) &= O(n^3 + n^2m + nm^2). \end{aligned}$$

Then,

$$O(\text{Step 3}) = O(n^3 + n^2p + n^2m + nm^2).$$

All Steps

$$O(\text{DGF}) = O((n + m)^3 + n^2p).$$

DH₂GF (Algorithm 3.2)

$$\begin{aligned} O(\text{DH}_2\text{GF}) &= O(\hat{x}_k) + O(\hat{P}_k^{\text{xx}}) \\ &= O(n^3 + n^2p + n^2m + nm^2). \end{aligned}$$

UKF (Algorithm A.4)

$$\begin{aligned} O(\text{Step 1}) &= O((n + q)^3), \\ O(\text{Step 2}) &= O((n + r)^3 + m^2(n + r) + mn(n + r)), \\ O(\text{Step 3}) &= O(m^3 + nm^2), \\ O(\text{Step 4}) &= O(nm), \\ O(\text{Step 5}) &= O(n^2 + nm^2). \end{aligned}$$

Then,

$$\begin{aligned} O(\text{UKF}) &= O((n+q)^3 + (n+r)^3 + m^3 + m^2(n+r) + mn(n+r)) \\ &= O(n^3 + q^3 + r^3 + m^3 + n^2(q+r) + n(q^2 + r^2) + (m^2 + mn)(n+r)). \end{aligned}$$

TUKF (Algorithm A.7)

$$\begin{aligned} O(\text{Step 1}) &= O(\text{UKF}), \\ O(\text{Step 2}) &= O(n^4). \end{aligned}$$

Then,

$$O(\text{TUKF}) = O(n^4 + (n+q)^3 + (n+r)^3 + m^3 + m^2(n+r) + mn(n+r)).$$

E.3 Set-based Approaches

Auxiliary Algorithms

ZT1 (Algorithm B.10)

Consider the zonotope $\mathcal{X}_1 = G^{x_1} \mathcal{B}_g^{n_{x_1}} \oplus \hat{x}_1 \subset \mathbb{R}^n$ and $\mathcal{X}_2 = G^{x_2} \mathcal{B}_g^{n_{x_2}} \oplus \hat{x}_2 \subset \mathbb{R}^r$, the deterministic vector $d \in \mathbb{R}^p$, and the nonlinear transformation $h : \mathbb{R}^n \times \mathbb{R}^r \times \mathbb{R}^p \rightarrow \mathbb{R}^m$.

$$\begin{aligned} O(\text{Steps 1–3}) &= O(1), \\ O(\text{Step 4}) &= O([M]G^x) + O(\text{Proposition 2.2.2}) \\ &= O(m(n+r)(n_g^{x_1} + n_g^{x_2})). \end{aligned}$$

Then,

$$O(\text{ZT1}) = O(m(n+r)(n_g^{x_1} + n_g^{x_2})).$$

ZT2 (Algorithm B.11)

Consider the zonotope $\mathcal{X}_1 = G^{x_1} \mathcal{B}_g^{n_{x_1}} \oplus \hat{x}_1 \subset \mathbb{R}^n$ and $\mathcal{X}_2 = G^{x_2} \mathcal{B}_g^{n_{x_2}} \oplus \hat{x}_2 \subset \mathbb{R}^r$, the deterministic vector $d \in \mathbb{R}^p$, and the nonlinear transformation $h : \mathbb{R}^n \times \mathbb{R}^r \times \mathbb{R}^p \rightarrow \mathbb{R}^m$.

$$\begin{aligned} O(\text{Steps 1–3}) &= O(1), \\ O(\text{Step 4}) &= O([M_1]G^{x_1}) + O([M_2]G^{x_2}) + O(\text{Proposition 2.2.2}) \\ &= O(m(n n_g^{x_1} + r n_g^{x_2})). \end{aligned}$$

Then,

$$O(\text{ZT2}) = O\left(m\left(nn_g^{x_1} + rn_g^{x_2}\right)\right).$$

ZT3 (Algorithm B.12)

Consider the zonotope $\mathcal{X}_1 = G^{x_1} \mathcal{B}_g^{n_g^{x_1}} \oplus \hat{x}_1 \subset \mathbb{R}^n$ and $\mathcal{X}_2 = G^{x_2} \mathcal{B}_g^{n_g^{x_2}} \oplus \hat{x}_2 \subset \mathbb{R}^r$, the deterministic vector $d \in \mathbb{R}^p$, and the nonlinear transformation $h : \mathbb{R}^n \times \mathbb{R}^r \times \mathbb{R}^p \rightarrow \mathbb{R}^m$.

$$\begin{aligned} O(\text{Step 1}) &= O(G^a) + O(\hat{y}) \\ &= O(m), \\ O(\text{Step 2}) &= O(1), \\ O(\text{Step 3}) &= O\left(mnn_g^{x_1}\right), \\ O(\text{Step 4}) &= O(1). \end{aligned}$$

Then,

$$O(\text{ZT3}) = O\left(mnn_g^{x_1}\right).$$

LD (Algorithm D.1)

Consider the zonotopes $\hat{\mathcal{X}}_{k-1} = \hat{G}_{k-1}^x \mathcal{B}_g^{n_g^x} \oplus \hat{x}_{k-1}$, $\mathcal{W}_{k-1} = G_{k-1}^w \mathcal{B}_g^{n_g^w} \oplus c_{k-1}^w$, and $\bar{\mathcal{W}}_{k-1} = G_{k-1}^{\bar{w}} \mathcal{B}_g^{n_g^{\bar{w}}} \oplus c_{k-1}^{\bar{w}}$, the reference input $u_{k-1}^{\text{tr}} \in \mathbb{R}^p$, the control input $u_{k-1} \in \mathbb{R}^p$, and the function $f : \mathbb{R}^n \times \mathbb{R}^p \rightarrow \mathbb{R}^n$ whose noise input is affine.

$$\begin{aligned} O(\text{Step 1}) &= O(F^{\text{xd}}) + O(F^{\text{ud}}) + O(F^{\text{wd}}) \\ &= O(n^3 + n^2p), \\ O(\text{Step 2}) &= O(1), \\ O(\text{Step 3}) &= O\left(\hat{G}_{k|k-1}^x\right) + O(\hat{x}_{k|k-1}) \\ &= O\left(n^2n_g^x + n^2n_g^w + np\right). \end{aligned}$$

Then,

$$O(\text{LD}) = O\left(n^2\left(n + p + n_g^x + n_g^w\right)\right).$$

DA (Algorithm D.2)

Consider the zonotopes $\hat{\mathcal{X}}_{k-1} = \hat{G}_{k-1}^x \mathcal{B}_g^{n_g^x} \oplus \hat{x}_{k-1}$, $\mathcal{W}_{k-1} = G_{k-1}^w \mathcal{B}_g^{n_g^w} \oplus c_{k-1}^w$, and $\bar{\mathcal{W}}_{k-1} = G_{k-1}^{\bar{w}} \mathcal{B}_g^{n_g^{\bar{w}}} \oplus c_{k-1}^{\bar{w}}$, the reference input $u_{k-1}^{\text{tr}} \in \mathbb{R}^p$, the control input $u_{k-1} \in \mathbb{R}^p$, and the function $f : \mathbb{R}^n \times \mathbb{R}^p \rightarrow \mathbb{R}^n$ whose noise input is affine.

$$\begin{aligned} O(\text{Step 1}) &= O(n^2), \\ O(\text{Step 2}) &= O(n), \end{aligned}$$

$$\begin{aligned} O(\text{Step 3}) &= O(n^2 n_g^x), \\ O(\text{Step 4}) &= O(nn_g^w). \end{aligned}$$

Then,

$$O(\text{DA}) = O(n^2 n_g^x + nn_g^w).$$

FL (Algorithm D.3)

Consider the zonotopes $\hat{\mathcal{X}}_{k|k-1} = \hat{G}_{k|k-1}^x \mathcal{B}_s^{n_x} \oplus \hat{x}_{k|k-1}$, $\mathcal{V}_k = G_k^v \mathcal{B}_s^{n_v} \oplus c_k^v$, and $\bar{\mathcal{V}}_k = G_k^{\bar{v}} \mathcal{B}_s^{n_{\bar{v}}} \oplus c_k^{\bar{v}}$, the measurement $y_{i,k} \in \mathbb{R}^m$, and the function $h_i : \mathbb{R}^n \rightarrow \mathbb{R}$ whose noise input is affine.

$$\begin{aligned} O(\text{Step 1}) &= O(1), \\ O(\text{Step 2}) &= O(n), \\ O(\text{Step 3}) &= O(n + n_g^v + n_g^{\bar{v}}), \\ O(\text{Step 4}) &= O(n + n_g^v + n_g^{\bar{v}}), \\ O(\text{Steps 5–6}) &= O(1). \end{aligned}$$

Then,

$$O(\text{FL}) = O(n + n_g^v + n_g^{\bar{v}}).$$

SM (Algorithm B.7)

Consider the zonotope $\mathcal{X} = G^x \mathcal{B}^{n_g} \oplus \hat{x} \subset \mathbb{R}^n$ and the strip $\mathcal{S} = \{x \in \mathbb{R}^n : |c^\top x - d| \leq \rho\}$.

$$\begin{aligned} O(\text{Step 1}) &= O(\check{\lambda}_k) \\ &= O(n^2 n_g), \\ O(\text{Step 2}) &= O(\bar{G}_k^x) + O(\bar{x}_k) \\ &= O(n^2 n_g). \end{aligned}$$

Then,

$$O(\text{SM}) = O(n^2 n_g).$$

VM2 (Algorithm B.9)

Consider the zonotope $\mathcal{X} = G^x \mathcal{B}^{n_g} \oplus \hat{x} \subset \mathbb{R}^n$ and the strip $\mathcal{S} = \{x \in \mathbb{R}^n : |c^\top x - d| \leq \rho\}$.

$$\begin{aligned} O(\text{Steps 1–2}) &= O(1), \\ O(\text{Step 3}) &= O(N(n_g, n)n^3), \\ O(\text{Step 6–7}) &= O(nn_g), \\ O(\text{Step 8}) &= O(n), \end{aligned}$$

$$\begin{aligned}
O(\text{Step 9}) &= O\left(N(n_g, n)n^3\right), \\
O(\text{Steps 10–14}) &= O(1), \\
O(\text{Steps 4–16}) &= O\left(n_g\left(nn_g + N(n_g, n)n^3\right)\right), \\
O(\text{Step 17}) &= O(1).
\end{aligned}$$

Then,

$$O(\text{VM2}) = O\left(n_g\left(nn_g + N(n_g, n)n^3\right)\right).$$

VM3 (Algorithm 3.7)

Consider the zonotope $\mathcal{X} = G^x \mathcal{B}^{n_g} \oplus \hat{x} \subset \mathbb{R}^n$ and the strip $\mathcal{S} = \{x \in \mathbb{R}^n : |c^\top x - d| \leq \rho\}$.

$$\begin{aligned}
O(\text{Step 1}) &= O\left(n^3 N(n_g, n) + N(n_g, n-1)(n^3 + n(n-1)^2)\right) \\
&= O\left(n^3 \left(N(n_g, n) + N(n_g, n-1)\right)\right), \\
O(\text{Step 2}) &= O(1), \\
O(\text{Step 3}) &= O\left((n+1 + N(n_g, n-1))\left(2(1 + N(n_g, n-1))\right)^3\right) \\
&= O\left((n + N(n_g, n-1))\left(N(n_g, n-1)\right)^3\right), \\
O(\text{Step 4}) &= O(1), \\
O(\text{Step 5}) &= O\left(n^2 n_g\right).
\end{aligned}$$

Then,

$$O(\text{VM3}) = O\left(n^3 \left[N(n_g, n) + N(n_g, n-1)\right] + \left[n + N(n_g, n-1)\right] \left[N(n_g, n-1)\right]^3 + n^2 n_g\right).$$

DRZF (Algorithm 3.3)

Step 1

$$O(\text{ZR}) = O\left(n_g(n + \log(n_g))\right).$$

Step 2

$$\begin{aligned}
O(F) &= O\left(n^2 \varphi_g\right), \\
O(H) &= O\left(mn \varphi_g\right), \\
O(\psi) &= O(1), \\
O(D^i) &= O\left(n \varphi_g + \varphi_g^2 + m \varphi_g + mn_g^w + \left(n_g^w\right)^2 + mn_{g,k-1}^v + \left(n_{g,k-1}^v\right)^2 + mn_{g,k}^v + \left(n_{g,k}^v\right)^2\right), \\
O(d^i) &= O(1),
\end{aligned}$$

$$O(D^e) = O(1).$$

Then,

$$O(\text{Step 2}) = O\left(n^2\varphi_g + mn\varphi_g + \varphi_g^2 + nn_g^w + (n_g^w)^2 + mn_{g,k-1}^v + (n_{g,k-1}^v)^2 + mn_{g,k}^v + (n_{g,k}^v)^2\right).$$

Steps 3 up to 6

$$\begin{aligned} O(\text{Steps 3--6}) &= O\left(n\left(n + 2m + \varphi_g + n_g^w + n_{g,k-1}^v + n_{g,k}^v\right)\left(2\left(\varphi_g + n_g^w + n_{g,k-1}^v + n_{g,k}^v\right) + n\right)^3\right) \\ &= O\left(n\left(n + m + \varphi_g + n_g^w + n_{g,k-1}^v + n_{g,k}^v\right)\left(n + \varphi_g + n_g^w + n_{g,k-1}^v + n_{g,k}^v\right)^3\right). \end{aligned}$$

Step 7

$$\begin{aligned} O(\hat{x}_k) &= O(n^3 + n^2p + n^2m), \\ O(\hat{G}_k^x) &= O\left(n^2\left(n + m + \varphi_g + n_g^w\right) + nm n_{g,k-1}^v + nm n_{g,k}^v\right). \end{aligned}$$

Then,

$$O(\text{Step 7}) = O\left(n^2\left(n + p + m + \varphi_g + n_g^w\right) + nm\left(n_{g,k-1}^v + n_{g,k}^v\right)\right).$$

ZKP (Algorithm B.2)

Step 1

$$O(\text{ZR}) = O\left(n_g\left(n + \log(n_g)\right)\right).$$

Step 2

$$\begin{aligned} O(\hat{x}_{k|k-1}) &= O(n^2 + np), \\ O(\hat{y}_{k-1}) &= O(mn), \\ O(\hat{C}_{k-1}^{xx}) &= O(n^2\varphi_g), \\ O(\hat{C}_{k-1}^{xy}) &= O(n^3 + n^2m), \\ O(C_{k-1}^{vv}) &= O(m^2n_g^v), \\ O(\hat{C}_{k-1}^{yy}) &= O(n^2m + nm^2). \end{aligned}$$

Then,

$$O(\text{Step 2}) = O(np + n^2\varphi_g + n^3 + n^2m + nm^2 + m^2n_g^v).$$

Step 3

$$\begin{aligned} O(\check{K}_{k-1}) &= O(m^3 + nm^2), \\ O(\hat{x}_k) &= O(nm), \\ O(\hat{G}_k^x) &= O(n^2m + n^2\varphi_g + nmn_g^v). \end{aligned}$$

Then,

$$O(\text{Step 3}) = O(m^3 + nm^2 + n^2m + n^2\varphi_g + nmn_g^v).$$

All Steps

$$\begin{aligned} O(\text{ZKP}) &= O(n_g(n + \log(n_g)) + np + n^2\varphi_g + n^3 + n^2m + nm^2 + m^2n_g^v + m^3 + nmn_g^v) \\ &= O(n_g(n + \log(n_g)) + n(p + mn_g^v) + n^2(n + m + \varphi_g) + m^2(n + m + n_g^v)). \end{aligned}$$

ZKF (Algorithm 3.4)

Step 1

$$O(\text{ZR}) = O(n_g(n + \log(n_g))).$$

Step 2

$$\begin{aligned} O(\hat{x}_{k|k-1}) &= O(n^2 + np), \\ O(\hat{y}_{k|k-1}) &= O(mn), \\ O(\hat{C}_{k|k-1}^{xx}) &= O(n^3 + n^2\varphi_g + n^2n_g^w), \\ O(\hat{C}_{k|k-1}^{xy}) &= O(n^2m), \\ O(C_k^{vv}) &= O(m^2n_g^v), \\ O(\hat{C}_{k|k-1}^{yy}) &= O(n^2m + nm^2). \end{aligned}$$

Then,

$$O(\text{Step 2}) = O(np + n^3 + n^2\varphi_g + n^2n_g^w + n^2m + nm^2 + m^2n_g^v).$$

Step 3

$$\begin{aligned}
O(\check{K}_k) &= O(m^3 + nm^2), \\
O(\hat{x}_k) &= O(nm), \\
O(\hat{G}_k^x) &= O(n^2m + n^2(\varphi_g + n_g^w) + nmn_g^v).
\end{aligned}$$

Then,

$$O(\text{Step 3}) = O(m^3 + nm^2 + n^2(m + \varphi_g + n_g^w) + nmn_g^v).$$

All Steps

$$\begin{aligned}
O(\text{ZKF}) &= O(n_g(n + \log(n_g)) + np + n^3 + nm^2 + m^2n_g^v + m^3 + n^2(m + \varphi_g + n_g^w) + nmn_g^v) \\
&= O(n_g(n + \log(n_g)) + n(p + mn_g^v) + n^2(n + n_g^w + m + \varphi_g) + m^2(n + m + n_g^v)).
\end{aligned}$$

NH_∞ZF (Algorithm 3.5)**Step 1**

$$\begin{aligned}
O(\Pi_k^x) &= O(n^2n_g), \\
O(\Pi_k^w) &= O(n^2n_g^w), \\
O(\Pi_k^v) &= O(n^2n_g^v), \\
O(\zeta_k^w) &= O(nn_g^w), \\
O(\zeta_k^v) &= O(nn_g^v).
\end{aligned}$$

Then,

$$O(\text{Step 1}) = O(n^2(n_g + n_g^w + n_g^v)).$$

Step 2

$$O(\text{Step 2}) = O(mn_g).$$

Step 3

$$\begin{aligned}
O(\tilde{A}\hat{x}_{k-1}) &= O(n^2), \\
O(\tilde{D}u_{k-1}) &= O(np), \\
O(\check{K}y_k) &= O(nm).
\end{aligned}$$

Then,

$$O(\text{Step 1}) = O(n(n + p + m)).$$

Step 4

$$O(\text{Step 4}) = O(n).$$

All Steps

$$O(\text{NH}_\infty\text{ZF}) = O\left(n^2\left(n_g + n_g^w + n_g^v\right) + n(p + m)\right).$$

DTZF (Algorithm B.3)**Step 1**

$$O(\text{ZR}) = O\left(n_g\left(n + \log(n_g)\right)\right).$$

Step 2

$$\begin{aligned}
O(\hat{C}_{k-1}^{\text{xx}}) &= O(n^2\varphi_g), \\
O(\hat{C}_{k|k-1}^{\text{xx}}) &= O(n^3 + n^2n_g^w), \\
O(\hat{C}_{k-1}^{\text{xy}}) &= O(n^3 + n^2m), \\
O(C_{k-1}^{\text{vv}}) &= O(m^2n_{g,k-1}^v), \\
O(C_k^{\text{vv}}) &= O(m^2n_{g,k}^v), \\
O(\hat{C}_{k-1}^{\text{yy}}) &= O(n^2m + nm^2), \\
O(\Pi_k) &= O(1), \\
O(\Sigma_k) &= O((n + 2m)^2) = O((n + m)^2), \\
O(\check{\Lambda}_k) &= O(n^2 + n^3 + (n + 2m)^3 + n(n + 2m)^2 + n^2(n + 2m))
\end{aligned}$$

$$\begin{aligned}
&= O((n+m)^3), \\
O(\Psi_k) &= O(n(n+2m) + n^2(n+2m) + n(n+2m)^2) \\
&= O(n(n+m)^2).
\end{aligned}$$

Then,

$$O(\text{Step 2}) = O(n^2\varphi_g + n^2n_g^w + m^2(n_{g,k-1}^v + n_{g,k}^v) + (n+m)^3).$$

Step 3

$$\begin{aligned}
O(\hat{x}_k) &= O(n^3 + n^2p + n^2m), \\
O(\hat{G}_k^x) &= O(n^2(n+m+\varphi_g + n_g^w) + nmn_{g,k-1}^v + nmn_{g,k}^v).
\end{aligned}$$

Then,

$$O(\text{Step 3}) = O(n^2(n+p+m+\varphi_g + n_g^w) + nm(n_{g,k-1}^v + n_{g,k}^v)).$$

All Steps

$$\begin{aligned}
O(\text{DTZF}) &= O(n_g(n + \log(n_g)) + n^2\varphi_g + n^2n_g^w + m^2(n_{g,k-1}^v + n_{g,k}^v) + (n+m)^3 \\
&\quad + n^2(n+p+m+\varphi_g + n_g^w) + nm(n_{g,k-1}^v + n_{g,k}^v)) \\
&= O(n_g(n + \log(n_g)) + (n+m)^3 + n^2(p+\varphi_g + n_g^w) \\
&\quad + (nm + m^2)(n_{g,k-1}^v + n_{g,k}^v)).
\end{aligned}$$

H_∞ZF (Algorithm B.4)

Step 2

Offline, then $O(\text{Input}) = O(1)$.

Step 3

$$\begin{aligned}
O(\tilde{A}\Pi_0^x) &= O(n^2n_g), \\
O(TG^w) &= O(n^2n_g^w), \\
O(-KG^v) &= O(nmn_g^v),
\end{aligned}$$

$$O(\text{rs}(\Pi_1^w)) = O(nn_g^w).$$

Then,

$$O(\text{Step 3}) = O(n^2n_g + n^2n_g^w + nmn_g^v).$$

Step 5

$$\begin{aligned} O(\tilde{A}\Pi_1^x) &= O(n^2n_g), \\ O(\tilde{A}\Pi_1^w) &= O(n^2n_g^w), \\ O(\tilde{A}\Pi_1^{va}) &= O(n^2n_g^v), \\ O(-(\tilde{A}N + K)G^v) &= O(n^2m + nmn_g^v), \\ O(\text{rs}(\Pi_2^w) + \zeta_1^w) &= O(nn_g^w), \\ O(\text{rs}(\Pi_2^{vc}) + \zeta_1^{vc}) &= O(nn_g^v). \end{aligned}$$

Then,

$$O(\text{Step 5}) = O(n^2n_g + n^2n_g^w + n^2n_g^v + n^2m + nmn_g^v).$$

Step 7

$$\begin{aligned} O(\tilde{A}\Pi_{k-1}^x) &= O(n^2n_g), \\ O(\tilde{A}\Pi_{k-1}^w) &= O(n^2n_g^w), \\ O(\tilde{A}\Pi_{k-1}^{va}) &= O(n^2n_g^v), \\ O(\tilde{A}\Pi_{k-1}^{vc}) &= O(n^2n_g^v), \\ O(\text{rs}(\Pi_k^w) + \zeta_{k-1}^w) &= O(nn_g^w), \\ O(\text{rs}(\Pi_k^{vc}) + \zeta_{k-1}^{vc}) &= O(nn_g^v). \end{aligned}$$

Then,

$$O(\text{Step 7}) = O(n^2n_g + n^2n_g^w + n^2n_g^v).$$

Step 9

$$O(\text{rs}(\Pi_k^x)) = O(nn_g),$$

$$\begin{aligned} O(\text{rs}(\Pi_k^{\text{va}})) &= O(nn_g^{\text{v}}), \\ O(\text{summation}) &= O(n). \end{aligned}$$

Then,

$$O(\text{Step 9}) = O(nn_g + nn_g^{\text{v}}).$$

Steps 10 and 11

$$\begin{aligned} O(\hat{x}_k) &= O(n^3 + n^2p + nm), \\ O(\text{summation}) &= O(n). \end{aligned}$$

Then,

$$O(\text{Steps 10–11}) = O(n^3 + n^2p + nm).$$

All Steps

$$\begin{aligned} O(H_{\infty}\text{ZF}) &= O(n^3 + n^2p + n^2n_g + n^2n_g^{\text{w}} + n^2n_g^{\text{v}} + n^2m + nmn_g^{\text{v}}) \\ &= O(n^2(n + p + n_g + n_g^{\text{w}} + n_g^{\text{v}} + m) + nmn_g^{\text{v}}). \end{aligned}$$

ZMV (Algorithm B.13)

$$\begin{aligned} O(\text{Step 1}) &= O(n^2n_g), \\ O(\text{Step 2}) &= O(1), \\ O(\text{Step 4}) &= O(1), \\ O(\text{Step 5}) &= O(n(2n + n_g + i - 1)), \\ O(\text{Steps 6–9}) &= O(1), \\ O(\text{Step 10}) &= O(n^2(2n + n_g + i - 1)), \\ O(\text{Steps 3–11}) &= O(mn^2(n + n_g + m)), \\ O(\text{Step 12}) &= O(1), \\ O(\text{Step 13}) &= O((2n + n_g + m)(n + \log(2n + n_g + m))) \\ &= O((n + n_g + m)(n + \log(n + n_g + m))). \end{aligned}$$

Then,

$$O(\text{ZMV}) = O\left((n + n_g + m)(mn^2 + \log(n + n_g + m))\right).$$

IZMV (Algorithm 3.6)

$$\begin{aligned} O(\text{Step 1}) &= O\left((n + n_g + m)(mn^2 + \log(n + n_g + m))\right), \\ O(\text{Steps 2–3}) &= O(1), \\ O(\text{Steps 5–8}) &= O(1), \\ O(\text{Step 9}) &= O\left(n^2(\varphi_g + i - 1)\right), \\ O(\text{Steps 4–10}) &= O\left(n^3(\varphi_g + m)\right), \\ O(\text{Step 11}) &= O(1). \end{aligned}$$

Then,

$$O(\text{IZMV}) = O\left((n + n_g + m)(mn^2 + \log(n + n_g + m)) + n^3\varphi_g\right).$$

ZEKP (Algorithm B.6)

Step 1

$$O(\text{ZR}) = O\left(n_g(n + \log(n_g))\right).$$

Step 2

$$O(\text{Step 2}) = O(1).$$

Step 3

$$\begin{aligned} O(\hat{x}_{k|k-1}) &= O(1), \\ O(\hat{y}_{k-1}) &= O(1), \\ O(\hat{C}_{k-1}^{\text{xx}}) &= O(n^2\varphi_g), \\ O(\hat{C}_{k-1}^{\text{xy}}) &= O(n^3 + n^2m), \\ O(C_{k-1}^{\text{vv}}) &= O(m^2n_g^v), \\ O(\hat{C}_{k-1}^{\text{yy}}) &= O(n^2m + nm^2). \end{aligned}$$

Then,

$$O(\text{Step 3}) = O\left(n^2(\varphi_g + n + m) + m^2(n + n_g^v)\right).$$

Step 4

$$\begin{aligned} O(\check{K}_{k-1}) &= O(nm^2 + m^3), \\ O(\hat{x}_k) &= O(nm), \\ O(\hat{G}_k^x) &= O\left(n^2(m + \varphi_g)\right). \end{aligned}$$

Then,

$$O(\text{Step 4}) = O\left(n^2(m + \varphi_g + n_g^v) + m^2(n + m)\right).$$

All Steps

$$O(\text{ZEKP}) = O\left(n_g(n + \log(n_g)) + n^2(n + m + \varphi_g) + m^2(n + m + n_g^v) + nmn_g^v\right).$$

CZLF (Algorithm B.5)

Step 1

$$\begin{aligned} O(\hat{X}_{k|k-1}) &= O(n^2n_g + np), \\ O(\hat{Y}_{k|k-1}) &= O(mn_g^v), \\ O(\check{X}_k) &= O(mn(n_g + n_g^w)). \end{aligned}$$

Then,

$$O(\text{Step 1}) = O\left(n^2n_g + np + mn_g^v + nm(n_g + n_g^w)\right).$$

Step 2

$$O(\text{Step 2}) = O\left(k_c(\tilde{n}_h + \tilde{n}_g)^3 + k_cn\tilde{n}_g^2 + (n + \varphi_c)^2(\tilde{n}_g - k_c) + k_g(\tilde{n}_g - k_c)(n + \varphi_c)\right),$$

where $\tilde{n}_h = (n_h + n_h^w + n_h^v + m)$, $\tilde{n}_g = (n_g + n_g^w + n_g^v)$, $k_c = \tilde{n}_h - \varphi_c$, and $k_g = \tilde{n}_g - k_c - \varphi_g$.

All Steps

$$\begin{aligned} O(\text{CZLF}) &= O\left(n^2 n_g + np + mn_g^v + nm(n_g + n_g^w)\right) \\ &\quad + k_c \left(\tilde{n}_h + \tilde{n}_g\right)^3 + k_c n \tilde{n}_g^2 + (n + \varphi_c)^2 (\tilde{n}_g - k_c) + k_g (\tilde{n}_g - k_c) (n + \varphi_c). \end{aligned}$$

where $\tilde{n}_h = (n_h + n_h^w + n_h^v + m)$, $\tilde{n}_g = (n_g + n_g^w + n_g^v)$, $k_c = \tilde{n}_h - \varphi_c$, and $k_g = \tilde{n}_g - k_c - \varphi_g$.

CZDC (Algorithm 3.8)**Forecast Step**

Let $\mathcal{X} = \{G^x, c^x, A^x, b^x\} \subset \mathbb{R}^n$, $u \in \mathbb{R}^p$, and $\mathcal{W} = \{G^w, c^w, A^w, b^w\} \subset \mathbb{R}^q$, where $G^x \in \mathbb{R}^{n \times n_g}$, $c^x \in \mathbb{R}^n$, $A^x \in \mathbb{R}^{n_h \times n_g}$, $b^x \in \mathbb{R}^{n_h}$, $G^w \in \mathbb{R}^{q \times n_g^w}$, $c^w \in \mathbb{R}^q$, $A^w \in \mathbb{R}^{n_h^w \times n_g^w}$, and $b^w \in \mathbb{R}^{n_h^w}$. Let $\mathcal{Z} = \mathcal{X} \times \mathcal{W} = \{G^z, c^z, A^z, b^z\} \subset \mathbb{R}^{\tilde{n}}$, where $G^z \in \mathbb{R}^{\tilde{n} \times \tilde{n}_g}$, $c^z \in \mathbb{R}^{\tilde{n}}$, $A^z \in \mathbb{R}^{\tilde{n}_h \times \tilde{n}_g}$, $b^z \in \mathbb{R}^{\tilde{n}_h}$, $\tilde{n} = n + q$, $\tilde{n}_h = n_h + n_h^w$, and $\tilde{n}_g = n_g + n_g^w$.

$$\begin{aligned} O(\mathcal{Z}) &= O(1), \\ O(\mathcal{T}) &= O\left(\tilde{n}_h (\tilde{n}_h + \tilde{n}_g)^3 + \tilde{n} (\tilde{n} + \tilde{n}_g) (\tilde{n} + \tilde{n}_h + \tilde{n}_g)^3\right), \\ O(\text{vert}) &= \tilde{n}^2 2^{\tilde{n}}, \\ O(\text{DC programming}) &= O(n 2^{\tilde{n}}), \\ O(\mathcal{X}_{k|k-1}) &= O\left(n \tilde{n} (n_g + n_g^w)\right). \end{aligned}$$

Then,

$$O(\text{Forecast}) = O\left(\tilde{n}_h (\tilde{n}_h + \tilde{n}_g)^3 + \tilde{n} (\tilde{n} + \tilde{n}_g) (\tilde{n} + \tilde{n}_h + \tilde{n}_g)^3 + \tilde{n}^2 2^{\tilde{n}}\right).$$

Data-assimilation Step

Let $\mathcal{X} = \{G^x, c^x, A^x, b^x\} \subset \mathbb{R}^n$, $y \in \mathbb{R}^m$, and $\mathcal{V} = \{G^v, c^v, A^v, b^v\} \subset \mathbb{R}^r$, where $G^x \in \mathbb{R}^{n \times n_g}$, $c^x \in \mathbb{R}^n$, $A^x \in \mathbb{R}^{n_h \times n_g}$, $b^x \in \mathbb{R}^{n_h}$, $G^v \in \mathbb{R}^{r \times n_g^v}$, $c^v \in \mathbb{R}^r$, $A^v \in \mathbb{R}^{n_h^v \times n_g^v}$, and $b^v \in \mathbb{R}^{n_h^v}$. Let $\mathcal{Z} = \mathcal{X} \times \mathcal{V} = \{G^z, c^z, A^z, b^z\} \subset \mathbb{R}^{\tilde{n}}$, where $G^z \in \mathbb{R}^{\tilde{n} \times \tilde{n}_g}$, $c^z \in \mathbb{R}^{\tilde{n}}$, $A^z \in \mathbb{R}^{\tilde{n}_h \times \tilde{n}_g}$, $b^z \in \mathbb{R}^{\tilde{n}_h}$, $\tilde{n} = n + r$, $\tilde{n}_h = n_h + n_h^v$, and $\tilde{n}_g = n_g + n_g^v$.

$$\begin{aligned} O(\mathcal{Z}) &= O(1), \\ O(\mathcal{T}) &= O\left(\tilde{n}_h (\tilde{n}_h + \tilde{n}_g)^3 + \tilde{n} (\tilde{n} + \tilde{n}_g) (\tilde{n} + \tilde{n}_h + \tilde{n}_g)^3\right), \\ O(\text{vert}) &= \tilde{n}^2 2^{\tilde{n}}, \\ O(\text{DC programming}) &= O(m 2^{\tilde{n}}), \\ O(\mathcal{Y}) &= O\left(m \tilde{n} + m r n_g^v + m^2\right) \\ O(\mathcal{X} \cap_{H^x} \mathcal{Y}) &= O\left(m n n_g + m (n_g^v + m)\right). \end{aligned}$$

Then,

$$O(\text{Update}) = O\left(\tilde{n}_h(\tilde{n}_h + \tilde{n}_g)^3 + \tilde{n}(\tilde{n} + \tilde{n}_g)(\tilde{n} + \tilde{n}_h + \tilde{n}_g)^3 + (m + \tilde{n}^2)2^{\tilde{n}} + m\tilde{n} + mrn_g^v + mnn_g + m^2\right).$$

Feasibility Step

Let $\mathcal{X} = \{G^x, c^x, A^x, b^x\} \subset \mathbb{R}^n$ and $\mathcal{X}^F = \{G^{xF}, c^{xF}, A^{xF}, b^{xF}\} \subset \mathbb{R}^n$, where $G^x \in \mathbb{R}^{n \times n_g}$, $c^x \in \mathbb{R}^n$, $A^x \in \mathbb{R}^{n_h \times n_g}$, $b^x \in \mathbb{R}^{n_h}$, $G^{xF} \in \mathbb{R}^{n \times n_g^{xF}}$, $c^{xF} \in \mathbb{R}^n$, $A^{xF} \in \mathbb{R}^{n_h^{xF} \times n_g^{xF}}$, and $b^{xF} \in \mathbb{R}^{n_h^{xF}}$. Then,

$$O(\mathcal{X} \cap_{\text{In}} \mathcal{X}^F) = O(n^2 n_g + nn_g^{xF}).$$

Consistency Step

Let $\mathcal{X} = \{G^x, c^x, A^x, b^x\} \subset \mathbb{R}^n$, where $G^x \in \mathbb{R}^{n \times n_g}$, $c^x \in \mathbb{R}^n$, $A^x \in \mathbb{R}^{n_h \times n_g}$, and $b^x \in \mathbb{R}^{n_h}$.

$$\begin{aligned} O(\mathcal{T}) &= O\left(n_h(n_h + n_g)^3 + n(n + n_g)(n + n_h + n_g)^3\right), \\ O(\text{vert}) &= n^2 2^n, \\ O(\text{DC programming}) &= O(m_c 2^n), \\ O(\mathcal{C}) &= O(m_c n + m_c^2) \\ O(\mathcal{X} \cap_H \mathcal{C}) &= O(m_c n n_g + m_c^2). \end{aligned}$$

Then,

$$O(\text{Consistency}) = O\left(n_h(n_h + n_g)^3 + n(n + n_g)(n + n_h + n_g)^3 + m_c^2 + m_c n n_g + (m_c + n^2)2^n\right).$$

Order-Reduction Step

Next, the orders of output CZs are made explicit:

- $\mathcal{X}_{k|k-1}$:

$$\begin{aligned} \tilde{n}_h &= n_h + n_h^w \\ \tilde{n}_g &= n_g + n_g^w + n \end{aligned}$$

- $\check{\mathcal{X}}_k$:

$$\begin{aligned} \tilde{n}_h &= n_h + n_h^w + n_h^v + m \\ \tilde{n}_g &= n_g + n_g^w + n + n_g^v + m \end{aligned}$$

- $\check{\mathcal{X}}_k$:

$$\tilde{n}_h = n_h + n_h^w + n_h^v + m + n_h^{xF} + n$$

$$\tilde{n}_g = n_g + n_g^w + n + n_g^v + m + n_g^{x^F}$$

• $\tilde{\mathcal{X}}_k$:

$$\begin{aligned}\tilde{n}_h &= n_h + n_h^w + n_h^v + m + n_h^{x^F} + n + m_c \\ \tilde{n}_g &= n_g + n_g^w + n + n_g^v + m + n_g^{x^F} + m_c\end{aligned}$$

• $\mathcal{X}_k$:

$$\begin{aligned}\tilde{n}_h &= \varphi_c \\ \tilde{n}_g &= \varphi_g\end{aligned}$$

Then,

$$O(\mathcal{X}^\downarrow) = O\left(k_c(\tilde{n}_h + \tilde{n}_g)^3 + k_c n \tilde{n}_g^2\right) + O\left((n + \varphi_c)^2(\tilde{n}_g - k_c) + k_g(\tilde{n}_g - k_c)(n + \varphi_c)\right),$$

where $k_c = (\tilde{n}_h - \varphi_c)$, $k_g = (\tilde{n}_g - k_c - \varphi_g)$, $\tilde{n}_h = (n_h + n_h^w + n_h^v + m + n_h^{x^F} + n + m_c)$, and $\tilde{n}_g = (n_g + n_g^w + n + n_g^v + m + n_g^{x^F} + m_c)$.

E.4 Mixed-Uncertainty Approaches

ZGKP (Algorithm C.1)

Step 1

$$O(\text{ZR}) = O\left(n_g\left(n + \log(n_g)\right)\right).$$

Step 2

$$\begin{aligned}O(\hat{P}_{k-1}^{zg}) &= O(n^2 \varphi_g), \\ O(\hat{R}_{k-1}^{zg}) &= O(m^2 n_g^v), \\ O(\check{K}_{k-1}) &= O(n^3 + m^3 + n^2 m + m^2 n).\end{aligned}$$

Then,

$$O(\text{Step 2}) = O\left(n^3 + m^3 + n^2 \varphi_g + n^2 m + n m^2 + m^2 n_g^v\right).$$

Step 3

$$\begin{aligned}
O(\hat{\mathcal{C}}_k^x) &= O(n^2m + np), \\
O(\hat{\mathcal{G}}_k^x) &= O(n^2m + n^2\varphi_g + nmn_g^v), \\
O(\hat{P}_k^{xx}) &= O(n^2m + n^3 + nm^2).
\end{aligned}$$

Then,

$$O(\text{Step 3}) = O(n^3 + n^2m + nm^2 + np + n^2\varphi_g + nmn_g^v).$$

Step 4

$$O(\mathcal{I}_k^a) = O(n(\varphi_g + n_g^w + n_g^v)).$$

All Steps

$$\begin{aligned}
O(\text{ZGKP}) &= O(n_g(n + \log(n_g)) + n^3 + m^3 + n^2\varphi_g + n^2m + nm^2 + m^2n_g^v + np + nmn_g^v + nn_g^w) \\
&= O(n_g(n + \log(n_g)) + n(p + mn_g^v + n_g^w) + n^2(n + \varphi_g + m) + m^2(m + n + n_g^v)).
\end{aligned}$$

ZGKF (Algorithm 4.1)**Step 1**

$$O(\text{ZR}) = O(n_g(n + \log(n_g))).$$

Step 2

$$\begin{aligned}
O(\hat{P}_{k-1}^{zg}) &= O(n^2\varphi_g), \\
O(Q_{k-1}^{zg}) &= O(n^2n_g^w), \\
O(\hat{P}_{k|k-1}^{zg}) &= O(n^3), \\
O(\hat{R}_k^{zg}) &= O(m^2n_g^v), \\
O(\check{K}_{k-1}) &= O(m^3 + n^2m + m^2n).
\end{aligned}$$

Then,

$$O(\text{Step 2}) = O\left(n^2(n + m + \varphi_g + n_g^w) + m^2(m + n_g^v + n)\right).$$

Step 3

$$\begin{aligned} O(\hat{c}_k^x) &= O(n^2m + n^2 + np), \\ O(\hat{G}_k^x) &= O(n^2m + n^3 + n^2\varphi_g + n^2n_g^w + nmn_g^v), \\ O(\hat{P}_k^{xx}) &= O(n^2m + n^3 + nm^2). \end{aligned}$$

Then,

$$O(\text{Step 3}) = O\left(n^3 + n^2m + nm^2 + np + n^2\varphi_g + n^2n_g^w + nmn_g^v\right).$$

Step 4

$$O(\mathcal{I}_k^\alpha) = O\left(n\left(\varphi_g + n_g^w + n_g^v\right)\right).$$

All Steps

$$O(\text{ZGKF}) = O\left(n_g\left(n + \log(n_g)\right) + n\left(p + mn_g^v\right) + n^2\left(n + m + \varphi_g + n_g^w\right) + m^2\left(m + n + n_g^v\right)\right).$$

EZGKP (Algorithm C.2)

Step 1

$$O(\text{ZR}) = O\left(n_g\left(n + \log(n_g)\right)\right).$$

Step 2

$$\begin{aligned} O(P_{k-1}^{zg}) &= O(n^2\varphi_g), \\ O(R_{k-1}^{zg}) &= O(m^2n_g^v), \\ O(T) &= O(n^2m + m^3 + nm^2), \\ O(N^\times) &= O(n^2m), \\ O(M^\times) &= O(n^3). \end{aligned}$$

Then,

$$O(\text{Step 2}) = O\left(n^2(m + n + \varphi_g) + m^2(n + m + n_g^{\bar{v}})\right).$$

Step 3

$$\begin{aligned} O(\hat{c}_k^x) &= O(n^2 + n + np + n^2m + nm) \\ &= O(np + n^2m), \\ O(\hat{G}_k^x) &= O\left(n^2\varphi_g + n^2m + nmn_g^{\bar{v}} + n_\delta\left(n^2 + nm + n^2\varphi_g + nmn_g^{\bar{v}} + n^2 + n^2\varphi_g + n^2n_g^{\bar{v}}\right)\right) \\ &= O\left(n^2m + n_\delta\left(n^2\varphi_g + nmn_g^{\bar{v}} + n^2n_g^{\bar{v}}\right)\right), \\ O(\hat{P}_k^{xx}) &= O(nm^2 + n^2m + n^3 + n_\delta n^3 + n_\delta^2 n^3) \\ &= O(nm^2 + n^2m + n_\delta^2 n^3). \end{aligned}$$

Then,

$$O(\text{Step 3}) = O\left(np + n^2m + nm^2 + n_\delta\left(n_\delta n^3 + n^2\varphi_g + nmn_g^{\bar{v}} + n^2n_g^{\bar{v}}\right)\right).$$

Step 4

The number of generators of $\hat{G}_k^x$ is $\tilde{n}_g = \varphi_g + n_g^{\bar{w}} + n_g^{\bar{v}} + n_\delta(1 + \varphi_g + n_g^{\bar{v}})$. Then,

$$\begin{aligned} O(\mathcal{I}_k^\alpha) &= O\left(n + n\left(\varphi_g + n_g^{\bar{w}} + n_g^{\bar{v}} + n_\delta(1 + \varphi_g + n_g^{\bar{v}})\right)\right) \\ &= O\left(nn_g^{\bar{w}} + nn_\delta(\varphi_g + n_g^{\bar{v}})\right). \end{aligned}$$

All Steps

$$O(\text{EZGKP}) = O\left(n_g \log(n_g) + n\left(n_g + p + n_g^{\bar{w}} + mn_\delta n_g^{\bar{v}}\right) + n^2\left(m + n_\delta\varphi_g + nn_\delta^2 + n_\delta n_g^{\bar{v}}\right) + m^2\left(n + m + n_g^{\bar{v}}\right)\right).$$

EZGKF (Algorithm 5.1)

Step 1

$$O(\text{ZR}) = O\left(n_g\left(n + \log(n_g)\right)\right).$$

Step 2

$$O(\hat{c}_{k|k-1}^x) = O(n^2 + np),$$

$$\begin{aligned}
O(\hat{G}_{k|k-1}^x) &= O(n^2\varphi_g + n_\delta(n^2 + n^2\varphi_g)) \\
&= O(n_\delta n^2\varphi_g), \\
O(\hat{P}_k^{xx}) &= O(n^2 + n^3 + n_\delta n^3 + [(n_\delta - 1)n^3 + (n_\delta - 2)n^3 + \dots + n^3]) \\
&= O(n^2 + n^3 + n_\delta^2 n^3).
\end{aligned}$$

Then,

$$\begin{aligned}
O(\text{Step 2}) &= O(n^2 + np + n_\delta n^2\varphi_g + n_\delta^2 n^3) \\
&= O(np + n_\delta n^2\varphi_g + n_\delta^2 n^3).
\end{aligned}$$

Step 3

$$\begin{aligned}
O(P_{k|k-1}^{zg}) &= O(n^2(n_\delta + n_g^{\bar{w}} + \varphi_g(1 + n_\delta)) + n^2) \\
&= O(n^2(n_g^{\bar{w}} + n_\delta\varphi_g)), \\
O(R_k^{zg}) &= O(m^2 n_g^{\bar{v}}), \\
O(S_{k|k-1}^{zg}) &= O(n^2 m + nm^2 + m^2), \\
O(\check{K}_k) &= O(n^2 m + m^3 + nm^2).
\end{aligned}$$

Then,

$$\begin{aligned}
O(\text{Step 3}) &= O(n^2(n_g^{\bar{w}} + n_\delta\varphi_g) + m^2 n_g^{\bar{v}} + n^2 m + nm^2 + m^3) \\
&= O(n^2(n_g^{\bar{w}} + n_\delta\varphi_g + m) + m^2(n_g^{\bar{v}} + n + m)).
\end{aligned}$$

Step 4

$$\begin{aligned}
O(\hat{C}_k^x) &= O(n^2 m), \\
O(\hat{G}_k^x) &= O(n^2 m + n^2(n_\delta + n_g^{\bar{w}} + n_\delta(1 + \varphi_g)) + nm n_g^{\bar{v}}) \\
&= O(n^2(m + n_g^{\bar{w}} + n_\delta\varphi_g) + nm n_g^{\bar{v}}), \\
O(\hat{P}_k^{xx}) &= O(n^2 m + n^2 + n^3 + nm^2 + n^2 m) \\
&= O(nm^2 + n^2 m + n^3).
\end{aligned}$$

Then,

$$\begin{aligned}
O(\text{Step 4}) &= O(n^2(m + n_g^{\bar{w}} + n_\delta\varphi_g) + nm n_g^{\bar{v}} + nm^2 + n^3) \\
&= O(n^2(n + m + n_g^{\bar{w}} + n_\delta\varphi_g) + nm(m + n_g^{\bar{v}})).
\end{aligned}$$

Step 5

The number of generators of $\hat{G}_k^x$ is $\tilde{n}_g = n_\delta + n_g^{\bar{w}} + n_g^{\bar{v}} + \varphi_g(1 + n_\delta)$. Then,

$$\begin{aligned} O(\mathcal{I}_k^\alpha) &= O\left(n + n\left(n_\delta + n_g^{\bar{w}} + n_g^{\bar{v}} + \varphi_g(1 + n_\delta)\right)\right) \\ &= O\left(n\left(n_g^{\bar{w}} + n_g^{\bar{v}} + n_\delta \varphi_g\right)\right). \end{aligned}$$

All Steps

$$O(\text{EZGKF}) = O\left(n_g \log(n_g) + n\left(n_g + p + mn_g^{\bar{v}}\right) + n^2\left(m + n_\delta \varphi_g + nn_\delta^2 + n_g^{\bar{w}}\right) + m^2\left(n + m + n_g^{\bar{v}}\right)\right).$$

