UNIVERSIDADE FEDERAL DE MINAS GERAIS
Instituto de Ciências Exatas
Programa de Pós-Graduação em Ciência da Computação

Alexander Decker de Sousa

TRANSMISSÃO UBÍQUA SEM-FIO DE ENERGIA PARA MÚLTIPLOS DISPOSITIVOS MÓVEIS

Belo Horizonte
2020

Alexander Decker de Sousa

TRANSMISSÃO UBÍQUA SEM-FIO DE ENERGIA PARA MÚLTIPLOS DISPOSITIVOS MÓVEIS

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação do Instituto de Ciências Exatas da Universidade Federal de Minas Gerais como requisito parcial para a obtenção do grau de Mestre em Ciência da Computação.

Orientador: Luiz Filipe Menezes Vieira

Coorientador: Marcos Augusto Menezes Vieira

Belo Horizonte
2020

Alexander Decker de Sousa

UBIQUITOUS WIRELESS POWER TRANSFER FOR MULTIPLE MOBILE DEVICES

Dissertation presented to the Graduate Program in Computer Science of the Universidade Federal de Minas Gerais in partial fulfillment of the requirements for the degree of Master in Computer Science.

Advisor: Luiz Filipe Menezes Vieira

Co-Advisor: Marcos Augusto Menezes Vieira

Belo Horizonte
2020

Sousa, Alexander Decker de

S725u Ubiquitous wireless power transfer for multiple mobile devices [manuscrito] / Alexander Decker de Sousa. — Belo Horizonte, 2020.
101 f. : il. ; 29cm

Orientador: Luiz Filipe Menezes Vieira
Coorientador: Marcos Augusto Menezes Vieira

Dissertação (mestrado) - Universidade Federal de Minas Gerais, Instituto de Ciências Exatas, Departamento de Ciência da Computação.
Referências: f. 98-101

1. Computação – Teses. 2. Pesquisa operacional - Teses. 3. Sistemas de comunicação sem fio – Teses. 4. Energia - Transmissão – Teses. 5. Problemas NP-completo - Teses. I. Vieira, Luiz Filipe Menezes. II. Vieira, Marcos Augusto Menezes. III. Universidade Federal de Minas Gerais, Instituto de Ciências Exatas, Departamento de Ciência da Computação. IV. Título.

CDU 519.6*22(043)

Ficha catalográfica elaborada pela bibliotecária Irénquer Vismeg Lucas Cruz - CRB 6ª Região nº 819.

UNIVERSIDADE FEDERAL DE MINAS GERAIS
INSTITUTO DE CIÊNCIAS EXATAS
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

FOLHA DE APROVAÇÃO

Ubiquitous Wireless Power Transfer for Multiple Mobile Devices

ALEXANDER DECKER DE SOUSA

Dissertação defendida e aprovada pela banca examinadora constituída pelos Senhores:

Prof. Luiz Filipe Menezes Vieira - Orientador
Departamento de Ciência da Computação - UFMG

Marcos Augusto Menezes Vieira
Prof. Marcos Augusto Menezes Vieira - Coorientador
Departamento de Ciência da Computação - UFMG

Profa. Olga Nikolaevna Goussevskaia
Departamento de Ciência da Computação - UFMG

Thiago F. Noronha
Prof. Thiago Ferreira de Noronha
Departamento de Ciência da Computação - UFMG

Belo Horizonte, 29 de Maio de 2020.

Dedico essa dissertação a toda a minha família. Dedico também aos meus amigos de infância, CEDAF, LECOM e vida. Agradeço especialmente a todos os professores que me guiaram ao longo desses 17 anos de estudo. Um agradecimento especial à Maria Luisa, que além de tudo foi muito paciente em esclarecer minhas incontáveis dúvidas em estatística. Por fim, gostaria de lembrar que enquanto escrevo essa dissertação o mundo passa por uma crise de saúde pública sem precedentes na história recente. Sendo assim, dedico esse trabalho a todos que estão envolvidos de forma a superar esse momento delicado, seja com seu trabalho essencial ou levando o consenso da comunidade científica a sério.

Acknowledgements

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001. The author would also like to thank the research agencies CNPq and FAPEMIG. Finally, the author would like to acknowledge professors Luiz Filipe Menezes Vieira and Marcos Augusto Menezes Vieira for their efforts in advising him in this work and Matheus Akira for the real-life implementation of a system which enables deploying the algorithms proposed here.

“Não há vantagem alguma em viver a vida correndo.”
(Shikamaru Nara)

Resumo

O carregamento de energia dos dispositivos móveis utilizando transferência de energia sem fio via indução de potência ganhou destaque nos últimos anos com a criação das técnicas de formação de feixo utilizando sistemas com múltiplas entradas e múltiplas saídas. Todavia, os trabalhos voltados à otimização destas aplicações são, em geral, focados na transmissão de potência, deixando de lado aspectos do processo de carregamento das baterias e de consumo de energia por parte dos dispositivos. Neste trabalho propomos dois novos problemas computacionais associados ao carregamento de dispositivos sem-fio utilizando transferência de energia sem fio via indução magnética e possibilitando um carregamento ubíquo, onde o usuário não precisa saber como os dispositivos são carregados e nem quando. O *problema do carregamento MIMO em tempo mínimo* consiste em encontrar séries temporais de tensão para cada dispositivo transmissor de forma a finalizar o carregamento de todos os nós o mais rápido possível. O *problema do carregamento MIMO sem inanição*, por sua vez, consiste em determinar as séries temporais de tensão de forma a maximizar o horizonte de tempo dentro do qual todos os dispositivos permanecem ligados. Provamos ambos os problemas como NP-Difíceis. Propomos três algoritmos de programação dinâmica pra resolvê-los em tempo exponencial com respeito ao número de dispositivos e em tempo linear com respeito ao tamanho do horizonte de tempo, sendo portanto exponenciais com respeito ao número de bits utilizados para representar esse horizonte. Propomos ainda três algoritmos gulosos como heurísticas para os problemas e elaboramos um algoritmo para gerar instâncias de teste aleatórias com solução garantida e dificuldade parametrizável. Experimentos indicam que o melhor algoritmo de programação dinâmica dentre os propostos é capaz de encontrar uma solução viável para 97% das "instâncias fáceis", enquanto o melhor guloso proposto é capaz de encontrar uma solução viável para 92% dessas instâncias. Para "instâncias difíceis", por sua vez, conseguem obter uma solução viável em 89% e 74% das vezes, respectivamente.

Keywords: transferência de energia sem-fio; redes de n-portas; np-completude; pesquisa operacional; distribuição de energia sem-fio; carregamento sem-fio de baterias; formação de feixe.

Abstract

The charging of wireless devices using inductive power transfer has gained prominence in recent years with the creation of beam-forming techniques using systems with multiple inputs and multiple outputs. However, previous work aimed at optimizing these applications is, in general, focused on power transmission, leaving aside aspects of the battery charging process and the energy consumed by the devices. In this work, we propose two new computational problems associated with the charging of wireless devices using wireless power transfer via magnetic induction, enabling ubiquitous charging, meaning the user does not need to know how and when the devices are charged. The *Minimum-Time MIMO Charging Problem* consists of finding voltage time-series for each transmitting device to finish charging all nodes as soon as possible. The *No-Starvation MIMO Charging Problem*, in turn, consists of determining the voltage time-series for maximizing the time window in which all devices remain alive. We prove both problems as being NP-Hard. We propose three dynamic-programming algorithms to solve them in exponential time regarding the number of devices and in linear time regarding the duration of the time window – or in exponential time regarding the number of bits required for representing the duration. We also propose three greedy algorithms as heuristics for the problems. We describe an algorithm to generate random test instances with a guaranteed solution and parameterizable difficulty. Experiments indicate that the best proposed dynamic-programming algorithm finds a feasible solution for 97% of the "easy instances", while the best proposed greedy algorithm finds a feasible solution for 92% of these instances. Furthermore, they obtain a feasible solution for 89% and 74% of the "hard instances", respectively.

Keywords: wireless power transfer; n-port networks; np-complete; operational research; wireless energy distribution; wireless charging; beam-forming.

List of Figures

1.1 *Beamforming* in IPT refers to applying different currents for each transmitting coil to control the power distribution between the receivers. The scheme represents the power within each coil (circular elements) and its color indicates intensity. 17

1.2 Power flow in an indoor wireless charging application. 18

1.3 Premature finishing the charging process of an intermediate device can cause the disconnection of other receivers which depend on it as a passive transmitter. 18

2.1 Qi Transmitter charging a cellphone (Source: WPC). 23

2.2 MagMIMO system illustration from Jadidian and Katabi [2014]. 26

2.3 Illustration of the system proposed by Cao et al. [2018]. 27

3.1 Main modules of a generic transmission unity. 31

3.2 Main modules of a receiving device. 31

4.1 Time progression of the charges. 47

4.2 The *Minimum-Time MIMO Charging Problem* aims at finding the fastest path to any valid final charge-vector. 51

4.3 The *No-Starvation MIMO Charging Problem* aims at finding a path to any valid final charge-vector within the last time-slot. 51

5.1 Failure ratio for different noise factors considering 10 and 100 maximum number of iterations, respectively. 58

6.1 Overview of the main data structure employed on both algorithms to represent cloud-based Feasible Futures. 67

6.2 Typical data distribution in the *point cloud*. 68

6.3 Structure of the hash considering MATLAB-related issues. 68

6.4 Time and space usage in the *fly-weight* cloud. 70

6.5 Time and space usage in the *floating-point* cloud. 70

8.1 Beta distribution with $\alpha_{in} = 3$ and $\beta_{in} = 2$ and the corresponding ERL score histogram. 80
8.2 The success ratio for some input parameters with $\alpha_{in} = 3, \beta_{in} = 2$. 81
8.3 Absolute number of successes for some input parameters with $\alpha_{in} = 3, \beta_{in} = 2$. 82
8.4 Execution times for some input parameters with $\alpha_{in} = 3, \beta_{in} = 2$. 83
8.5 Beta distribution with $\alpha_{in} = 5$ and $\beta_{in} = 0.5$ and the corresponding ERL score histogram. 84
8.6 Normalized number of successes for some input parameters with $\alpha_{in} = 5, \beta_{in} = 0.5$. 85
8.7 Execution times for some input parameters with $\alpha_{in} = 5, \beta_{in} = 0.5$. 86
8.8 Success ratios for some input parameters with $\alpha_{in} = 5, \beta_{in} = 0.5$, and only the greedy algorithms. 87
8.9 Normalized charging times. 89
8.10 Comparison of the normalized charging times considering the proposed algorithms and the baselines. 90
8.11 Comparison of the normalized charging times considering only instances for which *MultiSpot* was successful. 91

List of Tables

2.1 Summary of the main WPT approaches [Pudur et al., 2014]. 23

2.2 Comparison between the main related work and this work. 24

3.1 Summary of notations used in this work. 32

3.2 Summary of symbols which meaning keeps the same regardless the section. . . 33

8.1 The considered parameters for each algorithm. 80

8.2 P-values for the *Chi-Squared* independence test between input parameters and the number of successes of each considered algorithm. 82

8.3 P-values for the *Kruscal-Wallis* independence test between input parameters and the execution times of each considered algorithm. 82

8.4 *Pearson* correlation values between input parameters and the execution times of each considered algorithm. 83

8.5 P-values for the *chi-squared* independence test between input parameters and the successes of each considered algorithm. 88

8.6 P-values for the Kruskal-Wallis independence test between input parameters and the normalized charging times of each considered algorithm. 88

Contents

1 Introduction **15**

1.1 Motivation 18

1.2 Objectives 20

1.3 Contributions 20

1.4 Organization 21

2 Literature Review **22**

2.1 Run-Time Optimization of WPT 25

2.2 Charging Optimization 28

2.3 Wireless Energy Distribution 28

3 Preliminary Concepts **30**

3.1 Hardware Summary 30

3.2 Notation 31

3.3 Physical Modeling 31

3.4 Discretization 36

4 Problem Formulations **38**

4.1 Minimum-Time MIMO Charging Problem 38

4.2 No-Starvation MIMO Charging Problem 39

4.3 Time Complexity 40

4.4 The greedy approach for the optimization version 46

4.5 The Exact solution 48

4.6 Graphical Interpretation of the Exact Solution 50

5 Proposed Algorithms **52**

5.1 Overview 52

5.2 Exploration 54

5.3 Exploitation 56

5.4 Algorithms . . . 58

5.4.1 Dynamic-Programming Algorithms . . . 59

5.4.2 Greedy Algorithms . . . 61

5.5 Complexity Analysis . . . 63

5.6 Considerations Regarding Parameter Acquisition . . . 64

6 Implementation Decisions **66**

7 Methodology **71**

7.1 Random Instance Generation . . . 72

7.1.1 Generation of the coupling matrix . . . 74

7.1.2 Generation of the transmitting-impedance matrix . . . 75

7.2 Statistical Analysis Description . . . 76

8 Experimental Results **79**

8.1 No-Starvation MIMO Charging Problem: easy instances . . . 79

8.2 No-Starvation MIMO Charging Problem: hard instances . . . 84

8.3 No-Starvation MIMO Charging Problem: Comparison with baselines . . . 86

8.4 Minimum-Time MIMO Charging Problem . . . 88

8.5 Minimum-Time MIMO Charging Problem: Comparison with baselines . . . 89

9 Conclusions and Future Work **92**

Bibliography **94**

Chapter 1

Introduction

The power supply is one of the most significant challenges when it comes to extending the lifetime of a device in wireless networks. If there is no access to a virtually infinite power source, such as the power grid or a reliable energy harvesting element, the lifetime of devices such as those used in Wireless Sensor Networks is generally limited to the autonomy of their batteries. In difficult-to-reach environments or when there are a large number of devices, the manual replacement or recharging of batteries is often impractical. In this case, a very common approach is simply to treat such devices as disposable, which results in expenses related to hardware replacement and garbage accumulation in the environment.

However, the energy issue is remarkable even when there is the possibility of the user being responsible for replacing or recharging the batteries. In the case of mobile devices, for example, the massive use of network communication and the increasing demand for high processing applications is increasingly limiting battery autonomy, which generally requires to be recharged at least once a day, and opportunistically [Bulut et al., 2018]. Therefore, the user must often carry a tangle of wires and chargers, often one for each device, and risk forgetting to recharge and then run out of batteries.

A relatively old solution to this problem is to recharge the batteries by wireless power transfer (WPT). Indeed, with the advancement of mobile device usage, the WPT market has grown at an annual rate of 60.49% since 2014, introducing many mobile phone models integrated with wireless power transceivers. Research firm *MarketsandMarkets* predicts that the WPT market will reach US$ 17.04 billion by 2020 [MarketsandMarkets, 2017].

The advantages of using wireless power transfer are quite diverse. In addition to the already mentioned ease of battery renewal, we can highlight the reduction of the required cabling. This is especially positive for reducing hardware costs and for improving the mobility of the receiving device. Another notable advantage is the reduction of the receiving device's physical dimensions, as often, especially in small devices, the battery is a major determinant of the final chassis size. Besides that, WPT recharging also favors charger-receiver interoperability. Indeed, it eliminates the need for a male connector to be physically and electrically compatible with the female connector.

However, even developed since the late nineteenth century, the WPT area remained highly limited for a long time. The several approaches in the literature such as radiofrequency-based transfer, laser, microwave, and magnetic induction (MI or IPT, from *inductive power transfer*) always have tradeoffs between range, efficiency, user safety, and maximum transferred power.

In the last few years, nevertheless, some work in WPT focused on magnetic induction and, more precisely, on magnetic resonance, achieving promising results with high-efficiency at medium range [Jadidian and Katabi, 2014][Shi et al., 2015][Cao et al., 2018]. Since classical implementations of IPT were almost always applicable only at very short distances, sometimes requiring physical contact between transmitter and receiver, these results have proved themselves highly relevant. Also, they mitigated other disadvantages of the former IPT implementations, such as the need for precise alignment between the coils and the impossibility of charging more than one device at once. Those works were based on the *beamforming* technique, which uses a MISO (*Multiple Input Single Output*) or a MIMO (*Multiple Input Multiple Output*) setup in a way to control the power distribution between the devices by applying different electric currents in each transmitting coil, as illustrated by Figure 1.1.

In particular, the authors of those three papers indicated the recharging of mobile devices as the main use case. Using such methods, several devices in a user's pocket or body, such as cell phones, smartwatches, and more, might recharge simultaneously while sitting in front of their desk where an appropriate transmitter was previously installed.

In this work, we address two computational problems involving wireless power transfer and, more precisely, wireless charging using IPT. The input parameters of both problems are the temporal series of the circuit parameters as well as information about the behavior of the consumers in the system and the circuit limitations. The decision variables, in turn, are the temporal series of the voltages within each transmitting circuit. Unlike the previous works, whose optimization models aimed at maximizing the transferred power at each moment, our approach optimizes the charging process, following two main objectives. The first one is intuitive and consists of recharging all devices as soon as possible. This objective is addressed by the *Minimum-Time MIMO Charging Problem.*

The other one, in turn, consists of avoiding any device in a determined population from running out of batteries for a determined time-horizon. This objective is addressed by the *No-Starvation MIMO Charging Problem.* It is useful for applications such as the indoor charging of mobile devices, as illustrated in Figure 1.2. For these applications, a

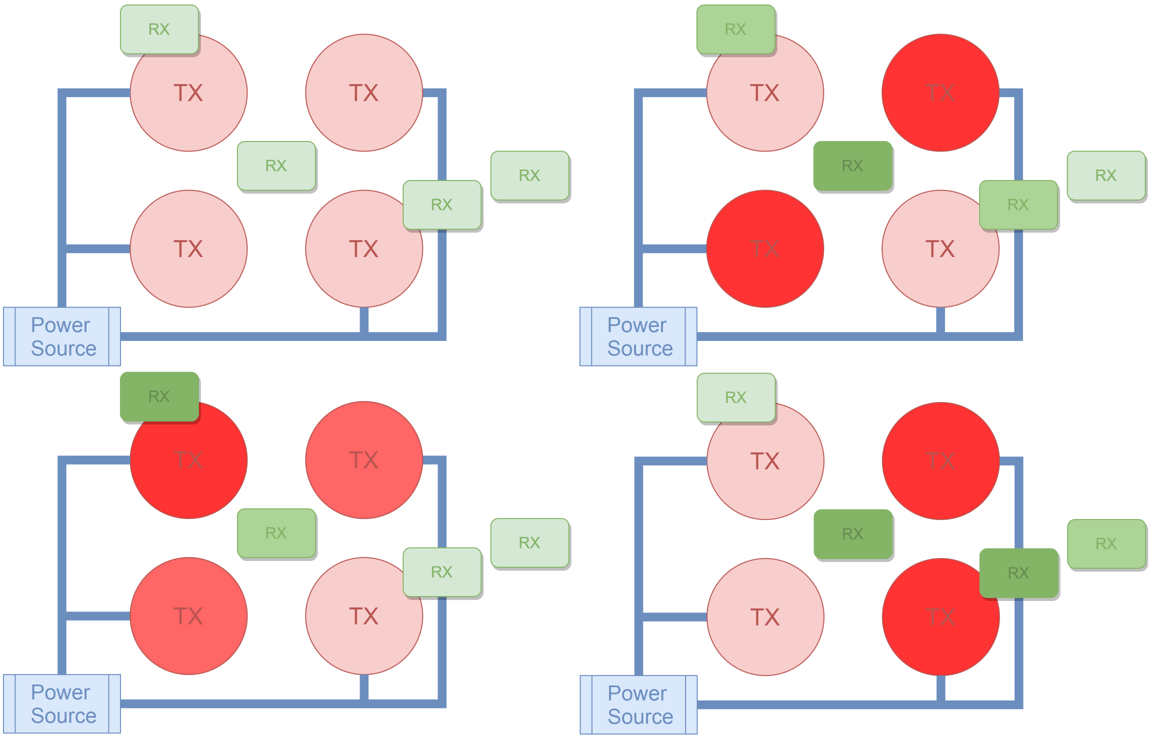

Figure 1.1. *Beamforming* in IPT refers to applying different currents for each transmitting coil to control the power distribution between the receivers. The scheme represents the power within each coil (circular elements) and its color indicates intensity.

set of heterogeneous devices moves through a restricted space together with some transmitting MI devices. The transmitters feature an eternal power supply, although it is limited regarding the electric-current magnitude and the dissipated power. Moreover, each receiving device has an individual IPT receiver, a rechargeable battery, and a power consumer with variable but deterministic behavior.

Let us assume that all devices remain confined into the referred space and that there is a centralized control system aware of all electrical parameters for the entire time-horizon. These parameters include but are not limited to batteries capacity, state-of-charge, resistances, and the mutual induction between each pair of coils at each moment. Let us also abstract the life-cycle of the batteries.

Thus, for the aforementioned application, the batteries do not need necessarily to be charged until they are full. Instead, the power transmitters might simply guarantee that all charges are always above the minimum limit to stay operational. Besides saving energy, this approach improves the connectivity between the devices. Indeed, the load resistances of the receiving devices increases as their charge increases, which prejudices their role as passive signal repeaters, as illustrated by Figure 1.3.

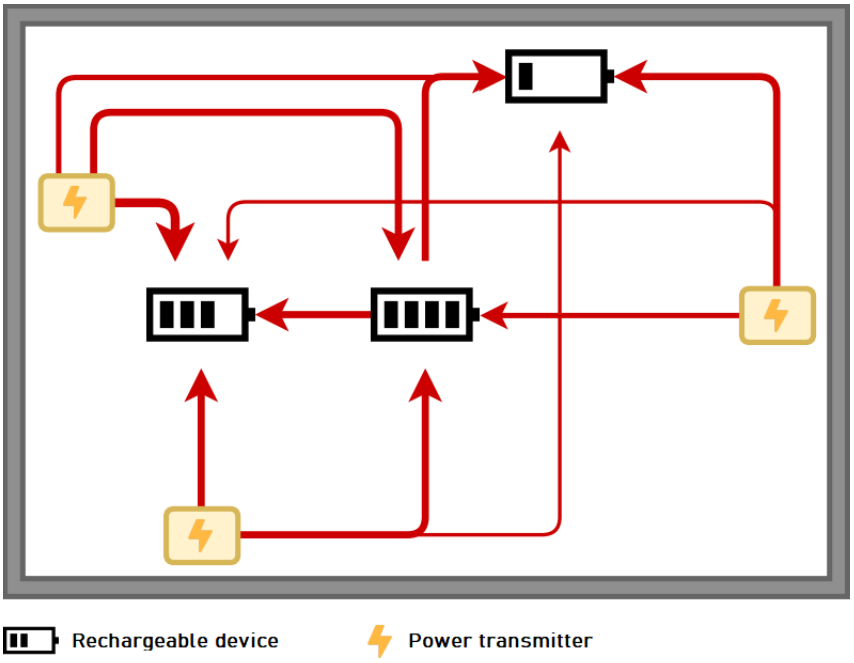

Figure 1.2. Power flow in an indoor wireless charging application.

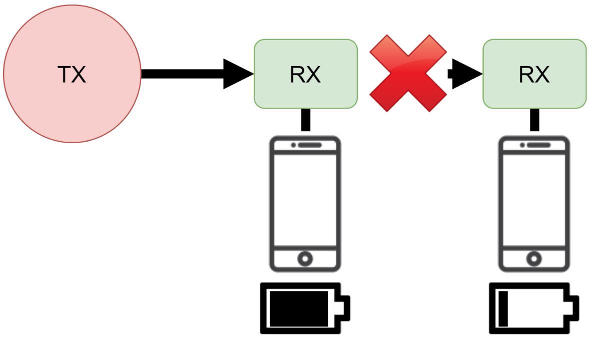

Figure 1.3. Premature finishing the charging process of an intermediate device can cause the disconnection of other receivers which depend on it as a passive transmitter.

1.1 Motivation

The global wireless charging market size was valued at $6,514.2 million in 2018 and is projected to reach $49,304.1 million by 2027 [Patil, 2020]. In particular, nearly 500 million devices with wireless charging shipped in 2017, according to *IHS Markit* [Tech, 2018]. Nevertheless, most wireless charging solutions in the literature do not address the charging process in their modeling, often focusing purely on wireless power transfer. In theory, maximizing the power transfer is equivalent to minimize the charging time and maximize the life-time of the devices around if the system features a single power transmitter. Indeed, maximizing the power applied to the single transmitter maximizes the delivered power. However, the problems became quite different for multiple transmitting devices, especially for heterogeneous networks.

Many of the most promising recent works [Shi et al., 2015; Yang, 2017] involving wireless power transfer consider more than one transmitting device, which is mainly used to improve control over power distribution and transmission range. Thus, the problems addressed by this work fill the gap between the approach employed by the aforementioned works and the charging processes of the powered devices.

The *Minimum-Time MIMO Charging Problem* applies to cases where a MIMO IPT transmitting station must charge a set of nodes as soon as possible. Use cases include the automated charging of wireless sensor nodes [Xie et al., 2012; Peng et al., 2010] and the power distribution among *unmanned aerial vehicles* (UAVs) in *Flying Ad hoc Networks* (FANETS) [Bekmezci et al., 2013]. The automated charging of wireless nodes is often implemented using a wireless transmitting vehicle that runs within the network area and meanwhile provides the power for charging the nodes. Thus, charging the nodes in a certain sub-area as fast as possible is fundamental for the vehicle to attend other sub-areas without disconnections due to energy issues. For the power distribution among FANETs, in turn, the nodes must approach each other for the power transmission to occur, so the main purposes of the network may be temporally prejudiced and, therefore, it is required for the charging process to be agile. Besides that, UAVs face landing inaccuracies that might lead to conditions of poor coupling with the transmitter [Wang et al., 2019]. Thus, a MIMO setup can bring strong tolerance to misalignment and distance.

The *No-Starvation MIMO Charging Problem*, in turn, paves the way for *ubiquitous wireless power charging*. Thus, it allows users of mobile devices to remain oblivious to battery charging, due to the ubiquity of transmitters around and the smart handling of both remaining charges and power transfer.

The *ubiquitous wireless power transfer* has already some glimpses since a few years ago. Huang et al. [2012], for instance, describes a simple power transmitter based on flexible sheets and solar panels, which are simple enough to allow extensive implantation and have its own power supply, avoiding issues towards energy distribution. Assuming that mobile receivers are always close enough to the populations of transmitting devices, the priority is no longer to minimize charging time. Indeed, if the devices are not expected to stay out-of-reach for long periods, the transmitting controller might just manage the available resources in a way to fulfill all energy requirements of the receivers around. However, if a particular device expects to move away from the transmitter population soon, it can simply send a leaving-notification to the controller so that it seeks to ensure a sufficient charge while the device is still reachable.

Besides the applications in low-power mobile devices, the *ubiquitous wireless power* *transfer* paradigm allows the reduction of the dimensions of batteries in electric vehicles through frequent, fast, and opportunistic charging [Ma et al., 2011]. Since vehicle trajectories are somewhat predictable, a controller that manages a population of transmitters along the roads can optimize charging processes with an awareness of device positions over an extended time window. Moreover, the predictability of routes and consequent positions of receivers will increase soon with the popularization of autonomous vehicles.

1.2 Objectives

Our main objective is to propose solutions to enable the *ubiquitous wireless power charging* using *ubiquitous wireless power transfer* and knowledge about the electrical parameters of the system. These parameters can be obtained using sensors, wireless communication, and access to the routes of the receivers. In short, the algorithms proposed in this work shall receive the time-series of all relevant parameters of the system in a given time window and then build a voltage time-series to every transmitting device to keep all devices operational for the considered time-horizon. Thus, the proposed algorithms must solve the *No-Starvation MIMO Charging Problem.* Moreover, we aim at proposing algorithms that can be used with few modifications to generate the voltage time-series to charge all nodes as quickly as possible, thus solving the *Minimum-Time MIMO Charging Problem.*

1.3 Contributions

We define two new computational problems involving the process of charging multiple devices into N-Port network systems [Cederbaum, 1956]. An application example for the proposed problems is the *inductive power transfer systems.* We prove both problems as NP-Hard and demonstrate that the greedy approach followed by the previous literature works may be sub-optimal. We propose a dynamic programming method to solve both problems in exponential time regarding the number of devices and in linear time regarding the duration of the time horizon – or in exponential time regarding the number of bits required to represent the time duration. We describe three different algorithms based on the proposed method and also three other greedy algorithms to serve as heuristics. We define a method to create random instances for the proposed problems which have guaranteed solution and parametrized difficulty. We validate the proposed algorithms using randomly-generated instances and extensive statistic support.

Experiments indicate that for "easy instances" the best dynamic-programming algorithm among those proposed finds a feasible solution for 97% of the instances, while the best greedy algorithm finds it for 92% of them. For "difficult instances", in turn, the rates are 89% and 74%, respectively. The proposed solutions surpass the considered baselines regarding the number of feasible solutions found, although they are equivalent regarding charging-time.

1.4 Organization

After the description of the related literature in Chapter 2, Chapter 3 states the assumptions and mathematical modeling followed by the other chapters. Then, Chapter 4 defines the two main computational problems addressed by this work – the *No-Starvation MIMO Charging Problem* and the *Minimum-Time MIMO Charging Problem*. Next, we prove both problems as NP-Hard and describe a dynamic-programming strategy for the solution of both.

Chapter 5 describes three algorithms that apply the concepts from Chapter 4 to solve the aforementioned problems. Chapter 5 also proposes three greedy algorithms and discusses the asymptotic complexity of the proposed methods. Chapter 6 presents implementation details and comments about the employed data structures. Chapter 7 describes the experimental methodology, the generation of random instances for the problems, and defines the statistical tools to evaluate the algorithms. Chapter 8, in turn, exposes the results considering many scenarios and input parameters. Finally, Chapter 9 presents our conclusions and future work.

Chapter 2

Literature Review

The *Wireless Power Transfer* concept refers to transmitting energy between two or more devices without using a conductor object that directly ties the sources to the destinations. The transmission medium is usually an air gap, although some applications consider underwater environments or even vacuum. Most literature work uses *inductive coupling*, *resonance coupling*, *radio-frequency radiation*, and *laser* for transmitting power, although there are other less popular approaches like *ultrasonic waves* [Pudur et al., 2014]. Table 2.1 summarizes some attributes of the main WPT methods.

The methods based on electromagnetic phenomena can be divided into two groups. The *near-field* WPT uses the near-field region of the electromagnetic field, whose energy decays fast with distance and is in part redirected to the transmitter. They are represented by the *inductive power transfer* methods and feature short-range and high-efficiency. There are two subdivisions for this group. The *inductive coupling* is the *inductive power transfer* method which uses signals with frequencies different than the natural oscillating frequency of the circuits. Thus, despite the high efficiency in semi-contact and the simplicity of implementation, these methods suffer from short-range and heating-effect, since the inductive and capacitive reactances do not cancel each other, thus increasing energy dissipation. The *inductive coupling* methods are widespread in the market, encompassing industry standards such as Qi [WPC, 2008], which is quite popular for the charging of modern mobile devices, as illustrated by Figure 2.1. The *resonant coupling*, in turn, uses signals in resonant-frequency. Therefore, it has medium-ranges and does not suffer from heating-effect.

The *far-field-based* Wireless Power Transfer methods, as known as *radiative* methods, use the far-field region of the electromagnetic field, whose energy decays by the *inverse-square law*. Their representatives include *laser-based techniques* and *radio-frequency-based transfer technologies*. In particular, the *radio-frequency-based technologies* are widely applicable to outdoor environments, including the automated charging of

Table 2.1. Summary of the main WPT approaches [Pudur et al., 2014].

	Inductive Coupling	**Resonant Coupling**	**Radio-Frequency**	**Laser**
Mode of Transfer	Magnetic Inductance	Magnetic Inductance	RF Radiation	Light
Range	Short-range (Omnidirectional)	Mid-Range (Omnidirectional)	Long-range	Long-range (line-of-sight)
Efficiency	High	High	Low	High
Media	Coils	Resonators	Antennas	Laser diodes & Photoelectric Panels
Power	High	High	Low	High

Figure 2.1. Qi Transmitter charging a cellphone (Source: WPC).

wireless nodes [Xie et al., 2012; Peng et al., 2010]. However, there are serious limitations regarding their use for mobile devices and general devices in indoor environments. Indeed, despite there are works like Mohanti et al. [2018] which adapts the power transfer protocol to avoid being harmful to the *802.11ac* communication, the WPT using electromagnetic radiation has a trade-off between transmitted power and damage to human and environment health [Dai et al., 2014]. Therefore, the remaining of this text addresses the main alternative for the *radiative* WPT, that is, the *Inductive Power Transfer*.

The following sections present the main literature works within the three main themes to be explored in this dissertation. These are the *run-time optimization of wireless power transfer*, the *charge optimization*, and the *wireless energy distribution*. Table 2.2

Table 2.2. Comparison between the main related work and this work.

Work	Similarities	Differences
Qi [WPC, 2008]	IPT method	Single transmitter, single receiver
MagMIMO [Jadidian and Katabi, 2014]	*Beam-forming*, IPT method	Single Receiver, disregards the amplitude limitations of each signal
MultiSpot [Shi et al., 2015]	*Beam-forming*, IPT method, MIMO setup	Disregards the amplitude limitations of each signal, disregards the charging process
Yang [2017]	*Beam-forming*, IPT method, MIMO setup, constrained signal amplitude	Disregards the charging process, focus on power transmission
Zhao et al. [2020] and Lin et al. [2019]	Charging optimization, mobile transmitter, multiple receivers	Single transmitter, sub-network scheduling
Madhja et al. [2016], Nikoletseas et al. [2017], and Madhja et al. [2018]	Energy distribution to extend networks' lifetime	Peer-to-peer transmission, simplified charging

summarizes the main similarities and differences between the main related work and this work. The "differences" column refers to attributes of the related work which are the opposite of this work. For instance, the table points out "Single receiver" as a difference of the *MagMIMO* system. Thus, it means *MagMIMO* admits a single receiver and this work admits multiple ones.

2.1 Run-Time Optimization of WPT

The energy transference methods may follow four different setups. The *Single-Input-Single-Output* (SISO) systems have a single transmitting device and a single receiving device. For instance, Wang et al. [2019] propose an asymmetric coupling system suitable for wireless charging of drones using IPT. The system is based on a SISO setup with a ground transmitting base and the receiver deployed in the drone. The ground transmitter is designed with a large-sized primary coil to overcome the lack of transmission power caused by landing error. The *Single-Input-Multiple-Output* (SIMO) setup, in turn, has a single transmitting device and multiple receiving devices.

Systems with more than one IPT transmitting device may follow the *beamforming* approach, which roughly enables the transmitting range and misalignment-tolerance to be improved. *MagMIMO* [Jadidian and Katabi, 2014] is one of the first works that addressed this approach. It follows the *Multiple-Input-Single-Output* (MISO) setup, although its name suggests the employment of the *Multiple-Input-Multiple-Output* (MIMO) setup. As the intensity of the magnetic field decreases very rapidly with distance, the transmitting side is composed of several coplanar coils arranged side by side in a matrix, as shown in Figure 2.2. By controlling the electric current over each transmitting coil, it is possible to compose the resulting magnetic field and increase the flux intensity over a point of interest in space, that is, over the receiving coil. One of the main advantages of this system is the simplicity of the receiver side implementation, which can be just a small *resistor-inductor-capacitor* (RLC) circuit resonating at the same frequency as the transmitting side. In short, the *beamforming* approach consists of a set of techniques based on the composition of a resulting magnetic field using multiple transmitter coils.

The *MagMIMO* algorithm is divided into two steps. The first step calibrates the parameters employed in the optimization model and the second one calculates the optimal transmitting currents and applies the corresponding voltages to the transmitting coils. The power transfer occurs effectively only in the second step, so the total efficiency of the method is consequently decreased by the time employed in the first step. Also, *MagMIMO* simplifies the constraints regarding the maximum active power and omits the constraints towards the maximum current of the devices, which is potentially problematic in realistic applications. Finally, the algorithm requires that parameters such as the mutual inductance between each pair of transmitter coils be pre-parametrized, which reduces the interoperability of the devices. Moreover, it requires manual pre-parametrization each time the transmitting coils are repositioned, which involves several measurements using a multimeter and an oscilloscope.

After the creation of *MagMIMO*, several works followed the *beamforming* strategy. Yang et al. [2016] address the optimization of the current over a single receiver by defining

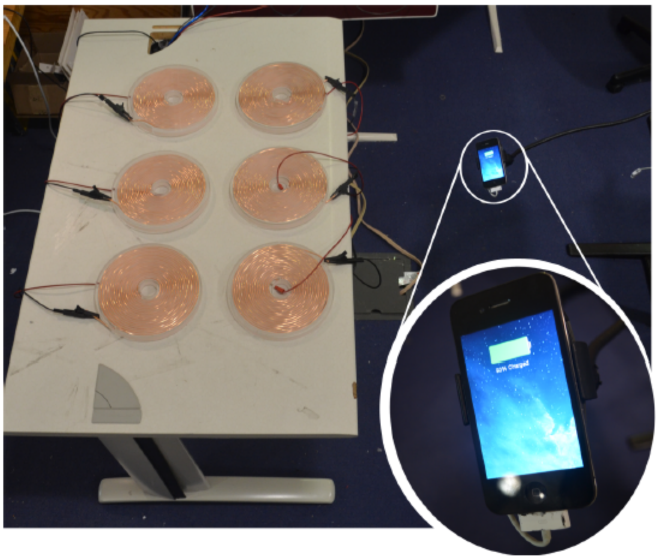

Figure 2.2. MagMIMO system illustration from Jadidian and Katabi [2014].

the currents over multiple resonant transmitters. Yang [2017], in turn, describes more general iterative algorithms for obtaining good results for systems with, in fact, a MIMO setup. Their model includes constraints for the amplitude of the electrical signals, which is very close to the modeling employed in this work.

Some work achieved performance improvements from hardware addends, such as Chang et al. [2017], which enables operational frequency optimization. The system is based on SISO and dynamically adjusts the frequency of the source using a pulse width modulation (PWM) generator. The controller evaluates the gain or loss in voltage in the transmitting coil itself which, by reflection, may indicate whether there was gain or loss in the receiving part. Therefore, it performs an exhaustive search that usually takes some time to reach the optimum and depends on the good behavior of the transferred power landscape. Kisseleff et al. [2015], in turn, proposes the use of orthogonal coils to improve *beamforming* performance.

Some works, in turn, considered software enhancements, such as Kim et al. [2017]. It uses several simplifications to get an extremely simple algorithm for finding a good current configuration in a *beamforming* system. Jiang et al. [2017], in turn, proposes an optimization model very similar to *MagMIMO*, which also uses a data harvesting step to obtain mutual inductance values. Its contribution is to consider a *de facto* MIMO system, that is, with support for multiple receivers, which optimizes the power received through Lagrange multipliers.

The direct evolution of *MagMIMO*, as known as *MultiSpot* [Shi et al., 2015], generalizes its equations to admit multiple receivers simultaneously. In addition to the gains in flexibility, it has also concluded that the presence of multiple receivers in the envi-

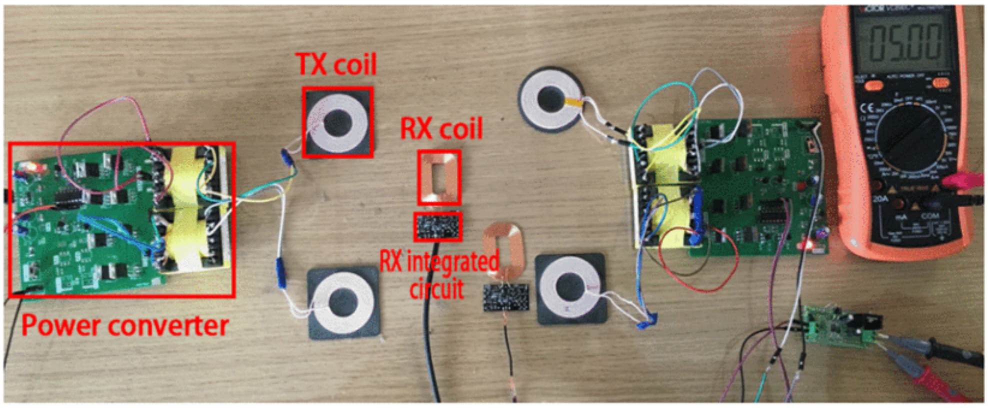

Figure 2.3. Illustration of the system proposed by Cao et al. [2018].

ronment improves range and efficiency results. These improvements occur because each receiver works passively as a signal repeater for the others. However, *MultiSpot* is not concerned with power distribution between receivers. So, it can often create injustices by putting all its efforts into optimizing only a portion of the devices. Also, it requires the pre-parameterization of the mutual inductances between the transmitting pairs, which restricts their positioning freedom. Finally, it does not limit the maximum amplitude of the signals, which may lead to harmful conditions for each transmitter or receiver.

The *MultiSpot* protocol has been extended to Cao et al. [2018] (Figure 2.3), where the distribution of the transmitted power among the receiving devices can be controlled using a weight vector. The algorithm is applicable when there are devices of different kinds and different energy requirements in the same environment. However, the algorithm requires explicit inference of all mutual inductance values, which leads to the same problem of loss of efficiency faced by *MagMIMO*.

This work proposes two new computational problems regarding the prolonging of the battery autonomy of wireless devices using WPT. We demonstrate that the approach of always maximizing the immediately received power is sub-optimal and may lead to very unfavorable scenarios, although the average case has good results. Notice that this greedy approach is employed by all works covered in this section. Indeed, it may interrupt waveguide states and disconnect devices which were once reachable using other receivers as passive signal repeaters. The proposed optimization models consider the time evolution of the charges and other characteristics of the devices, such as current-conversion efficiency, limits for the charge current, discharge current, and more.

2.2 Charging Optimization

As far as we know, all works which address the optimization of the wireless charging process instead of simply optimizing the power transference focus on scheduling wireless nodes in wireless sensor networks. In short, those works address the optimization of the temporal-series of the battery charges by selecting sub-networks to be prioritized at each time-interval. Zhao et al. [2020], for instance, schedules the charging processes along the local sub-networks by controlling the path of a mobile agent with a SIMO wireless transmitter.

Analogously, Lin et al. [2019] aim at maximizing the delivered charge in a SIMO system with a mobile transmitter. They use a drone with a power transceiver to travel between energy-transmitting bases and the rechargeable wireless sensors in a 3D landscape. They prove the problem of scheduling the position of the drone for maximizing the delivered charge with discrete space and time as being NP-hard and provide a greedy heuristic to solve it.

Unlike these works, we consider a MIMO setup, which allows *beamforming* and enables the transmitting-voltages to be used as decision variables. Indeed, maximizing the transferred power with a single transmitter is a polynomial problem and might be solved by the method described in Section 5.2.

2.3 Wireless Energy Distribution

Some works focused on the distribution of energy across populations of energy transceiver devices. Most works abstracted the WPT method and focused on optimizing the scheduling of the devices to be charged. Nikoletseas et al. [2017] showed that the energy balance between a population of power transceiver devices can be obtained by following an opportunistic strategy. The proposed algorithm analyzes each moment when two devices establish a feasible link for the power transfer and dynamically decides if the transfer will be effectively performed or not. The algorithm was validated via probabilistic simulations that assume a constant energy loss for each transfer.

Madhja et al. [2016], in turn, describe a protocol from which the network self-organizes using a star topology in which the central node concentrates half of the available energy. Again, the proposed algorithm was validated via simple probabilistic simulations, although these, different from Nikoletseas et al. [2017], allow variable energy losses according to the link quality. Finally, Madhja et al. [2018] describe distributed algorithms that aim to organize the population into spanning trees where parents have at least twice the charge of their children. Their objective, in short, is to reduce the number of times the users have to manually charge their device. They also obtained their results through simulations, since the authors do not admit limits for the capacity of the batteries and, therefore, the real-world implementation is impracticable.

The *Dynamic Wireless Power Transfer* (DWPT), addressed by Bi et al. [2019], enables *charging-while-driving* and offers opportunities for eliminating range anxiety, stimulating market penetration of electric vehicles (EVs), and enhancing the sustainability performance of electrified transportation. The paper focuses on infrastructure issues and employs genetic algorithms to optimize the DWPT deployment encompassing two dimensions: (i) a spatial dimension, i.e., where to deploy DWPT; and (ii) a temporal dimension, i.e., when to deploy DWPT, considering the EV market boosting.

As a wireless energy distribution technology, this work differs from its predecessors for (i) admitting charge limits within each battery can operate, (ii) considering bases dedicated to power transmission, and (iii) aiming, under ideal conditions, that users do not have to worry about the explicit charging of their devices and, therefore, the power sourcing of their devices be truly ubiquitous.

Chapter 3

Preliminary Concepts

The mathematical modeling employed in this work is widespread along with the state-of-the-art works, such as Jadidian and Katabi [2014], Shi et al. [2015], Cao et al. [2018], and Jung and Lee [2019]. All the aforementioned works use *active/passive Resistor-Inductor-Capacitor* (RLC) rings as building blocks for the IPT systems.

3.1 Hardware Summary

The transmission unity (Figure 3.1) is a set of n_a transmitting devices connected by a single controller. The controller is connected to a transceiver and each transmitting device is composed of a PWM/DC converter, a square-wave converter, and an RLC circuit in series with a digital ammeter. The controller must be able to read the absolute value of the current $\|i_k\|$ of each active RLC circuit and its phase ϕ_k. Thus, the current in *phasor notation* is given by $i_k = \|i_k\| \, cos\phi_k + \sqrt{-1} \cdot \|i_k\| \, sin\phi_k$. The controller also must control the amplitude of the signal generated by each square-wave converter. The angular frequency of all signals is ω.

Each passive device (Figure 3.2) is composed of a passive RLC circuit in series with a digital ammeter. This ammeter is analogous to the ones in the transmitting part and thus its value can be read by the controller in *phasor form* during the runtime. Unlike the transmitting part, each passive device has its own controller and transceiver, which can also be used by the powered device. The device can be a WSN mote, a cellphone, or even a car. The passive device is also composed of an AC/DC converter, a battery controller, and a rechargeable battery.

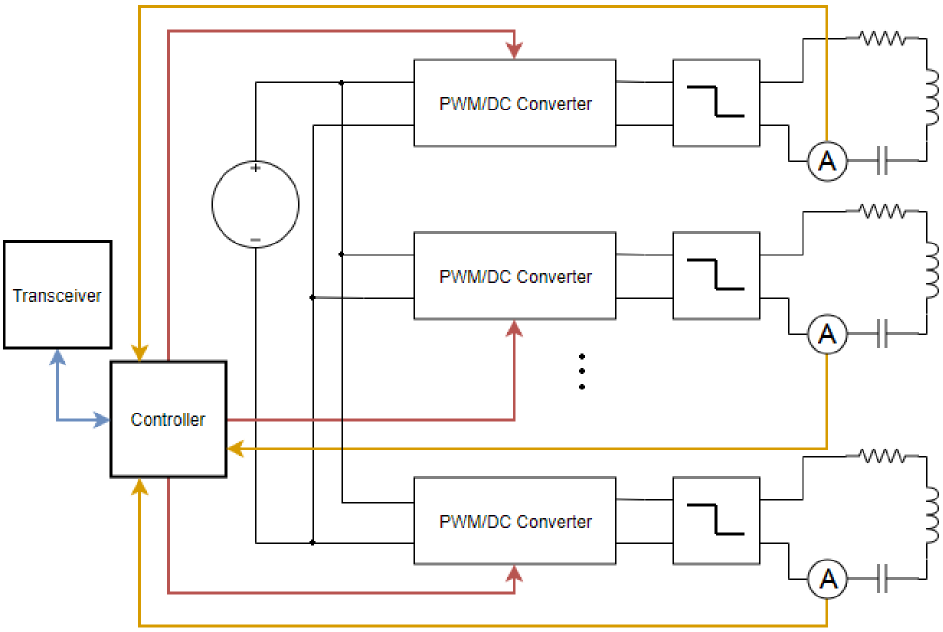

Figure 3.1. Main modules of a generic transmission unity.

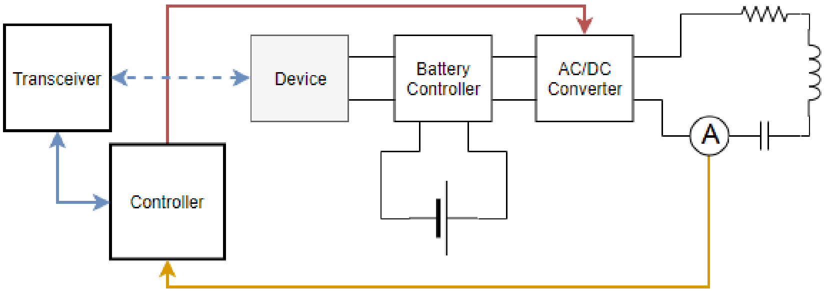

Figure 3.2. Main modules of a receiving device.

3.2 Notation

The notations used in this work are summarized in Table 3.1. The character □ is employed as a wildcard.

3.3 Physical Modeling

In terms of physical modeling, each set of PWM/DC converter and square-wave converter can be approximated by an individual sinusoidal-wave generator [Chen et al., 2014], which voltage is set by the controller in runtime. Thus, each transmitting device can be approximated by an independent active AC RLC circuit. It is important to emphasize that the total power consumed by the system cannot exceed the maximum supported by the

Table 3.1. Summary of notations used in this work.

Notation	***Meaning***
$\square^+$	the *Moore-Penrose* matrix inverse of $\square$
I	the identity matrix. Its size is determined by the context
$\vec{1}$	A vector in which every entry is equal to 1. Its size is determined by the context
$\vec{0}$	A retracted form for $0 \cdot \vec{1}$
$\vec{\square}$	A column vector
$\square_k$	The k-th element of $\vec{\square}$
$\vec{\square}^a$	The sub-vector of $\vec{\square}$ which refers to the active circuits
$\vec{\square}^p$	The same for passive circuits. In short, $\vec{\square} = \left[(\vec{\square}^a)^T \ \ (\vec{\square}^p)^T\right]^T$
$\hat{\square}$	The maximum acceptable value of $\square$
$\underline{\square}$	The minimum acceptable value of $\square$
$\Vert\square\Vert$	The element-wise *euclidean norm* of $\square$
$\square \circ \triangle$	The *Hadamard* (element-wise) product between $\square$ and $\triangle$
$\square \oslash \triangle$	The *Hadamard* (element-wise) division between $\square$ and $\triangle$
$diag(\vec{\square})$	The diagonal matrix whose main diagonal is equal to $\vec{\square}$
$Re(\square)$	Real part of $\square$
$Im(\square)$	Imaginary part of $\square$ without the imaginary unity. $(Im(\square) = -\sqrt{-1}(\square - Re(\square)))$
$max\{\vec{\square}; \vec{\triangle}\}$	Vector where the k-th position is $max\{\square_k; \triangle_k\}$
$min\{\vec{\square}; \vec{\triangle}\}$	Vector where the k-th position is $min\{\square_k; \triangle_k\}$
$\square^{-1}()$	Inverse function
$\vec{\square}()$	Vector function
$\square_<, \square_>$	Limits for a interval such that $\square_< \leq \square \leq \square_>$

source.

Let $< \vec{v}(t), \vec{i}(t), Z(t) >$ be the state of a reciprocal n-port WPT system [Monti et al., 2017] using magnetic resonance at the moment t. $\vec{v}(t)$ is a column vector containing the source voltages of each device in *phasor* notation, which can be defined as $\vec{v}(t) \in \mathbb{R}^n$, if we assume that all voltages are always in phase. $\vec{i}(t)$ is a column vector containing the current across each device also in *phasor* notation and $Z(t)$ is the square impedance matrix, in such a way that $\vec{v}(t) = Z(t)\vec{i}(t)$. Assuming $Z(t)$ always invertible, we can also

Table 3.2. Summary of symbols which meaning keeps the same regardless the section.

Symbol	***Meaning***
$\vec{v}(t)$	*Phasor* voltage vector at moment t (AC).
$\vec{i}(t)$	*Phasor* current vector at moment t (AC).
$\vec{i}^d(t)$	Discharge current vector at moment t (DC).
n_a	Number of active devices
n_p	Number of passive devices
Z_T	Transmitting impedance matrix
Z_R	Receiving impedance matrix
M	Coupling matrix between each receiver and each transmitter
ω	Operational angular frequency
$\vec{q}(t)$	Battery-charge vector
$\vec{\mathcal{R}}(\vec{q})$	Equivalent-resistance vector values if the charges are equal to $\vec{q}$
$\vec{C}(\left\Vert\vec{i}^p\right\Vert)$	Direct-current values after the conversion of the receiving currents $\vec{i}^p$
$\Delta t \cdot \vec{\eta}(\vec{j})$	Charge variation if the input DC currents are $\vec{j}$
$\mathcal{Z}(t)$	See Equation 3.7

write the following.

$$\vec{i}(t) = Z(t)^{-1}\vec{v}(t) \tag{3.1}$$

The order of the voltage values and current values inside $\vec{v}$ and $\vec{i}$ may be arbitrary, requiring only that $v_k(t)$ and $i_k(t)$ correspond to the same device for all t. Thus, to simplify the equations, we assume that the first n_a positions of each vector correspond to the transmitting devices and the last n_p correspond to the receivers. Thus, all the n_a *transmitting devices* form the *transmission unity* and each one of the n_p *receiving devices* is a *passive device*, both described in Section 3.1

The source voltages of the receivers are always equal to zero because they are passive from the point of view of WPT and therefore do not have a voltage source. Thus, the voltage values associated with the transmitters are the only directly controllable ones. Table 3.2 summarizes the main variables used for mathematical modeling.

The matrix $Z(t)$ abstracts environment parameters, such as the position and orientation of the coils, and internal parameters, such as the resistance of each circuit and the equivalent resistance of each battery [Chen et al., 2014]. Each parameter varies relatively slow compared to a computer clock rate or even compared to the human perception because they depend mainly on the handling of the coils and the charging of the batteries, which are both slow processes. Thus, one might perform a set of m samples $\left\{ \left[\vec{v}(t_1), \vec{i}(t_1)\right] ; \left[\vec{v}(t_2), \vec{i}(t_2)\right] ; ... ; \left[\vec{v}(t_m), \vec{i}(t_m)\right] \right\}$ with a small-time variation between them and then approximate all the corresponding impedance matrices $Z(t)$ to a single matrix Z. Using that approximation, the states of the m moments are related as in Equation 3.2.

$$[\vec{v}(t_1) \quad \vec{v}(t_2) \quad ... \quad \vec{v}(t_m)] = Z[\vec{i}(t_1) \quad \vec{i}(t_2) \quad ... \quad \vec{i}(t_m)] \tag{3.2}$$

The formulas and meanings for different interpretations of electric power used in this work are from the *IEEE Standard Definitions for the Measurement of Electric Power Quantities Under Sinusoidal, Non-sinusoidal, Balanced, or Unbalanced Conditions* [IEEE, 2010]. Thus, the active power refers to the power effectively dissipated by the system, and its definition for a given moment t is shown in Equation 3.3.

$$\rho_{act}(t) = Re(\vec{i}(t)^* \, \vec{v}(t)) \tag{3.3}$$

All coupled circuits are simple RLC and therefore the $Z(t)$ matrix can be block-wise defined as in Equation 3.4. The matrices $Z_T(t) \in \mathbb{C}^{n_a \times n_a}$ and $Z_R(t) \in \mathbb{C}^{n_p \times n_p}$ carry the impedance of the transmitting and receiving setups, respectively. The real parts of both are non-negative diagonal matrices, while the imaginary parts are symmetric matrices. $M(t) \in \mathbb{C}^{n_p \times n_a}$, in turn, is the coupling matrix between both setups and is defined as $M(t) = -\sqrt{-1}\omega\mathcal{M}$, where $\mathcal{M} \in \mathbb{R}^{n_p \times n_a}$ consists of the mutual inductances between each pair of active/passive circuits.

$$\begin{bmatrix} \vec{v}^a(t) \\ \vec{0} \end{bmatrix} = Z(t) \begin{bmatrix} \vec{i}^a(t) \\ \vec{i}^p(t) \end{bmatrix} = \begin{bmatrix} Z_T(t) & M(t)^T \\ M(t) & Z_R(t) \end{bmatrix} \begin{bmatrix} \vec{i}^a(t) \\ \vec{i}^p(t) \end{bmatrix} \tag{3.4}$$

Applying block-wise multiplication on Equation 3.4, the following is obtained.

$$\begin{cases} \vec{v}^a(t) = Z_T(t) \; \vec{i}^a(t) + M(t)^T \; \vec{i}^p(t) \\ M(t) \; \vec{i}^a(t) = -Z_R(t) \; \vec{i}^p(t) \end{cases} \tag{3.5}$$

Let $\lambda \in \mathbb{C}^{n_a}$. Then the second equality in Equation 3.5 matches one of the following cases.

$$\vec{i}^a(t) = \begin{cases} -M(t)^{-1}\ Z_R(t)\ \vec{i}^p(t), & det(M(t)) \neq 0 \\ -(M(t)^*\ M(t))^{-1}\ M(t)^*\ Z_R(t)\ \vec{i}^p(t), & det(M(t)^*\ M(t)) \neq 0 \\ -M(t)^+\ Z_R(t)\ \vec{i}^p(t) + \left(I - M(t)^+\ M(t)\right)\lambda, & M(t)\,M(t)^+\ Z_R(t)\ \vec{i}^p(t) = Z_R(t)\ \vec{i}^p(t) \end{cases} \tag{3.6}$$

Each passive circuit has a power consumption device that can be approximated to a variable resistor for simplification purposes [Chen et al., 2014]. We assume that each device k has a battery whose charge $q_k(t)$ is the only significant variable for determining the equivalent resistance at moment t. Thus, let us define $\mathcal{R}_k(q_k(t)) : \mathbb{R}_{\geq 0} \to \mathbb{R}$, that is, the *monotonically increasing* function of the equivalent resistance of the k-th device given its charge is equal to $q_k(t)$. Thus, $Z_R(t)$ can be expanded as in Equation 3.7, considering $\mathcal{Z}$ the part of $Z_R(t)$ that is independent regarding $\vec{q}(t)$.

$$Z_R(t) = \mathcal{Z} + diag\left(\vec{\mathcal{R}}(\vec{q}(t))\right) \tag{3.7}$$

The charges of the batteries depend on the *euclidean norm* of the receiving current $\|i_k^p(t)\|$ of the device over time, the discharge current $i_k^d(t)$ over time, and the conversion efficiency. The conversion efficiency is not constant regarding the input current and limits the range of current values for which the charging is effective. For now, let us abstract the function which converts the receiving current into direct-current for the k-th device at moment t using the *monotonically increasing* function $C_k(x) : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$. Furthermore, let us also abstract the function which converts the direct-current into effective charge variation for the k-th device at moment t using the also monotonically increasing function $\eta_k(x) : \mathbb{R} \to \mathbb{R}$.

Thus, the charge of the k-th battery at moment t is given by Equation 3.8.

$$q_k(t) = q_k(0) + \int_0^t \eta_k\left(C_k(\|i_k^p(\tau)\|) - i_k^d(\tau)\right) d\tau \tag{3.8}$$

Equation 3.8 can be approximated by discretizing the time interval into homogeneous slots of Δt units of time. Let us consider the electric current and other time-variant parameters as constants inside each slot. Thus, the charging integral can be approximated using the *composite trapezoidal rule* as follows.

$$\begin{aligned} q_k(t) \approx \Delta t \sum_{\tau=1}^{t/\Delta t - 1} & \left\{\eta_k\left(C_k(\|i_k^p(\Delta t \cdot \tau)\|) - i_k^d(\tau)\right)\right\} \\ & + \tfrac{\Delta t}{2}\left\{\eta_k\left(C_k(\|i_k^p(0)\|) - i_k^d(0)\right) + \eta_k\left(C_k(\|i_k^p(t)\|) - i_k^d(t)\right)\right\} \end{aligned}$$

The formula for the approximated charge variation within a single time-slot of Δt units of time is shown in Equation 3.9. Consider the time-slot being represented by a sequential indexing $\tau \in \mathbb{N}$ where $\tau = 0$ corresponds to the initial state.

$$q_k(\tau) - q_k(\tau - 1) \approx \Delta t \left\{ \eta_k \left(C_k(\|i_k^p(\tau)\|) - i_k^d(\tau) \right) \right\} \tag{3.9}$$

Since both C and η functions are monotonically increasing, they admit eventual sub-domains where the functions are constant. Thus, the inverse functions C^{-1} and η^{-1} are not guaranteed to be possible. Instead, one might define functions $C_<^{-1}$, $C_>^{-1}$, $\eta_<^{-1}$, and $\eta_>^{-1}$ such that

$$C(x) = y \iff C_<^{-1}(y) \leq x \leq C_>^{-1}(y)$$

$$\eta(x) = y \iff \eta_<^{-1}(y) \leq x \leq \eta_>^{-1}(y)$$

Notice that the referred intervals are convex since the functions are monotonically increasing. Besides that, if x does not belong to a constant sub-domain, the limits for the corresponding interval will be coincident.

3.4 Discretization

For practical reasons, the charge of each device is considered to be discrete. Thus, the charge of the k-th device is lower-bounded by $\underline{q}_k$ and upper-bounded by $\hat{q}_k$ with $s - 1$ uniformly-distributed intermediate states. It is notable that in nature the electrical charge is, in fact, a discrete quantity based on the charge of a single electron. However, this elementary charge is small enough to allow its representation as a continuous quantity ($\approx 1.6 \times 10^{-19}$ C). In short, we consider $s \in \mathbb{N} \,|\, s > 1$ as being one of the parameters of the problems and $\underline{q}_k < q_k \leq \hat{q}_k$ as being the feasible charge interval of the k-th device.

Equation 3.10 describes a *surjective and non-injective vector function* which maps each possible charge vector into a natural vector $\vec{d} \in \mathbb{N}^{n_p}$, where $0 \leq d_i < s \; \forall \; i$.

$$\vec{d} = \left\lceil s \cdot (\vec{q} - \underline{\vec{q}}) \oslash (\hat{\vec{q}} - \underline{\vec{q}}) \right\rceil \tag{3.10}$$

Equations 3.11 and 3.12, in turn, provide a lower bound $\vec{q}_<$ and an upper bound $\vec{q}_>$ to the $\vec{q}$ vectors which map into a determined $\vec{d}$. Moreover, $\vec{q}_< < \vec{q} \leq \vec{q}_>$ if and only if $\vec{q}$ maps into the corresponding $\vec{d}$.

$$\vec{q}_< = \underline{\vec{q}} + \frac{1}{s} \cdot \vec{d} \circ (\hat{\vec{q}} - \underline{\vec{q}}) \tag{3.11}$$

$$\vec{q}_{>} = \underline{\vec{q}} + \frac{1}{s} \cdot (\vec{d} + \vec{1}) \circ (\hat{\vec{q}} - \underline{\vec{q}}) \tag{3.12}$$

Thus, vectors $\vec{q}_{<}$ and $\vec{q}_{>}$ determine a *polytope* in $\mathbb{R}^{n_p}$ space, whose center is placed in $\frac{1}{2}(\vec{q}_{<} + \vec{q}_{>})$. **In this work, we admit two charge vectors to be the same if they belong to the same polytope**. For the algorithms described in Chapter 5, the *polytope* might be represented by its center or by the first charge vector found inside its domain.

Chapter 4

Problem Formulations

We propose here two problems that can be directly applied to Wireless Power Charging using magnetic induction. The previous works [Shi et al., 2015; Cao et al., 2018] focused on optimizing only the power transference, which can be done without considering the time component by using some resources like phasors. On the other hand, optimizing the charging process includes several time-dependent parameters and functions, which demands new and more complex modeling. Therefore, the proposed problems consist of determining the time-evolution of the voltages across each transmitting coil to minimize the time for which the charges of all batteries exceed a given threshold (*Minimum-Time MIMO Charging Problem*) or to maximize the time-horizon in which all nodes remain online (*No-Starvation MIMO Charging Problem*). We consider the omniscient versions of the problems, which means that every system parameter (all resistance values, couplings between each pair of coils, etc.) are known at each moment.

4.1 Minimum-Time MIMO Charging Problem

This problem applies to networks where there is no guarantee of the receivers being reachable from the transmitters for a long time, so the best approach is to recharge them as soon as possible. The decision variables for this optimization problem are the functions $v_k(\tau) : \mathbb{R}_{\geq 0} \to \mathbb{R}$, which map the phasor-notation voltage across the k-th transmitting coil to each moment τ. Since we admit that all voltages are in phase, they are represented by real values instead of complex numbers. The optimization version of the problem is stated by Definition 4.1.1.

Definition 4.1.1. *The* ***Optimization Version of the Minimum-Time MIMO Charging Problem*** *consists of finding the voltage-vector time-series $\vec{v}^a(\tau) : \mathbb{R}_{\geq 0} \to \mathbb{R}^{n_a}$ which solves Equation 4.1.*

$$\begin{aligned} \min \quad & \mathrm{t} \\ \text{s.t.} \quad & q_k(t) \geq \underline{Q}_k && \forall\ 1 \leq k \leq n_p \\ & q_k(\tau) \geq \underline{q}_k && \forall\ 0 \leq \tau \leq t,\ 1 \leq k \leq n \\ & \rho_{act}(\tau) \leq \hat{\rho}_{act} && \forall\ 0 \leq \tau \leq t \\ & \|i_k(\tau)\| \leq \hat{i}_k && \forall\ 0 \leq \tau \leq t,\ 1 \leq k \leq n \end{aligned} \tag{4.1}$$

The first set of constraints in Equation 4.1 ensures all charges are over the minimum acceptable threshold after the last time-slot. The charge progression is modeled by Equation 3.9 and depends on the receiving currents, current conversion function, discharge currents, charging efficiency and the charges in the previous time-slot, starting from the initial charges. The second one ensures the required discharge currents are always possible, and the device always has at least the minimal charge to stay operational. The third one limits the active power, which is modeled by Equation 3.3, and the last one limits the maximum amplitude of the electric currents. The voltages and currents are related according to Equation 3.5, following a N-port network model where the resistances and inductances are modeled in a impedance matrix. The charges and the receiving impedance are related according to Equation 3.7. This optimization problem has a decision version, which is summarized in Definition 4.1.2.

Definition 4.1.2. *The* ***Decision Version of the Minimum-Time MIMO Charging Problem*** *consists of deciding if there is a voltage-vector time-series* $\vec{v}^a(\tau) : \mathbb{R}_{\geq 0} \to \mathbb{R}^{n_a}$ *for which the constraints in Equation 4.1 are respected considering* $t \leq \hat{t}$ *for a given time-horizon* $\hat{t} \in \mathbb{R}_{\geq 0}$.

4.2 No-Starvation MIMO Charging Problem

This problem applies to networks in which all nodes are restricted to a space that is well-covered by a set of wireless transmitters. Thus, each node is reachable from the transmitter set at any time. Better than complete the charging processes as soon as possible, one might require that the nodes be operational for the longest possible time. The *No-Starvation MIMO Charging Problem* consists of finding the voltage time-series $\vec{v}^a(t) : \mathbb{R}_{\geq 0} \to \mathbb{R}^{n_a}$ which maximizes the time-horizon t in which all nodes remain alive. This problem formulation is suitable for use cases involving house automation, mobile devices under indoor situations, electrical vehicles, and more. The optimization version of the problem is stated by Definition 4.2.1.

Definition 4.2.1. *The* ***Optimization Version of the No-Starvation MIMO Charging Problem*** *consists of finding a voltage-vector time-series $\vec{v}^a(\tau) : \mathbb{R}_{\geq 0} \to \mathbb{R}^{n_a}$ which solves Equation 4.2.*

$$\begin{aligned} \max \quad & \text{t} \\ \text{s.t.} \quad & q_k(\tau) \geq \underline{q}_k \quad \forall\ 0 \leq \tau \leq t,\ 1 \leq k \leq n \\ & \rho_{act}(\tau) \leq \hat{\rho}_{act} \quad \forall\ 0 \leq \tau \leq t \\ & \|i_k(\tau)\| \leq \hat{i}_k \quad \forall\ 0 \leq \tau \leq t,\ 1 \leq k \leq n \end{aligned} \tag{4.2}$$

A valid voltage time-series is such that, for all time-slots from 1 to t, *(i)* all devices have at least the minimal charges to stay operational, *(ii)* the consumed power never exceeds the given limit according to Equation 3.3, and *(iii)* the amplitude of the currents never exceed the given limit. Analogously to the *Minimum-Time MIMO Charging Problem*, the voltages and currents are related according to Equation 3.5 and the charges and the receiving impedance are related according to Equation 3.7. The decision version of the problem is formally described by Definition 4.2.2.

Definition 4.2.2. *The* ***Decision Version of the No-Starvation MIMO Charging Problem*** *consists of deciding if there is a voltage-vector time-series $\vec{v}^a(\tau) : \mathbb{R}_{\geq 0} \to \mathbb{R}^{n_a}$ which respects all constraints from Equation 4.2 considering a time-horizon of $\hat{t} \in \mathbb{N}$ of time-slots.*

4.3 Time Complexity

This section proves the problems from Definitions 4.1.2 and 4.2.2 as being in NP-Hard. For that, we first prove an auxiliary problem as being in NP-Hard, and then we show that both problems can be used to solve any instance of it by setting some polynomial-size parameters.

Definition 4.3.1. *The well-known NP-Complete* ***0-1 Knapsack Problem*** *can be formulated as follows. Given a real prize vector $\vec{p}$ and a real weight vector $\vec{w}$, decide if there is $\vec{x}$ such that $\vec{p}^t\vec{x} \geq \underline{p}$, $\vec{w}^t\vec{x} \leq \hat{w}$ and $x_k \in \{0, 1\}\ \forall\, k$.*

Definition 4.3.2. *The* ***Auxiliary Problem*** *can be formulated as follows. Let $\vec{d}$ be a real vector and $\mathcal{A}$ be a real matrix with one or two rows, such that the rows are linearly independent. Decide if there is $\vec{z}$ such that $-\vec{d} \leq \mathcal{A}\vec{z} \leq \vec{d}$ and $z_k \in \{-1, 1\}\ \forall\, k$.*

Lemma 4.3.3. *Let $\vec{d}$ be a real vector and $\mathcal{A}$ be a real matrix. The sandwich inequality*

$$-\vec{d} \leq \mathcal{A}\begin{bmatrix}1\\ \vec{y}\end{bmatrix} \leq \vec{d} \;\mid\; y_i \in \{-1,1\}\,\forall\, i$$

has a solution if and only if

$$-\vec{d} \leq \mathcal{A}\vec{z} \leq \vec{d} \;\mid\; z_i \in \{-1,1\}\,\forall\, i$$

is solvable. In other words, the second inequality has a solution if and only if it has a solution where $z_1 = 1$.

Proof. Suppose there is a certain $\vec{y}$ which solves the first inequality. $\vec{z} = \begin{bmatrix}1\\ \vec{y}\end{bmatrix}$ is a trivially valid solution for the second inequality. Now suppose there is a valid $\vec{z}$ which solves the second inequality. If z_1 is 1, $\vec{y} = \{z_i \mid i \geq 2\}$ is trivially a valid solution for the first inequality. Otherwise, $\vec{y} = \{-z_i \mid i \geq 2\}$ is a valid solution for the first inequality, since $-\vec{z}$ is also a valid solution for the second inequality. Indeed, $-\vec{d} \leq \mathcal{A}\vec{z} \leq \vec{d}$ is equivalent to $(-1) - \vec{d} \geq (-1)\mathcal{A}\vec{z} \geq (-1)\vec{d}$ and, consequently, to $-\vec{d} \leq \mathcal{A}((-1)\vec{z}) \leq \vec{d}$. □

Lemma 4.3.4. *The **0-1 Knapsack Problem** (Definition 4.3.1) can be polynomially reduced to the **Auxiliary Problem** (Definition 4.3.2).*

Proof. There are two mutually disjoint cases.

1. $\vec{p}$ **and** $\vec{w}$ **are linearly dependent**. So, there is $K \in \mathbb{R}$ such that $\vec{p} = -K\vec{w}$. If $K \geq 0$, the knapsack formulation $\vec{p}^t\vec{x} \geq \underline{p} \wedge \vec{w}^t\vec{x} \leq \hat{w}$ is equivalent to $\vec{w}^t\vec{x} \leq -\frac{1}{K}\underline{p} \wedge \vec{w}^t\vec{x} \leq \hat{w}$ and, therefore, to

   $$\vec{w}^t\vec{x} \leq min(-\tfrac{1}{K}\underline{p}; \hat{w})$$

   In short, if $K \geq 0$, there is a valid solution if and only if $\vec{w}^t\vec{x}' \leq min(-\frac{1}{K}\underline{p}; \hat{w})$, where $x'_i = 0$ if $w_i > 0$ and $x'_i = 1$ otherwise. This test can be computed in *polynomial-time* regarding the size of $\vec{x}$ and, therefore, the polynomial reduction towards this sub-problem is trivial.

   If $K < 0$, in turn, $\vec{p}^t\vec{x} \geq \underline{p} \wedge \vec{w}^t\vec{x} \leq \hat{w}$ is equivalent to $-\frac{1}{K}\underline{p} \leq \vec{w}^t\vec{x} \leq \hat{w}$. By adding $\frac{1}{2K}\underline{p} - \frac{1}{2}\hat{w}$ to both sides and substituting $\vec{x} = \frac{1}{2}\vec{y} + \frac{1}{2}$, the following inequality is obtained.

   $$-\left(\frac{1}{2}\hat{w} + \frac{1}{2K}\underline{p}\right) \leq \frac{1}{2}\vec{w}^t\vec{y} + \frac{1}{2}\vec{w}^t\vec{1} + \frac{1}{2K}\underline{p} - \frac{1}{2}\hat{w} \leq \left(\frac{1}{2}\hat{w} + \frac{1}{2K}\underline{p}\right)$$

The expression is equivalent to

$$-\left(\frac{1}{2}\hat{w} + \frac{1}{2K}\underline{p}\right) \leq \left[\frac{1}{2}\vec{w}^t\vec{1} + \frac{1}{2K}\underline{p} - \frac{1}{2}\hat{w};\ \frac{1}{2}\vec{w}^t\right]\begin{bmatrix}1\\ \vec{y}\end{bmatrix} \leq \left(\frac{1}{2}\hat{w} + \frac{1}{2K}\underline{p}\right)$$

which can be solved by any algorithm that solves

$$-\vec{d} \leq \mathcal{A}\begin{bmatrix}1\\ \vec{y}\end{bmatrix} \leq \vec{d} \ \mid\ y_i \in \{-1, 1\}\,\forall\, i$$

2. **$\vec{p}$ and $\vec{w}$ are linearly independent**. Let us define

$$A = \begin{bmatrix}-\vec{p}^t\\ \vec{w}^t\end{bmatrix} \qquad \vec{b} = \begin{bmatrix}-\underline{p}\\ \hat{w}\end{bmatrix} \qquad \vec{\beta} = \begin{bmatrix}min_{\vec{x}}\,\{-\vec{p}^t\vec{x}\}\\ min_{\vec{x}}\,\{\vec{w}^t\vec{x}\}\end{bmatrix}$$

Thus, the original inequalities can be rewritten as $\vec{\beta} \leq A\vec{x} \leq \vec{b}$, since $\vec{\beta}$ is a lower bound of $A\vec{x}$. By adding $-\frac{1}{2}(\vec{\beta} + \vec{b})$ to both sides and substituting $\vec{x} = \frac{1}{2}\vec{y} + \frac{1}{2}$, the following inequality is obtained.

$$-\frac{1}{2}(\vec{b} - \vec{\beta}) \leq \left[\frac{1}{2}A\vec{1} - \frac{1}{2}(\vec{\beta} + \vec{b});\ \frac{1}{2}A\right]\begin{bmatrix}1\\ \vec{y}\end{bmatrix} \leq \frac{1}{2}(\vec{b} - \vec{\beta})$$

Analogously to the previous item, it can be solved by any algorithm that solves

$$-\vec{d} \leq \mathcal{A}\begin{bmatrix}1\\ \vec{y}\end{bmatrix} \leq \vec{d} \ \mid\ y_i \in \{-1, 1\}\,\forall\, i$$

According to Lemma 4.3.3, any instance of

$$-\vec{d} \leq \mathcal{A}\begin{bmatrix}1\\ \vec{y}\end{bmatrix} \leq \vec{d} \ \mid\ y_i \in \{-1, 1\}\,\forall\, i$$

can be polynomially reduced to

$$-\vec{d} \leq \mathcal{A}\vec{z} \leq \vec{d} \ \mid\ z_i \in \{-1, 1\}\,\forall\, i$$

The $\mathcal{A}$ matrices in the enumerated cases are either a row vector or a matrix with two linearly independent rows and, therefore, the *0-1 Knapsack Problem* can be polynomially reduced to the *Auxiliary Problem*. □

Lemma 4.3.5. *The **Auxiliary Problem** (Definition 4.3.2) is polynomially reducible to the **Minimum-Time MIMO Charging Problem** (Definition 4.1.2).*

Proof. Let $<\vec{d}, \mathcal{A}>$ be any instance of the *Auxiliary Problem.*

Let the instance of the decision version of the *Minimum-Time MIMO Charging Problem* described below. The load resistances are constant regardless of the charges. Notice that $\vec{\mathcal{R}}(\vec{q})$ being a constant function does not disrespect the requirement of being monotonically increasing. Besides that, the time-horizon is composed of a single time-slot with length equal to $\hat{t}$. Furthermore, the discharge currents are zero and the current conversion functions are such that $\vec{\eta}\left(\vec{C}\left(\left\|\vec{i^p}\right\|\right)\right) = \left\|\vec{i^p}\right\|$. Let the charge thresholds $\underline{\vec{q}}$ be always zero. Therefore, this instance of the problem is equivalent to decide if there is $\vec{v^a}$ such that

$$\begin{cases} q_k(0) + \hat{t} \cdot \|i^p_k\| \geq \underline{Q}_k & \forall\ 1 \leq k \leq n_p \\ \rho_{act} \leq \hat{\rho}_{act} \\ \|i^a_k\| \leq \hat{i}^a_k & \forall\ 1 \leq k \leq n_a \\ \|i^p_k\| \leq \hat{i}^p_k & \forall\ 1 \leq k \leq n_p \end{cases}$$

Let $\hat{\rho}_{act} = +\infty$. The power constraint becomes trivially satisfied and then the system of inequalities is equivalent to

$$\begin{cases} q_k(0) + \hat{t} \cdot \|i^p_k\| \geq \underline{Q}_k & \forall\ 1 \leq k \leq n_p \\ \|i^a_k\| \leq \hat{i}^a_k & \forall\ 1 \leq k \leq n_a \\ \|i^p_k\| \leq \hat{i}^p_k & \forall\ 1 \leq k \leq n_p \end{cases}$$

Let $\underline{Q}_k = q_k(0) + \hat{t}\ \forall\ k$. Let $q_k(0) < \underline{Q}_k\ \forall\ k$. The duration $\hat{t}$ of the time-horizon is always positive, so for the first set of constraints to be satisfied, $\|i^p_k\|$ must be at least 1. If we set $\hat{i}^p_k = 1\ \forall\ k$, in turn, the system is equivalent to Equation 4.3.

$$\begin{cases} \|i^a_k\| \leq \hat{i}^a_k & \forall\ 1 \leq k \leq n_a \\ \|i^p_k\| = 1 & \forall\ 1 \leq k \leq n_p \end{cases} \tag{4.3}$$

Let $n_a = n_p = \frac{1}{2}n$. Let the impedance matrices be defined as $Z_T = T + \sqrt{-1}U$ and $Z_R = \sqrt{-1}S$, where T is an arbitrary real-positive diagonal matrix – which is always invertible –, S is an arbitrary symmetric real invertible matrix with all values in the main diagonal equal to zero and U is such that $U = -Im(M)^t S^{-1} Im(M)$. Notice that U is symmetric since

$$-Im(M)^t S^{-1} Im(M) = -(Im(M)^t S^{-1} Im(M))^t = -Im(M)^t (S^{-1})^t Im(M)$$

and S^{-1} is symmetric.

The $M = \sqrt{-1}\Upsilon^{-1}$ matrix and the values $\hat{i}^a_k$ are constructed as follows.

- If $\mathcal{A}$ has a single row, the first line of Υ is equal to $-\mathcal{A}S^{-1}$ and the other lines are any set of rows in such a way that all lines are linearly independent and, therefore, Υ is invertible. In this case, $\hat{i}^a_1 = d$ and $\hat{i}^a_k = +\infty$, $k \neq 1$.

- If $\mathcal{A}$ has two linearly independent rows, the first two rows of Υ are equal to $-\mathcal{A}S^{-1}$ and the other rows are composed in order to Υ be invertible. This composition is always possible since the first two rows are linearly independent. Indeed, if they were linearly dependent the rows of $\mathcal{A}$ would be linearly dependent, which leads to a contradiction. Analogously to the previous item, $\hat{i}^a_1 = d_1$, $\hat{i}^a_2 = d_2$ and $\hat{i}^a_k = +\infty$, $k > 2$.

M is invertible and, therefore, the first definition of Equation 3.6 can be used to build a bijection which relates $\vec{i^a}$ and $\vec{i^p}$, that is, $\vec{i^a} = -M^{-1}Z_R\vec{i^p}$. For the considered values of M and Z_R, the bijection is equivalent to $\vec{i^a} = -\Upsilon S\vec{i^p}$. Thus, from Equation 3.5, it follows that

$$\vec{v^a} = Z_T\vec{i^a} + M^T\vec{i^p} = (-T\Upsilon S - \sqrt{-1}U\Upsilon S + \sqrt{-1}(\Upsilon^{-1})^T)\vec{i^p}$$

Substituting $U = -Im(M)^t S^{-1} Im(M) = (\Upsilon^{-1})^t S^{-1}\Upsilon^{-1}$, the following is obtained

$$\vec{v^a} = -T\Upsilon S\,\vec{i^p}$$

The function which relates $\vec{v^a}$ to $\vec{i^p}$ is a bijection, since S, Υ, and T are invertible and, therefore, the problem can be restructured in order to $\vec{i^p}$ be the decision variable. Notice that $\vec{v^a}$, T, Υ, and S are real and so is $\vec{i^p}$. Thus, the restrictions $\|i^p_k\| = 1$ are equivalent to $\vec{i^p_k} \in \{-1, 1\}^{n_p}$. Furthermore, $-\Upsilon S$ is also real and so is $\vec{i^a}$. So, the inequality system is equivalent to

$$-\widehat{\vec{i^a}} \leq -\Upsilon S\vec{i^p} \leq \widehat{\vec{i^a}} \quad | \quad \vec{i^p} \in \{-1, 1\}^{n_p}$$

The first one or two rows of Υ are $-\mathcal{A}S^{-1}$, so the first one or two rows of $-\Upsilon S$ are $-(-\mathcal{A}S^{-1})S = \mathcal{A}$. Therefore, the first one or two inequalities are equivalent to

$$-\vec{d} \leq \mathcal{A}\vec{i^p} \leq \vec{d} \quad | \quad \vec{i^p} \in \{-1, 1\}^{n_p}$$

As $det(\Upsilon S) \neq 0$, the linear coefficients of the other inequalities are not all zero. So, the other inequalities are trivially satisfied since the lower bounds are all $-\infty$ and the upper bounds are $+\infty$.

In short, this instance of the *Minimum-Time MIMO Charging Problem* is equivalent to a general instance $< \vec{d}, \mathcal{A} >$ of the *Auxiliary Problem*. The reduction consists of defining a polynomial-size set of parameters and each parameter is polynomial-sized regarding the number of devices in the system. Therefore, the *Auxiliary Problem* is polynomially reducible to the *Minimum-Time MIMO Charging Problem.*

□

Theorem 4.3.6. *The* ***Minimum-Time MIMO Charging Problem*** *(Definition 4.1.2) is in NP-Hard.*

Proof. The problem is NP-Hard because the *0-1 Knapsack Problem* can be polynomially reduced to it. Indeed, the *0-1 Knapsack Problem* can be polynomially reduced to the *Auxiliary Problem* according to Lemma 4.3.4 and the *Auxiliary Problem* can be polynomially reduced to the *Minimum-Time MIMO Charging Problem* according to Lemma 4.3.5. □

Theorem 4.3.7. *The* ***No-Starvation MIMO Charging Problem*** *(Definition 4.2.2) is in NP-Hard.*

Proof. The problem is in *NP-Hard* because the *Auxiliary Problem* can also be polynomially reduced to it in a way similar to the *Minimum-Time MIMO Charging Problem* as follows. Let the instance of the decision version of the *No-Starvation MIMO Charging Problem* where the equivalent resistances are constant regarding the charges, the maximum active power $\hat{\rho}$ is such that $\hat{\rho} = +\infty$ and the time-horizon is equal to a single slot with length $\hat{t}$. Besides that, the current conversion functions are such that $\vec{\eta}\left(\vec{C}\left(\left\|\vec{i}^p\right\|\right) - \vec{i}^d\right) = \left\|\vec{i}^p\right\| - \vec{i}^d$. Thus, the instance is equivalent to deciding if there is $\vec{v}^a$ such that the current amplitudes do not exceed the maximum limit and the charges at the end of the slot allow the devices to stay operational. In short,

$$\begin{cases} q_k(0) + \hat{t} \cdot \left(\|i_k^p\| - i_k^d\right) \geq \underline{q}_k & \forall\ 1 \leq k \leq n_p \\ \|i_k^a\| \leq \hat{i}_k^a & \forall\ 1 \leq k \leq n_a \\ \|i_k^p\| \leq \hat{i}_k^p & \forall\ 1 \leq k \leq n_p \end{cases}$$

After some algebraic manipulations, the system can be rewritten as

$$\begin{cases} \|i_k^p\| \geq i_k^d + \dfrac{\left(\underline{q}_k - q_k(0)\right)}{\hat{t}} & \forall\ 1 \leq k \leq n_p \\ \|i_k^a\| \leq \hat{i}_k^a & \forall\ 1 \leq k \leq n_a \\ \|i_k^p\| \leq \hat{i}_k^p & \forall\ 1 \leq k \leq n_p \end{cases}$$

Assume that $\vec{q}(0) \geq \underline{\vec{q}}$ – otherwise, the problem would be trivially unsolvable. Let the discharge currents be such that

$$i_k^d = 1 + \frac{q_k(0) - \underline{q}_k}{\hat{t}}$$

Notice that they are valid, since they are real and positive as $\vec{q}(0) - \underline{\vec{q}} \geq 0$ and $\hat{t} \geq 0$. Substituting in the system of equations, it follows that

$$\begin{cases} \|i_k^p\| \geq 1 & \forall\ 1 \leq k \leq n_p \\ \|i_k^a\| \leq \hat{i}_k^a & \forall\ 1 \leq k \leq n_a \\ \|i_k^p\| \leq \hat{i}_k^p & \forall\ 1 \leq k \leq n_p \end{cases}$$

If we set $\hat{i}_k^p = 1\ \forall\ k$, the system is equivalent to Equation 4.3. As demonstrated by the proof of the Lemma 4.3.5, an algorithm which solves that system of equations can also be used to solve a generic instance of the *Auxiliary Problem.*

□

Corollary 4.3.7.1. *Deciding if $\vec{q} \in F_\tau$ is reachable from $\vec{q}' \in F_{\tau-1}$ is NP-Complete.*

Proof. This problem can be summarized as deciding if there is $\vec{v}^a$ for which

$$\begin{cases} q_k' + \Delta t \cdot C_k\left(\eta_k\left(\|i_k^p\|\right) - i_k^d\right) = q_k & \forall\ 1 \leq k \leq n_p \\ \rho_{act}(\tau) \leq \hat{\rho}_{act} & \\ \|i_k^a(\tau)\| \leq \hat{i}_k^a & \forall\ 1 \leq k \leq n_a \\ \|i_k^p(\tau)\| \leq \hat{i}_k^p & \forall\ 1 \leq k \leq n_p \end{cases}$$

Let $\Delta t = 1$, $\hat{\rho}_{act} = \infty$, $\hat{i}_k^p = \infty\ \forall\ k$, $i_k^d = 0\ \forall\ k$, $q_k' = 0\ \forall\ k$, $q_k = 1\ \forall\ k$, $C_k(x) = x\ \forall\ k$, and $\eta_k(x) = x\ \forall\ k$. This instance of the problem is equivalent to decide if there is a voltage vector for which

$$\begin{cases} \|i_k^p(\tau)\| = 1 & \forall\ 1 \leq k \leq n_p \\ \|i_k^a(\tau)\| \leq \hat{i}_k^a & \forall\ 1 \leq k \leq n_a \end{cases}$$

This system is equivalent to Equation 4.3 and, as demonstrated by the proof of the Lemma 4.3.5, an algorithm which solves that system of equations can also be used to solve a generic instance of the *Auxiliary Problem.* Thus, this problem is Np-Hard. Since a given voltage-vector can be verified in polynomial-time – notice that this time we have a single time-slot to check – the problem is NP and therefore it is NP-Complete. □

4.4 The greedy approach for the optimization version

The greedy approach consists of applying the voltages which maximize the sum of the charging currents at each instant. Similar methods are employed by MagMIMO [Jadidian and Katabi, 2014], by *MultiSpot* [Shi et al., 2015] and by Cao et al. [2018]. So, for each moment t, the method applies the voltage $\vec{\mathbf{v}}^a$ calculated as in Equation 4.4.

$$\begin{aligned} \vec{\mathbf{v}}^a &= arg\ max\ \left\{\sum_{j=1}^{n_p} \eta_j \left(C_j\left(\|i_j^p\|\right)\right)\right\} && \\ \text{s.t.} \quad & \rho_{act}(\tau) \le \hat{\rho}_{act} && \forall\ 0 \le \tau \le t \\ & \|i_k(t)\| \le \hat{i}_k && \forall\ 0 \le \tau \le t,\ 1 \le k \le n \end{aligned} \tag{4.4}$$

Although simple, the greedy approach has no guarantees of optimality. Let us consider a system with two transmitting coils and two charging devices. The impedance sub-matrices are defined as follows.

$$Z_T = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad Z_R = \begin{bmatrix} 1 & -10\sqrt{-1} \\ -10\sqrt{-1} & 3 \end{bmatrix}, \quad M = \begin{bmatrix} -10\sqrt{-1} & -5\sqrt{-1} \\ -1\sqrt{-1} & -5\sqrt{-1} \end{bmatrix}$$

Assume the receiving device with 1 Ω static resistance is **Device A** and the one with 3 Ω is **Device B**. Device B needs at least 875 mA to charge, while **Device A** has no requirements. Besides the eventual lower bound, both devices convert the receiving current to charging current with total efficiency. The maximum amplitude for the currents is set as infinity, while the maximum allowed active power is 10 W. For both devices, $\mathcal{R}(q) = q$ and the battery capacity is 10 units of charge. The discharge currents are always zero.

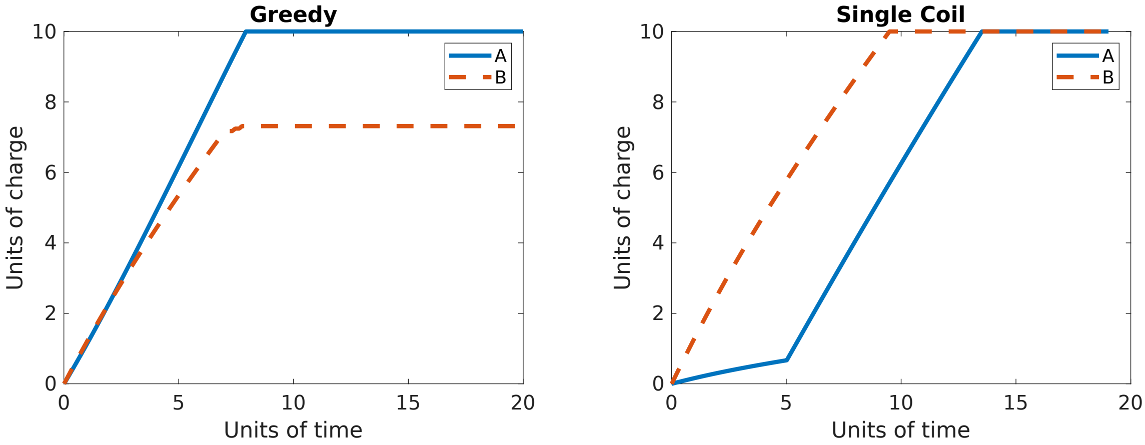

Figure 4.1. Time progression of the charges.

The example is constructed in such a way that **Device A** acts as a waveguide to **Device B**, that is, the current induced from coil **A** to coil **B** is quite significant to the total receiving current of **B**. Furthermore, when **Device A** finishes charging, the load resistance is large enough for the waveguide to be interrupted. In this case, the receiving current of **Device B** drops below 875 mA and its charging is also interrupted. Like the load resistance of **A**, the load resistance of **B** does not change anymore, since it depends only on its charge. So, the system turns stationary and, therefore, **Device B** will never finish charging.

Figure 4.1 compares the outcomes when the greedy approach is employed and when a single transmitting coil is excited at a time. The integration step is 0.01 units of time and both simulations last for 20 units of time. Under the greedy approach, **Device A** finishes charging too soon, so **Device B** cannot finish charging. Despite being naive, the other approach successfully finishes both charging processes, since **Device B** is completely prioritized at the beginning of the simulation. The transmitting coil with higher coupling to coil B is excited with full power until 5 units of time and, after that, the other coil receives full power. As a result, **Device B** finishes charging before **Device A** and, therefore, it is contemplated with waveguiding for the whole process.

4.5 The Exact solution

This section describes a gadget for solving both the optimization version of the *Minimum-Time MIMO Charging Problem* and the decision version of the *No-Starvation MIMO Charging Problem.* We focused on the decision version of the *No-Starvation MIMO Charging Problem* instead of the optimization version because it might have an infinite-sized solution, which is impossible to compute since the problem formulation states that each time-slot has independent parameters. Thus, it would be necessary to compute each voltage vector of the infinite time-series. The decision version, on the other hand, has a fixed-size time horizon.

From a practical point of view, this decision makes the timeline have to be divided into time horizons, each one being an instance of the **No-Starvation MIMO Charging Problem**. Therefore, the algorithms described here introduce a new constraint to guarantee that the batteries at the end of a time horizon have enough charge to survive the next one. Otherwise, a device might be at imminent disconnection when the next time-horizon starts. Thus, the charge vector $\vec{q}(\hat{t})$ in the last time-slot must be such that $\vec{q}(\hat{t}) \geq \underline{\vec{Q}}$ for $\underline{\vec{Q}} \in \mathbb{R}^{n_p}_{\geq 0}$.

First, we get a method for deciding if a given instance is solvable using exactly $t \in \mathbb{N}$ time-slots. Moreover, the method determines a sequence of voltage-vectors whereby the instance is solved. The asymptotic complexity of these methods is linear regarding the number of time-slots, which represents an enormous advantage. Indeed, the number of time-slots must be very large for the discrete approximation of the charge integral to be accurate.

Definition 4.5.1. *Given an instance of the Minimum-Time MIMO Charging Problem/No-Starvation MIMO Charging Problem,* $\vec{q}_1$ ***is reachable from*** $\vec{q}_0$ ***at time***

τ if and only if there is at least a feasible voltage-vector $\vec{v}^a(\tau)$ for which $\vec{q}(\tau) = \vec{q}_1$ given $\vec{q}(\tau - 1) = \vec{q}_0$.

Definition 4.5.2. *Given a set $\mathcal{Q}$ of charge-vectors and a time-slot τ from a determined instance of the Minimum-Time MIMO Charging Problem/No-Starvation MIMO Charging Problem, the charge-vector $\vec{q}_1$ is a **Feasible Future Charge-Vector** (FFCV) at time τ if and only if there is at least one charge-vector $\vec{q}_0 \in \mathcal{Q}$ such that $\vec{q}_1$ is reachable from $\vec{q}_0$ at time τ.*

Definition 4.5.3. *Given a set $\mathcal{Q}$ of charge-vectors and a time-slot τ from a determined instance of the Minimum-Time MIMO Charging Problem/No-Starvation MIMO Charging Problem, the **Feasible Future** $\mathcal{F}(\mathcal{Q}\,|\,\tau)$ of $\mathcal{Q}$ given τ is the set of all FFCV of $\mathcal{Q}$ at τ.*

Definition 4.5.4. *Given an instance of the Minimum-Time MIMO Charging Problem/No-Starvation MIMO Charging Problem, the j**-th Feasible Future** F_j is recursively defined as follows.*

$$\begin{cases} F_0 = \{\vec{q}(0)\} \\ F_j = \mathcal{F}(F_{j-1} \mid j) \end{cases}$$

Lemma 4.5.5. *The j-th Feasibility Future is equivalent to the set of all charge-vectors reachable from $\vec{q}(0)$ within exactly j time-slots.*

Proof. We divide the proof into *(1)* all elements in F_j are reachable from $\vec{q}(0)$ within exactly j time-slots and *(2)* the elements in F_j are the only reachable ones starting from $\vec{q}(0)$ and using exactly j time-slots.

1. Using mathematical induction. **Base case:** F_0 is $\vec{q}(0)$ by definition, which is trivially the only vector reachable from itself without any step. **Induction step:** Suppose F_{j-1} is the set of all charge-vectors reachable from $\vec{q}(0)$ within $j - 1$ time-slots. $F_j = \mathcal{F}(F_{j-1} \mid j)$, so F_j is the set of all charge-vectors reachable from at least one element from F_{j-1} at time j. Thus, the elements in F_j are reachable from $\vec{q}(0)$ by turning $\vec{q}(0)$ into the corresponding element from F_{j-1} using $t - 1$ time-slots and then into the corresponding element from F_j using slot j. Therefore, each element in F_j is reachable from $\vec{q}(0)$ using exactly $(j - 1) + 1 = j$ time-slots.

2. Proof by contradiction. Suppose $\vec{q'}$ such that $\vec{q'} \notin F_j$ and $\vec{q'}$ is reachable from $\vec{q}(0)$ using exactly j time-slots. Thus, there is a feasible voltage time-series $\vec{v}^a(\tau)$ for $0 < \tau \leq j$ which turns $\vec{q}(0)$ into $\vec{q'}$ using j time-slots and a charge time-series $\vec{q'}(\tau)$ where $\vec{q'}(0) = \vec{q}(0)$ and $\vec{q'}(j) = \vec{q'}$. Using mathematical induction, it is possible to demonstrate that $\vec{q'}(\tau) \in F_\tau \ \forall\ 0 \leq \tau \leq j$ as follows.

Base case: $\vec{q'}(0) = \vec{q}(0)$ is trivially reachable from $\vec{q}(0)$ without any time-slot. **Induction step:** Suppose $\vec{q'}(\tau-1) \in F_{\tau-1}$. $\vec{q'}(\tau)$ is reachable from $\vec{q'}(\tau-1)$ at time τ since $\vec{v}^a(\tau)$ is feasible for all $0 < \tau \leq j$ by hypothesis. Thus, $\vec{q'}(\tau) \in \mathcal{F}(F_{\tau-1} \mid \tau)$, which is equivalent to $\vec{q'}(\tau) \in F_\tau$.

In short, $\vec{q'}(\tau) \in F_\tau\ \forall\ 0 \leq \tau \leq j$ implies $\vec{q'} \in F_j$, which is a contradiction.

□

Theorem 4.5.6. *A given instance of the No-Starvation MIMO Charging Problem is solvable in exactly t time-slots if and only if $\exists\ \vec{q'} \in F_t$ such that $\vec{q'} \geq \underline{\vec{Q}}$.*

Proof. According to Lemma 4.5.5, F_t is exactly the set of all charge-vectors reachable from $\vec{q}(0)$ using exactly t time-slots. So, any vector from F_t is associated with at least one voltage time-series which is feasible regarding maximum power, maximum current, and minimal charge constraints for each time-slot. Thus, the solution associated with any $\vec{q'} \in F_t$ trivially satisfies all constraints but $\vec{q}(t) \geq \underline{\vec{Q}}$ and, therefore, the solution is feasible if and only if $\vec{q'} \geq \underline{\vec{Q}}$. □

Theorem 4.5.7. *The optimization version of the Minimum-Time MIMO Charging Problem is equivalent to find the minimal $\tau \in \mathbb{N}$ for which $\exists\ \vec{q'} \in F_\tau$ such that $\vec{q'} \geq \underline{\vec{Q}}$*

Proof. Similarly to the proof of Theorem 4.5.6, the solution associated with any $\vec{q'} \in F_\tau$ trivially satisfies all constraints but $\vec{q}(\tau) \geq \underline{\vec{Q}}$ and, therefore, the solution is feasible if and only if $\vec{q'} \geq \underline{\vec{Q}}$. By hypothesis, τ is the first time-slot with a feasible solution and, therefore, the solution is minimal. □

4.6 Graphical Interpretation of the Exact Solution

Figure 4.2 illustrates how Theorem 4.5.7 can be used to solve an instance of the *Minimum-Time MIMO Charging Problem.* Let the large squares be the charge-vector space for each time-slot. Let the small red squares in slot j be the vector sub-spaces which belong to F_j. In other words, these squares are the ones whose charge-vectors are reachable from vectors within red squares in the previous time-slot. Let the blue squares be the vector sub-spaces whose charge-vectors surpass the threshold $\underline{\vec{Q}}$. The blue path runs through feasible intermediate states and reaches a feasible final state whose charges exceed the threshold. Each hop is associated with a voltage-vector which enables the previous state to reach the actual state. Thus, the storage of these vectors allows finding the voltage time-series that solves the problem.

Figure 4.3, in turn, illustrates how Theorem 4.5.6 can be used to solve an instance of the *No-Starvation MIMO Charging Problem.* Notice that it works in a very similar way

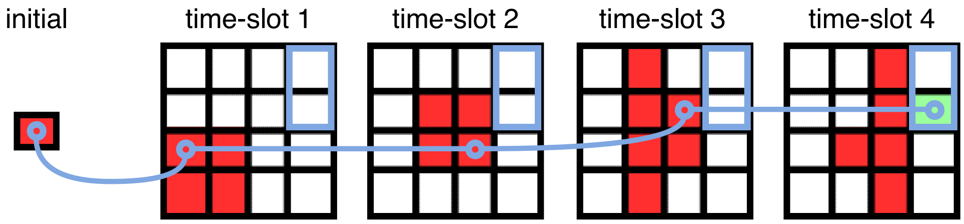

Figure 4.2. The *Minimum-Time MIMO Charging Problem* aims at finding the fastest path to any valid final charge-vector.

as in Figure 4.2, although it does not stop enumerating the feasible futures after finding the first charge-vector which exceeds the threshold. In exchange, it finds a path that runs through feasible states within all time-slots in the time horizon until it finds a valid final state in the last time-slot. Thus, the voltage time-series associated with the blue path ensures the no-starvation of the devices throughout the whole time window.

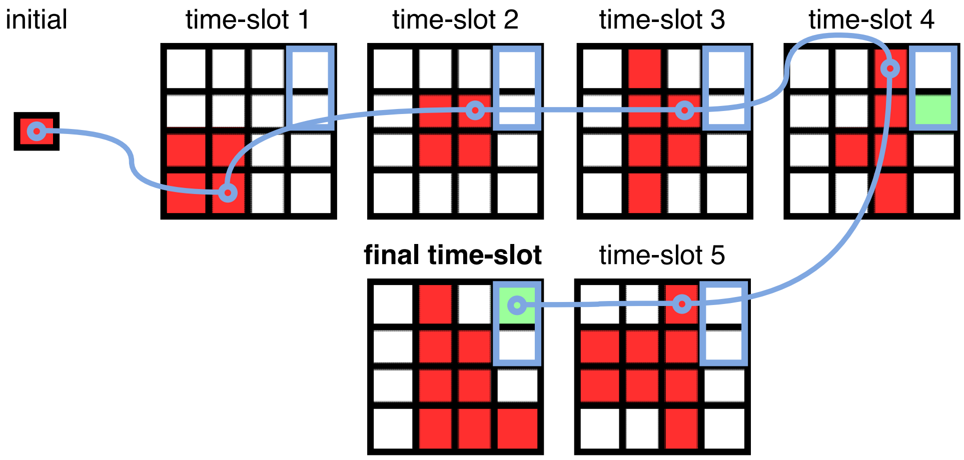

Figure 4.3. The *No-Starvation MIMO Charging Problem* aims at finding a path to any valid final charge-vector within the last time-slot.

Chapter 5

Proposed Algorithms

5.1 Overview

This chapter describes algorithms for solving the problems proposed in Chapter 4. It first introduces two methods used to populate the feasible futures (Definition 4.5.4). Then it describes the proposed algorithms themselves. Next, it analyzes the asymptotic complexity of those and, finally, it discusses the parameter acquisition in a real-life system.

Strictly speaking, all of the algorithms described here are heuristics. Despite that, the algorithms from Section 5.4.1 approach the exact method from Chapter 4, differing only in that they do not necessarily fully populate the feasible futures for reasons of efficiency. All those algorithms can be used to solve both problems, with the only difference being the stop condition. For the *Minimum-Time MIMO Charging Problem*, the algorithms must stop populating the feasible futures as soon as they find an acceptable finishing state. For the *No-Starvation MIMO Charging Problem*, in turn, they must populate all feasible futures within the time horizon and then evaluate if the last feasible future has at least an acceptable finishing state.

The simplified generic algorithm for the *No-Starvation MIMO Charging Problem* is summarized in Algorithm 1. It is based on Theorem 4.5.6, which allows the expansion of the feasible future states through the time-slots parting form a determined initial state using a dynamic-programming strategy. The feasible futures are represented as hash-based sets of points in $\mathbb{R}^{n_p}$, which will be mentioned hereinafter as *point clouds*. Each state vector $\vec{q}$ in a point cloud F_τ is related to a previous state vector $\vec{q}_0 \in \mathbb{R}^{n_p}$ in $F_{\tau-1}$ and also to a voltage vector $\vec{v}^a \in \mathbb{R}^{n_a}$ such that $\vec{q}$ is reachable from $\vec{q}_0$ at time τ by applying $\vec{v}^a$ voltages into the active circuits.

The simplified generic algorithm for the *Minimum-Time MIMO Charging Problem*, in turn, is summarized in Algorithm 2. It follows an idea very similar to that of Algo-

Algorithm 1 Algorithm for the No-Starvation MIMO Charging Problem

```
$t \leftarrow 0$
$F_t \leftarrow \{\vec{q}(0)\}$
$S \leftarrow \emptyset$
while $t \leq max_t$ do
  if $F_t$ full or $S = \emptyset$ then
    if $F_t = \emptyset$ then
      return failure
    else
      $t \leftarrow t + 1$
      $S \leftarrow F_{t-1}$
    end if
  else
    $\vec{q}' \leftarrow$ get any from $S$
    $S \leftarrow S - \{\vec{q}'\}$
    $F_t = F_t \cup \mathcal{F}(\{\vec{q}'\} \,|\, t)$
  end if
end while
if $\exists\, \vec{q} \in F_{max_t} \,|\, q \geq \underline{\vec{Q}}$ then
  return Voltage sequence in $F_{0..max_t}$ starting from $\vec{q}$
else
  return failure
end if
```

Algorithm 2 Algorithm for the *Minimum-Time MIMO Charging Problem*

```
$t \leftarrow 0$
$F_t \leftarrow \{\vec{q}(0)\}$
$S \leftarrow \emptyset$
while $\nexists\, \vec{q} \in F_t \,|\, q \geq \underline{\vec{Q}}$ do
  if $F_t$ full or $S = \emptyset$ then
    if $t \geq max_t$ or $F_t = \emptyset$ then
      return failure
    else
      $t \leftarrow t + 1$
      $S \leftarrow F_{t-1}$
    end if
  else
    $\vec{q}' \leftarrow$ get any from $S$
    $S \leftarrow S - \{\vec{q}'\}$
    $F_t = F_t \cup \mathcal{F}(\{\vec{q}'\} \,|\, t)$
  end if
end while
return Voltage sequence in $F_{0..max_t}$ starting from $\vec{q}$
```

rithm 1, except for the stop condition, as mentioned before. Both algorithms build the final solution starting from the acceptable finishing state and then going way back to the initial state, as described by Algorithm 3.

Algorithm 3 Voltage sequence in $F_{0..t}$ starting from $\vec{q}$

```
if $t = 0$ then
  return []
else
  $\vec{q}' \leftarrow F_t$.findQ0 ( $\vec{q}$ )
  $\vec{v}^a \leftarrow F_t$.findV ( $\vec{q}$ )
  return [voltage sequence in $F_{0..t-1}$ starting from $\vec{q}'$, $\vec{v}^a$]
end if
```

One might create the τ-th Feasible Future F_τ from the $(\tau - 1)$-th Feasible Future $F_{\tau-1}$ by systematically testing the reachability between each *polytope* from $F_{\tau-1}$ and each one of the s^{n_p} *polytopes* that may compose F_τ. Thus, for each $\vec{q} \in F_{\tau-1}$ and each candidate $\vec{q}'$ for F_τ, the algorithm would search for a $\vec{v}^a$ which enables $\vec{q}'$ to be reachable by $\vec{q}$ time τ.

However, as demonstrated by Corollary 4.3.7.1, deciding if $\vec{q}(\tau)$ is reachable from $\vec{q}(\tau - 1)$ is *NP-Complete*. So, it is reasonable to assume that an algorithm that solves the general case of the reachability problem belongs to $\Omega(2^n)$ complexity class. Therefore, in the worst case, the algorithm for populating the feasible features mentioned above would perform a $\Omega(2^n)$ operation $O(s^{n_p}) \cdot s^{n_p} = O(s^{2n_p})$ times.

Thus, the algorithms used in this work for building new feasible futures are based on two different methods. The **exploration** method creates a set of feasible future charge-vectors in F_τ using a $\vec{q}(\tau-1)$ charge-vector and random voltage-vectors. The **exploitation** method, in turn, searches for a $\vec{v}^a(\tau)$ which enables $\vec{q}(\tau)$ to be reachable from $\vec{q}(\tau-1)$, just like the aforementioned algorithm. Despite that, the exploitation is easier in the average case because it starts searching from a *quasi-optimal* solution.

5.2 Exploration

This section describes a method for creating new feasible futures $\vec{q}(\tau)$ from a given past charge-vector $\vec{q}(\tau - 1)$ and a random vector $\vec{\mathbf{v}}^a$ where $\mathbf{v}^a_j \sim U(-1, 1)$.

Let us define the voltage-vector $\vec{v}$ as $\vec{v} \triangleq k \cdot \vec{\mathbf{v}}$, where $k \in \mathbb{R}$. The corresponding current vector $\vec{i}$ given the Z impedance matrix is then $\vec{i} = k \cdot Z^{-1}\,\vec{\mathbf{v}}$. Furthermore, the current of the j-th device is given by $i_j = k \cdot (Z^{-1})_{j,\bullet}\,\vec{\mathbf{v}}$ and its absolute value by $\|i_j\| = \|k\| \cdot \|(Z^{-1})_{j,\bullet}\,\vec{\mathbf{v}}\|$. Thus, the necessary and sufficient condition for $\|i_j\|$ to be upper-limited by $\hat{i}_j$ is shown in Equation 5.1.

$$\|k\| \leq \frac{\hat{i}_j}{\|(Z^{-1})_{j,\bullet}\,\vec{\mathbf{v}}\|} \tag{5.1}$$

Similarly, one might define the active power in terms of $\vec{\mathbf{v}}$, so achieving

$$\rho = Re\left((k \cdot Z^{-1}\,\vec{\mathbf{v}})^* \,(k \cdot \vec{\mathbf{v}})\right)$$

Rearranging the terms, it follows that

$$\rho = Re\left(k^2 \cdot \mathbf{v}^*(\vec{Z}^{-1})^*\,\vec{\mathbf{v}}\right)$$

And, since $\vec{\mathbf{v}}$ and k are both real, the active current can be described as in Equation 5.2.

$$\rho = k^2 \cdot Re\left(\vec{\mathbf{v}}^t(Z^{-1})^*\,\vec{\mathbf{v}}\right) \tag{5.2}$$

The necessary and sufficient condition for ρ to be upper-limited by $\hat{\rho}$ is shown in Equation 5.3.

$$\|k\| \leq \sqrt{\frac{\rho}{Re\left(\vec{\mathbf{v}}^t(Z^{-1})^*\,\vec{\mathbf{v}}\right)}} \tag{5.3}$$

According to Equation 3.9, the charging of the j-th device within a single slot τ is described as follows.

$$q_j(\tau) = q_j(\tau-1) + \Delta t \cdot \eta_j\left(C_j(\|i_j^p(\tau)\|) - i_j^d(\tau)\right)$$

To ensure that $q_j(\tau) > \underline{q}_j$, $\|i_j^p(\tau)\|$ must be such that

$$\eta_j\left(C_j(\|i_j^p(\tau)\|) - i_j^d(\tau)\right) > \frac{1}{\Delta t}\left(\underline{q}_j - q_j(\tau-1)\right)$$

Notice that both η and C conversion functions are monotonically increasing and $i_j^d(\tau)$ is constant, so the necessary and sufficient condition for the minimum charge constraint to be respected at device j and time-slot τ is given by

$$\|i_j^p(\tau)\| > C_<^{-1}\left(\eta_<^{-1}\left(\frac{1}{\Delta t}\left(\underline{q}_j - q_j(\tau-1)\right)\right) + i_j^d(\tau)\right) \triangleq x_j$$

Moreover, a voltage vector $\vec{v} = k \cdot \vec{\mathbf{v}}$ is feasible regarding the minimum charge constraint for a device j and time-slot τ if and only if the inequality from Equation 5.4 is respected.

$$\|k\| > \frac{x_j}{\left\|(Z^{-1})_{(n_a+j),\bullet}\,\vec{\mathbf{v}}\right\|} \tag{5.4}$$

Thus, for a given voltage vector $\vec{v} = k \cdot \vec{\mathbf{v}}$ to be feasible regarding all constraints, k must be as described by Equation 5.5.

$$max_j \left\{ \frac{x_j}{\|(Z^{-1})_{j,\bullet}\,\vec{\mathbf{v}}\|} \right\} < \|k\| \leq min \left\{ \sqrt{\frac{\rho}{Re\left(\vec{\mathbf{v}}^t (Z^{-1})^* \,\vec{\mathbf{v}}\right)}};\, min_j \left\{ \frac{\hat{i}_j}{\|(Z^{-1})_{j,\bullet}\,\vec{\mathbf{v}}\|} \right\} \right\} \tag{5.5}$$

In short, let $(Z^{-1})_R$ be the last n_p rows of Z^{-1} and ϵ be a small real positive scalar. Any future state

$$\vec{q}(\tau) = \vec{q}(\tau - 1) + \Delta t \cdot \vec{\eta}\left(\vec{C}(\left\|k' \cdot (Z^{-1})_R \vec{\mathbf{v}}\right\|) - \vec{i}^d\right)$$

where

$$k' \sim U\left(max_j \left\{ \frac{x_j}{\|(Z^{-1})_{j,\bullet}\,\vec{\mathbf{v}}\|} \right\} + \epsilon, min \left\{ \sqrt{\frac{\rho}{Re\left(\vec{\mathbf{v}}^t (Z^{-1})^* \,\vec{\mathbf{v}}\right)}};\, min_j \left\{ \frac{\hat{i}_j}{\|(Z^{-1})_{j,\bullet}\,\vec{\mathbf{v}}\|} \right\} \right\} \right)$$

is feasible if and only if Equation 5.5 is valid.

5.3 Exploitation

The exploitation aims at finding a voltage vector $\vec{v}^a$ which enables a given state $\vec{q}(\tau-1)$ to reach $\vec{q}$ at time-slot τ. The algorithm admits knowledge about an initial solution $\vec{\mathbf{v}}^a$ which enables $\vec{q}(\tau - 1)$ to reach $\vec{q} + \vec{\epsilon}$ at time-slot τ, where ϵ_k is a very small real value. Thus, for $\vec{q}(\tau - 1)$ to reach $\vec{q}$, each receiving current $\left\|i^p_j\right\|$ must be such that $x^<_j \leq \left\|i^p_j\right\| \leq x^>_j$, where

$$x^<_j \triangleq C^{-1}_< \left(\eta^{-1}_< \left(\frac{1}{\Delta t} \left(q_j - q_j(\tau - 1)\right) \right) + i^d_j(\tau) \right)$$

$$x^>_j \triangleq C^{-1}_> \left(\eta^{-1}_> \left(\frac{1}{\Delta t} \left(q_j - q_j(\tau - 1)\right) \right) + i^d_j(\tau) \right)$$

Let $\vec{\mathbf{i}}^p$ be the receiving current vector corresponding to the initial solution $\vec{\mathbf{v}}^a$. The method chooses a target receiving current vector $\vec{i}^{target}$ such that

$$\begin{cases} i_j^{target} = x_j^< & if \quad \left\| i_j^p \right\| < x_j^< \\ i_j^{target} = \left\| i_j^p \right\| & if \quad x_j^< \leq \left\| i_j^p \right\| \leq x_j^> \\ i_j^{target} = x_j^> & if \quad \left\| i_j^p \right\| > x_j^> \end{cases}$$

which respects the intervals for reaching $\vec{q}$ and is as close as possible to the initial solution. Then, the problem to be solved is summarized as the following.

$$\begin{cases} \left\| i_j^p \right\| = i_j^{target} & \forall 1 \leq j \leq n_p \\ \left\| i_j^a \right\| \leq \hat{i}_j & \forall 1 \leq j \leq n_a \\ Re\{(\vec{i}^a)^t \vec{v}^a\} \leq \rho & \end{cases}$$

Let z_j be the first n_a columns of the j-th row of Z^{-1}. Let $\mathbf{Z}$ be the first n_a columns of the first j rows of Z^{-1}. Thus, $i_j^p = z_j \vec{v}^a$ and $\vec{i}^a = \mathbf{Z}\vec{v}^a$. Moreover, the problem can be rewritten as follows. Notice the use of *slack variables* in order to transform inequalities into equations.

$$\begin{cases} (\vec{v}^a)^t (z_j^p)^* (z_j^p) \vec{v}^a = (i_j^{target})^2 & \forall 1 \leq j \leq n_p \\ (\vec{v}^a)^t (z_j^a)^* (z_j^a) \vec{v}^a + (s_j^a)^2 = (\hat{i}_j)^2 & \forall 1 \leq j \leq n_a \\ Re\{(\vec{v}^a)^t \mathbf{Z}^t \vec{v}^a\} + s_\rho^2 = \rho & \end{cases}$$

All decision variables can be reorganized as a single vector $\vec{x} \triangleq [\vec{v}^a; s_\rho; \vec{s}^a]$. Let $\vec{u}_k \triangleq [0_{j-1}, 1, 0_{n_a - j}]$, that is, the vector where all n_a positions are zero except for the j-the entry, which is one. For convenience, the following definitions are pertinent.

$$A_j \triangleq \begin{bmatrix} (z_j^p)^*(z_j^p) & 0_{n_a} & \vec{0} \\ 0_{n_a} & 0_{n_a} & \vec{0} \\ \vec{0}^t & \vec{0}^t & 0 \end{bmatrix} \quad B_j \triangleq \begin{bmatrix} (z_j^a)^*(z_j^a) & 0_{n_a} & \vec{0} \\ 0_{n_a} & diag(\vec{u}_j) & \vec{0} \\ \vec{0}^t & \vec{0}^t & 0 \end{bmatrix} \quad C \triangleq \begin{bmatrix} Re\{\mathbf{Z}^t\} & 0_{n_a} & \vec{0} \\ 0_{n_a} & 0_{n_a} & \vec{0} \\ \vec{0}^t & \vec{0}^t & 1 \end{bmatrix}$$

Using the new definitions, the problem can be rewritten as

$$\begin{cases} \vec{x}^t A_j \vec{x} - (i_j^{target})^2 = f_j^p & \forall 1 \leq j \leq n_p \\ \vec{x}^t B_j \vec{x} - (\hat{i}_j)^2 = f_j^a & \forall 1 \leq j \leq n_a \\ \vec{x}^t C \vec{x} - \rho = f^\rho & \end{cases}$$

For a given vector $\vec{x}$ to solve the problem, the residue vector $\vec{f} = \left[\vec{f^p}; \vec{f^a}; \vec{f^\rho}\right]$ must be a zero vector. For the first solution, the slack variables are

$$s_\rho = \sqrt{\rho - Re\{(\vec{\mathbf{i}}^a)^t \vec{\mathbf{v}}^a\}}$$

$$s_j^a = \sqrt{(\hat{i}_j^a)^2 - \left\|\mathbf{i}_j^a\right\|^2)}$$

and the residue vector is all zeros except by $\vec{f}^p$. The slack variables are guaranteed to be real since the original solution $\vec{\mathbf{v}}^a$ is feasible by definition.

The next solutions are calculated iteratively using the *Pseudo-Inverse Newton-Raphson* method, which converges locally if the rank of the *Jacobian* matrix is constant [Gatilov, 2014]. Since the initial solution is assumed to be quasi-optimal, these conditions are assumed to be respected. The main formula is shown by Equation 5.6, where J is the *Jacobian* of the system.

$$\vec{x} = \vec{x}_0 - J^+ \vec{f} \tag{5.6}$$

Since all equations are bilateral-quadratic, the calculations for the *Jacobian* are quite simple. The j-th row of the matrix is the transpose of the gradient of the j-th equation $\vec{x}^t D_j \vec{x}$ evaluated at $\vec{x}_0$. Thus, $J_{j\bullet} = (\nabla(\vec{x}^t D_j \vec{x}))^t = \vec{x}^t(D_j + D_j^t)$.

Figure 5.3 shows the effectiveness of the described method for different magnitudes of $\vec{\epsilon}$ considering 1000 repetitions. The referred noise factor is the ratio between the maximum magnitude of $\vec{\epsilon}$ and the maximum value of $\vec{q}$.

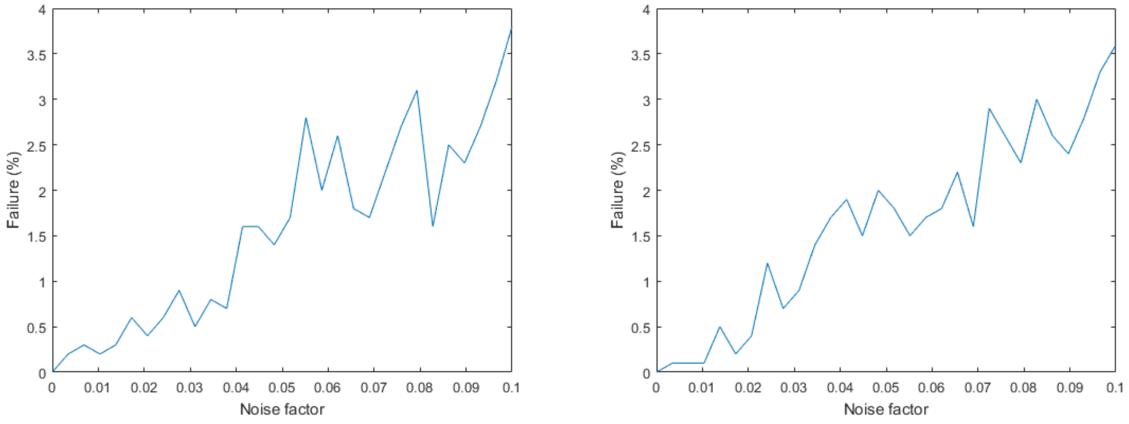

Figure 5.1. Failure ratio for different noise factors considering 10 and 100 maximum number of iterations, respectively.

5.4 Algorithms

The algorithms below are all based on algorithms 1 and 2, which, in short, build sequentially new feasible future sets at each time-slot based on the immediately preceding sets. Therefore, the difference between the algorithms described in this section is basically how they build the τ-th Feasible Future F_τ from the previous set $F_{\tau-1}$.

Moreover, each one of the algorithms described below can be used to solve both problems. If they follow the processing flow defined by Algorithm 1, they will handle instances of the *No-Starvation MIMO Charging Problem.* If they follow the processing flow defined by Algorithm 2, in turn, they will handle instances of the *Minimum-Time MIMO Charging Problem.*

The flag *stopOnThresholdReached* is set to true if the algorithm aims at solving an instance of the *Minimum-Time MIMO Charging Problem* or if the actual time-slot is the last one. Thus, there is no point in keeping the search after a valid final state is found.

5.4.1 Dynamic-Programming Algorithms

5.4.1.1 Simple Algorithm

This algorithm is entirely based on the exploration method and on a floating-point based point cloud. The feasible future F_τ starts empty and then is filled using feasible charge vectors generated from a random choice of $\vec{q}(\tau - 1) \in F_{j-1}$ and a randomly sampled base-voltage vector $\vec{\mathbf{v}}^a$. The algorithm stops filling the set when (i) it has a maximum number of elements, (ii) a predefined time-to-leave reaches zero, or (iii) the last attempts using a random previous-charge vector and a random base-voltage vector could not produce any new feasible future state.

Let *thr* be the maximum acceptable number of consecutive failed attempts to produce a feasible new future state. Let *ttl* be the maximum number of attempts to produce a new future state. Let ϵ be a very small real positive scalar. The *Simple* Algorithm is summarized in Algorithm 4.

The execution flow is cascaded in three nested loops to allow the optimization of space usage. For each previous-state chosen, an entire population of new states is generated. Therefore, all F_τ *polytopes* may be filled using a small number of previous states. Moreover, the system can discard previous-states that did not generate new states in F_τ. Indeed, they cannot be intermediate states between the first and the last time-slot. Thus, if memory-space is scarce, the system may reduce memory usage using the following procedure. All elements from $F_{\tau-1}$ start signaled with *false.* When a previous-vector is successful at generating a valid next state, it is signaled with *true.* Then, after F_τ is ready, the system might create a new $F_{\tau-1}$ with only the elements signaled with *true.*

5.4.1.2 Fly-Weight Algorithm

This algorithm is based on a fly-weight point cloud. Thus, the charge vectors and the previous charge vectors are stored using their discretized versions to save memory usage. In particular, this algorithm uses around 5 times less memory than the *Simple* algorithm.

Algorithm 4 Feasible Future set generation for using the *Simple* Algorithm

```
$F_j \leftarrow \emptyset$
while $TTL_1 > 0$ and $failures < thr$ and $F_\tau$ not full do
  $TTL_1 \leftarrow TTL_1 - 1$
  $(\vec{q}, \vec{d}) \leftarrow$ any element of $F_{\tau-1}$
  while $TTL_2 > 0$ and $failures < thr$ and $F_\tau$ not full do
    $TTL_2 \leftarrow TTL_2 - 1$
    Sample $\vec{\mathbf{v}}^a$ where $\mathbf{v}_j \sim U(-1, 1)$
    Get $(k_<, k_>)$ for $(\vec{q}, \vec{\mathbf{v}}^a)$ using the exploration method
    if $k_< + \epsilon > k_>$ then
      $failures \leftarrow failures + 1$
    else
      for $k \leftarrow k_< + \epsilon \, : \, k_>$ do
        $\vec{v}^a \leftarrow k \cdot \vec{\mathbf{v}}^a$
        $\vec{q} \leftarrow$ the charge vector corresponding to $\vec{v}^a$
        if $\vec{q} \notin F_\tau$ then
          Insert $(\vec{q} \notin, \vec{v}^a, \vec{d})$ into $F_\tau$
          $failures \leftarrow 0$
        else
          $failures \leftarrow failures + 1$
        end if
        if $returnIfThreholdReached$ and $\vec{q} \geq \hat{\vec{Q}}$ then
          return $F_\tau$
        end if
      end for
    end if
  end while
end while
return $F_\tau$
```

Therefore, the centroids are used as representants of the charge-vectors within the same *polytope*, so that the error propagation along the time-slots is minimized.

Populating the *fly-weight* point cloud is similar to populating the *floating-point* version using the *Simple* algorithm. However, each charge vector $\vec{q}$ generated from the previous charge vector $\vec{q}_0$ is substituted by the centroid $\vec{q}'$ of its *polytope* using the exploitation method. If the exploitation method fails to find a vector $\vec{v}^a$ which enables $\vec{q}_0$ to reach $\vec{q}'$, the new element is not inserted into F_τ and the attempt is counted as a failure.

5.4.1.3 Pareto Algorithm

This algorithm is also based on a floating-point point cloud but has two differences towards the *Simple* Algorithm. The first difference is that at most one next state is inserted into F_τ for each pair of base-voltage vector $\vec{\mathbf{v}}^a$ and previous-charge vector $\vec{q}_0$. If the feasible interval $[k_<, k_>]$ is non-empty, the chosen next state is the one obtained using $\vec{v}^a = k_> \cdot \vec{\mathbf{v}}^a$, that is, the feasible voltage vector which maximizes the power transfer. Intuitively, using many of these charge vectors as intermediate states will reduce the chances of no feasible future state to be found for one of the next time-slots. Furthermore, these vectors speed up the charging of the devices, which can be interesting for solving the *Minimum-Time MIMO Charging Problem.*

Besides that, the elements from $F_{\tau-1}$ used as previous-charge vectors are selected sequentially without repetitions. Thus, if TTL and THR are large enough, the algorithm will select all elements from $F_{\tau-1}$ exactly one time. This strategy is positive for problem instances with very few feasible solutions when compared with the total number of values for the decision variables. Thus, using all possible previous-states prevents the algorithm to lose a rare intermediate state which enables finding the solution.

5.4.2 Greedy Algorithms

The algorithms described in this sub-section are all greedy and, therefore, store a single state of each feasible future.

5.4.2.1 Max-Sum-of-Currents Algorithm

This algorithm is an adaptation of algorithms such as *MultiSpot* and Yang [2017], which employs the voltage vector which maximizes the receiving power. Thus, this algorithm is basically the one referred in Section 4.4. In this case, besides the insertion of the maximum current and minimum charge constraints, the objective function was altered to maximize the sum of the absolute value of the receiving currents. This modification was due the currents are more intimately related to the charge variations than the transferred power. Algorithm 5 summarizes the *Max-Sum-of-Currents* Algorithm. The small real-positive scalar ϵ is used to guarantee the limit $k_<$ as exclusive since it represents the threshold for which the charge of at least one device drops below the minimum allowed.

5.4.2.2 Max-Sum Algorithm

This algorithm is almost the same as *Max-Sum-of-Currents* Algorithm but maximizes the sum of the charges instead of the magnitude of the receiving currents. The idea behind this algorithm is that the current-conversion functions can be complex and depend on the used devices. Thus, it searches locally for the best average state for the whole network.

Algorithm 5 Feasible Future set generation for the Max-Sum-of-Currents Algorithm

```
$F_j \leftarrow \emptyset$
$(\vec{q}, \vec{d}) \leftarrow$ the only element of $F_{\tau-1}$
$s \leftarrow 0$
while $TTL > 0$ do
    $TTL \leftarrow TTL - 1$
    Sample $\vec{\mathbf{v}}^a$ where $\mathbf{v}_j^a \sim U(-1, 1)$
    Get $(k_<, k_>)$ for $(\vec{q}, \vec{\mathbf{v}}^a)$ using the exploration method
    $\lambda \leftarrow (1 - \epsilon) \cdot \lambda' + \epsilon$, where $\lambda' \sim U(0, 1)$
    $\vec{v}^a \leftarrow (\lambda \cdot k_< + (1 - \lambda) \cdot k_>)\vec{\mathbf{v}}^a$
    $\vec{q} \leftarrow$ the charge vector corresponding to $\vec{v}^a$
    $\vec{i}^p \leftarrow$ the receiving-current vector corresponding to $\vec{v}^a$
    $s' \leftarrow \sum \left\|\vec{i}^p\right\|$
    if $s' > s$ then
        $\vec{q}' \leftarrow \vec{q}$
        $\vec{v}^{a\prime} \leftarrow \vec{v}^a$
        $s \leftarrow s'$
    end if
    if $returnIfThreholdReached$ and $\vec{q} \geq \hat{\vec{Q}}$ then
        Insert $(\vec{q}, \vec{v}^a, \vec{d})$ into $F_\tau$
        return $F_\tau$
    end if
end while
Insert $(\vec{q}', \vec{v}^{a\prime}, \vec{d})$ into $F_\tau$
return $F_\tau$
```

5.4.2.3 Max-Min Algorithm

Analogously to the *Max-Sum* Algorithm, this one is very similar to the *Max-Sum-of-Currents* Algorithm. The difference is again the objective function. In this case, it is the maximization of the charge q_k of the weakest device k, that is, the device whose charge is the closest to the minimum $\underline{q}_k$.

This approach is notable because there is already an algorithm for selective charging which can be adapted for maximizing the delivered power for a single passive device. In particular, the algorithm proposed by Jung and Lee [2019] employs *semi-definite relaxation* and *rank reduction based on randomization* to maximize the total delivered power. To allow the selective optimization, it provides constraints to impose an upper-bound for the power delivered to unintended devices and a lower-bound to the power of the intended ones. Since the conversion constraints are guaranteed to be monotonically increasing and the load resistance is real-positive, maximizing the charge variation for a single device is equivalent to maximize the power delivered to it.

5.4.2.4 Random-Feasible-Future Algorithm

This heuristic is used exclusively to evaluate the random generation of problem instances from Section 7.1. The generation of the τ-th feasible feature set F_τ given the $(\tau - 1)$-th feasible feature set $F_{\tau-1}$ is described by Algorithm 6.

Algorithm 6 Feasible Future set generation for the Random-Feasible-Future Algorithm

```
$F_j \leftarrow \emptyset$
$(\vec{q}, \vec{d}) \leftarrow$ the only element of $F_{\tau-1}$
Sample $\vec{\mathbf{v}}^a$ where $\mathbf{v}_j \sim U(-1, 1)$
Get $(k_<, k_>)$ for $(\vec{q}, \vec{\mathbf{v}}^a)$ using the exploration method
$\lambda \leftarrow (1 - \epsilon) \cdot \lambda' + \epsilon$, where $\lambda' \sim U(0, 1)$
$\vec{v}^a \leftarrow (\lambda \cdot k_< + (1 - \lambda) \cdot k_>)\vec{\mathbf{v}}^a$
$\vec{q} \leftarrow$ the charge-vector corresponding to $\vec{v}^a$
Insert $(\vec{q}, \vec{v}^a, \vec{d})$ into $F_\tau$
**return** $F_\tau$
```

5.5 Complexity Analysis

The decision variable $\vec{v}^a(\tau)$ can be represented by a $n_a \times t$ matrix, where t is the number of time-slots and n_a is the number of transmitters. Thus, a simple *Brute-force Search Algorithm* would enumerate

$$O(s_a^{n_a \cdot t}) = O(exp\,(log(s_a) \cdot n_a \cdot t))$$

solutions, where s_a is the number of voltage possibilities within a single transmitter.

The dynamic-programming algorithms, in turn, perform $O(s_p^{n_p})$ insertions into each feasible-future set. In total, there are $O(t \cdot s_p^{n_p})$ insertions. The number of evaluated feasible futures is, roughly, $O(s_a^{n_a})$ for each previous state from each time-slot. Thus, the total number of relevant operations is

$$O\left(t \cdot exp(log(s_a) \cdot n_a + log(s_p) \cdot n_p)\right)$$

Thus, the algorithms are also exponential regarding the number of devices. However, they are linear towards the number of time-slots, which is expected to be large.

Except by the *Random Feasible Future* algorithm, the asymptotic complexity of all proposed greedy algorithms can be similarly deduced as

$$O(t \cdot s_a^{n_a}) = O(t \cdot exp\,(log(s_a) \cdot n_a))$$

5.6 Considerations Regarding Parameter Acquisition

The dynamic-programming algorithms admit complete knowledge about all parameters of the system for all time-slots. This assumption is not applicable for all situations and depends on network communication, multiple sensors, information towards the electrical components of all devices, and tight movement planning for all devices at every moment within the time window.

From the positions, geometries, and orientations of all coils, the value of each mutual induction $\mathcal{M}_{i1,i2}$ can be expressed by *Neumann*'s formula, shown in Equation 5.7. The μ_0 factor corresponds to the magnetic permeability of the medium, ζ is the path of one coil, ds is the infinitesimal slice of ζ, and $|D_{i1,i2}|$ is the absolute distance between ds_{i1} and ds_{i2}.

$$M_{i1,i2} = \frac{\mu_0}{4\pi} \oint_{\zeta_{i1}} \oint_{\zeta_{i2}} \frac{ds_{i1} \cdot ds_{i2}}{|D_{i1,i2}|} \tag{5.7}$$

The greedy algorithms, in turn, do not require knowledge about all future time-slots, since the decisions are taken locally. Let us admit no knowledge about the positions, geometries, and orientations of the coils. From Equation 3.2, one might relate a set of voltage samples and the corresponding currents as

$$[\vec{v}(t_1) \quad \vec{v}(t_2) \quad ... \quad \vec{v}(t_m)] = Z[\vec{i}(t_1) \quad \vec{i}(t_2) \quad ... \quad \vec{i}(t_m)]$$

Assuming n_a linearly independent voltage samples, any possible voltage vector $\vec{v}$ can be expressed as a linear combination of the samples, that is,

$$\vec{v} = [\vec{v}(t_1) \quad \vec{v}(t_2) \quad ... \quad \vec{v}(t_m)]\vec{\lambda}, \qquad \vec{\lambda} \in \mathbb{R}^{n_a}$$

Moreover, the transmitting voltages can be expressed as

$$\vec{v}^a = [\vec{v}^a(t_1) \quad \vec{v}^a(t_2) \quad ... \quad \vec{v}^a(t_m)]\vec{\lambda}$$

Notice that $[\vec{v}^a(t_1) \quad \vec{v}^a(t_2) \quad ... \quad \vec{v}^a(t_m)]$ is a square matrix with linearly independent columns and, therefore, the coefficient vector can be calculated using Equation 5.8

$$\vec{\lambda} = [\vec{v}^a(t_1) \quad \vec{v}^a(t_2) \quad ... \quad \vec{v}^a(t_m)]^{-1}\vec{v}^a \tag{5.8}$$

Furthermore, $\vec{i} = Z^{-1}\vec{v}$ is equivalent to Equation 5.9.

$$\vec{i} = Z^{-1}[\vec{v}^a(t_1) \quad \vec{v}^a(t_2) \quad ... \quad \vec{v}^a(t_m)]\vec{\lambda} = [\vec{i}(t_1) \quad \vec{i}(t_2) \quad ... \quad \vec{i}(t_m)]\vec{\lambda} \tag{5.9}$$

Finally, from Equations 5.8 and 5.9, it follows that

$$\vec{i} = [\vec{i}(t_1) \quad \vec{i}(t_2) \quad ... \quad \vec{i}(t_m)][\vec{v}^a(t_1) \quad \vec{v}^a(t_2) \quad ... \quad \vec{v}^a(t_m)]^{-1}\vec{v}^a$$

Notice that this formula is enough for the utilization of the exploration method, which is the base of all proposed greedy algorithms.

Chapter 6

Implementation Decisions

The *Point Cloud* data structure is responsible for storing the charge-vectors which compose a feasible future set. The set has $O(s^{n_p})$ charge-vectors and therefore the structure must be implemented in such a way that the associated operations are time-efficient. In short, the minimal set of operations is the following.

- Search. Verify if exists a charge-vector in the *point cloud* which belongs to the same *polytope* as the informed charge-vector.
- Insertion. Insert the informed charge-vector, its predecessor, and the associated voltage-vector into the *point cloud.*
- Reading. Return the charge-vector, its predecessor, and the associated voltage-vector inside a given *polytope.*
- Signaling. Toggle a boolean flag associated with a determined *polytope.* This operation is used to identify charge-vectors that are predecessors of ones from the next feasible future. Those that aren't can be deleted to save memory.
- Counting. Return the number of elements inside the *point cloud.*
- Uniform sampling. Return one of the charge-vectors inside the *point cloud.* Every element in the *point cloud* must have the same probability to be chosen.
- Sequential sampling. Return the next charge-vector from the *point cloud.* Each element is returned exactly one time.

Populating the *point cloud* with one charge-vector for each possible *polytope* or even storing one structure for each possible *polytope* may be intractable. Thus, the *point cloud* structure must be able to efficiently store large sparse sets of multi-dimensional points. In this work, we implement the *point cloud* as a *hash table* with H buckets, where H

is an input parameter. Figure 6.1 shows a simplified scheme of the *point cloud* internal structure, considering a *fly-weight* approach. In short, the vectors are stored using their discretized versions instead of their original values. Thus, each charge or voltage value can be represented using a single integer or even a single byte. In addition to the *fly-weight* hash, we also use a hash whose elements are based on double-precision numerals.

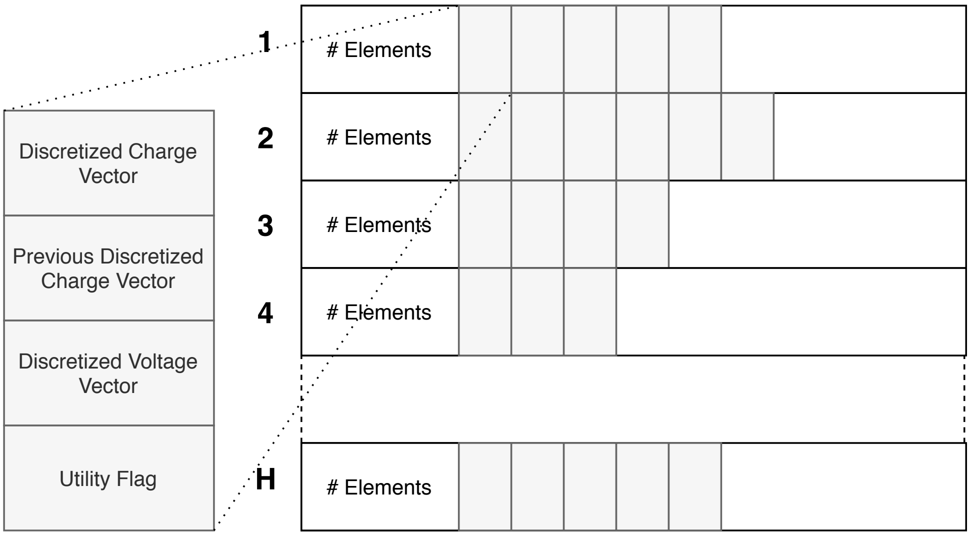
Discretized Charge Vector
Previous Discretized Charge Vector
Discretized Voltage Vector
Utility Flag
1
2
3
4
H
# Elements
# Elements
# Elements
# Elements
# Elements

Figure 6.1. Overview of the main data structure employed on both algorithms to represent cloud-based Feasible Futures.

The mapping between a *polytope* $\vec{d}$ and its corresponding bucket is based on a *polynomial rolling hash function*, as shown in Equation 6.1. The multiplier m is the first prime number such that $m \geq s$ and $gcd(m, H) = 1$. The main goal of this function, apart from being deterministic, is to generate an uniformly distributed mapping, that is, the probability of any vector $\vec{d}$ to be mapped into a determined bucket h is as close as possible to $\frac{1}{H}$.

$$h = \sum_{i=1}^{n_p} m^{i-1} d_i \tag{6.1}$$

Figure 6.2 shows a typical distribution of the elements along with the buckets (or entries) and the histogram of the number of elements in a single bucket. Both charts were generated using the same hash table, which has 1000 buckets and 5000 elements.

Since all the code is written in MATLAB environment, the hash structure was adapted to cover certain peculiarities of the language. Implementing tables whose rows have different lengths is only possible using *Cell Arrays* or *Structs*. However, the access to these structures is much more time-expensive than to regular matrices. Thus, the internal structure of the hash table was adapted to work with one main matrix, one auxiliary matrix, and one vector of integers, as illustrated by Figure 6.3.

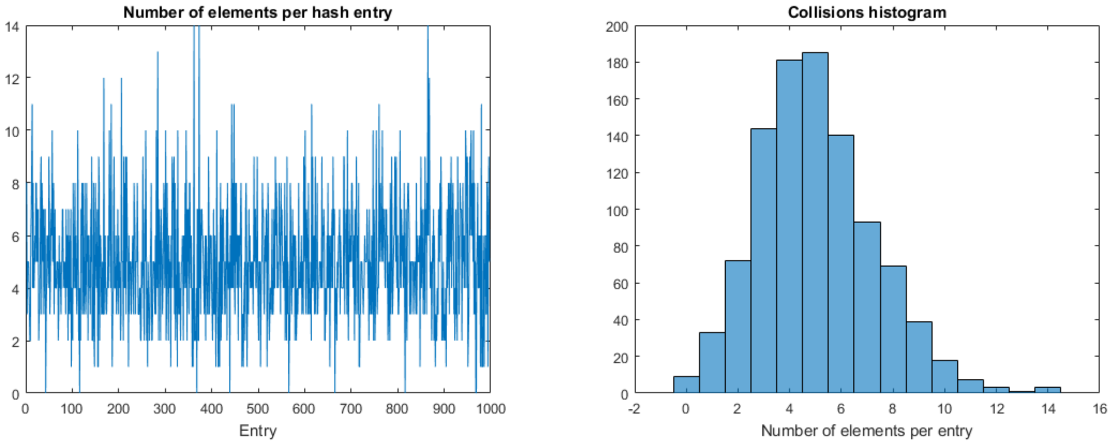

Figure 6.2. Typical data distribution in the *point cloud*.

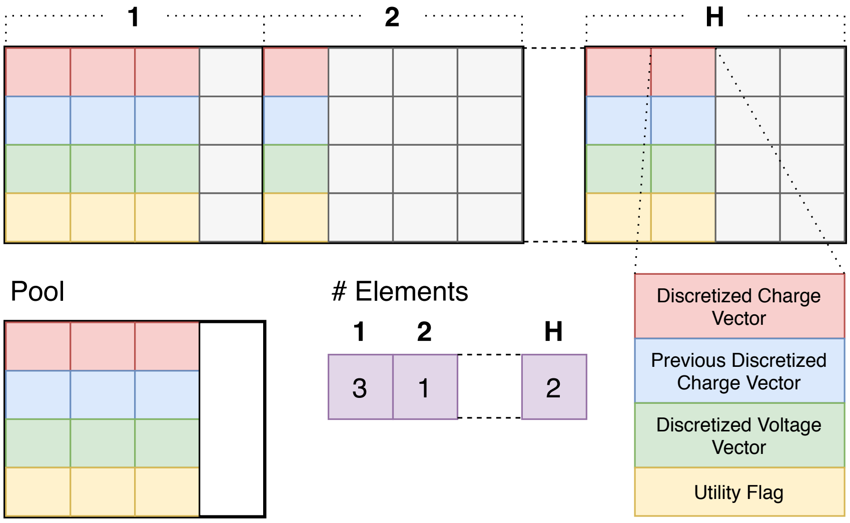
1
2
H
Pool
# Elements
1
2
H
3
1
2
Discretized Charge Vector
Previous Discretized Charge Vector
Discretized Voltage Vector
Utility Flag

Figure 6.3. Structure of the hash considering MATLAB-related issues.

Each bucket has a maximum number of elements, which are previously allocated and sequentially organized in the main matrix. If a bucket receives more data than it is able to store, the exceeding elements are stored in the pool matrix, whose length is not fixed. In particular, the pool is reallocated with double the length when required. Besides these matrices, the structure is also composed of an integer vector that stores the number of elements within each bucket, even the ones stored in the pool. The operations mentioned at the beginning of the section are implemented as follows.

- Search. The hash function is used to transform $\vec{d}$ into the bucket identifier h. Then, the length l of the h-th bucket is verified. If it is smaller than the actual size, $\vec{d}$ is searched in the first l positions of the sub-matrix. If l is larger than the actual size, $\vec{d}$ is searched in the entire sub-matrix and next in the pool. Thus, in the average case, the search has constant-time complexity, since the vast majority of buckets are limited to their sub-matrices in the main matrix. In the worst case, however, the entire pool must be verified.

- Insertion. It is assumed that the search function was called externally before insertion. Therefore, the new element can be inserted without collisions with eventual other elements within the same *polytope*. Then, the length of the corresponding bucket is incremented. If its new value is larger than the fixed size of the bucket sub-matrix, the element is appended in the pool. If the pool is full, it is reallocated with double the length. Thus, the average case of the operation is time-constant and, in the worst case, the pool is reallocated.

- Reading. The same as the search operation, but the element is returned instead of a single boolean.

- Signaling. The same as the search operation, but the utility flag of the corresponding element is toggled instead of returning anything.

- Counting. Sum of the elements of the vector of lengths. In this case, its time-complexity is $O(H)$. It may also be implemented using a "size" global variable which is incremented at each insertion. In this case, the complexity of the counting itself is constant. However, each insertion must perform an additional sum.

- Uniform sampling. If the *point cloud* is not empty, the bucket identifier is chosen as $h = \lceil h' \rceil,\ h' \sim U(0, H)$. If the corresponding bucket is empty, the identifier is incremented as $h = (h \, mod \, H) + 1$ until a bucket with $l > 0$ elements is found. Then, the element identifier is chosen as $j = \lceil j' \rceil,\ j' \sim U(0, l)$. If it is smaller then the fixed size of the bucket sub-matrix, the corresponding element is returned. Otherwise, the element is in the pool. Then, all elements in the pool receive the same probability to be chosen, regardless of the corresponding bucket. Since each element has the same probability to be chosen and the buckets are expected to have the same number of elements, in average case the choice of the returned element has a uniform distribution. The time-complexity of choosing the bucket is $O(H)$ and the sample itself is $O(1)$.

- Sequential sampling. All elements in the main matrix can be represented by their bucket index and their index inside the corresponding bucket. The first element to be returned is the index 1 from the first bucket. Then, the index is incremented at each new sample, until it reaches the fixed size of the bucket sub-matrix or the length of the bucket. The bucket index is incremented and the internal index is reset. After finishing the main matrix, the elements in the pool are sequentially returned. Each sample has constant complexity.

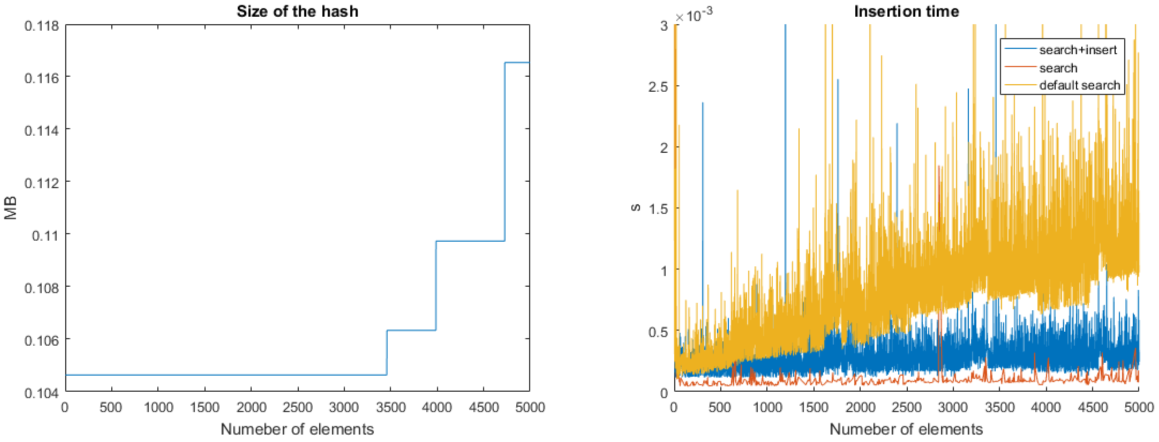

Figure 6.4. Time and space usage in the *fly-weight* cloud.

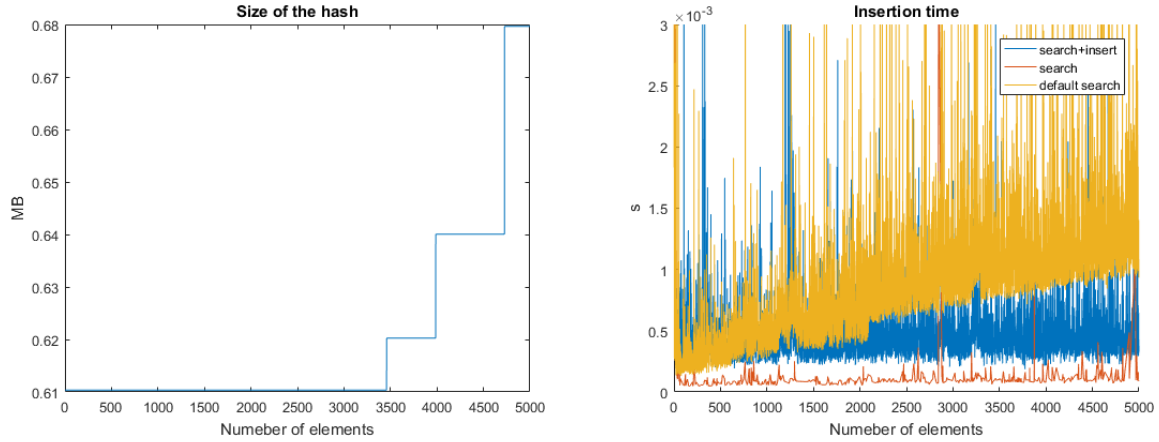

Figure 6.5. Time and space usage in the *floating-point* cloud.

Figure 6.4 shows the space-usage and processing-time data measured in the same experiment than the one from Figure 6. The fixed size of each bucket sub-matrix is 5 and the initial size of the pool is 100. For the processing-time evaluation, the *find* function, which is a MATLAB built-in function, was used as a reference and is referred to in the chart as the *default search*. Figure 6.5, in turn, shows the same results for a *floating-point* cloud. The *fly-weight* cloud is around 5 times more efficient than the *floating-point* cloud regarding memory-usage.

Chapter 7

Methodology

The experiments described in this chapter aim at evaluating the *No-Starvation MIMO Charging Problem* and *Minimum-Time MIMO Charging Problem* as computational problems, regardless of the applications on Wireless Power Transfer. Therefore, we admit the modeling from Chapter 3 for generating a large set of diverse and statistically-representative random problem instances with a guaranteed solution.

We evaluate the tractability of the problems and the capability of the proposed *dynamic-programming* algorithms to find a feasible solution. We also evaluate the efficiency regarding processing time. Finally, the three *greedy* heuristic solutions are tested as alternatives to the *dynamic-programming* algorithms.

We compare the results with three baseline approaches. The *Max-Power* algorithm is an representation of WPT algorithms based on maximizing transferred power in MIMO systems, like Yang [2017]. Thus, the algorithm maximizes the power dissipated by the resistive components of the receiving elements and limits the maximum active power dissipated by the system and the maximum amplitudes of each signal. The *MultiSpot* algorithm also maximizes the power dissipated by the resistive components of the receiving elements constrained to a maximum active power dissipated by the system, but has no limitations regarding each signal. The *Max-Sum-Of-Currents* is also compared since it can be implemented by a simple adjustment in Yang [2017] implementation, that is, using an identity matrix instead of the receiving-resistances matrix in the bilateral-quadratic objective function.

We performed all experiments using an *Intel Core i7-3537u* CPU with 2 GHz of clock-frequency and 8GB of RAM and using *MATLAB R2016a* environment. The code is available in `github.com/AlexDecker/DCCSystem`.

7.1 Random Instance Generation

First, functions $\mathcal{R}$, C, and η are defined for each device using lookup tables. For convenience, function $\mathcal{R}$ is defined using SOC values instead of the charges themselves. Thus, it is generated starting from $(SOC = 0, \mathcal{R} \sim U(0,1))$, and then iteratively by adding $(\Delta SOC \sim U(0,1), \Delta\mathcal{R} \sim U(0,1))$ into the previous values. If SOC surpasses 1, it is fixed by assigning $SOC = 1$ and then the table is considered to be ready. The other lookup tables are generated analogously, although they are guaranteed to cross the origin of the coordinates and the domain of η is guaranteed to be compatible with the image of C. Moreover, the addend limits are set in a way to guarantee the inputs are always larger than the outputs. Indeed, they represent energy conversion devices and, consequently, a larger output would imply energy creation. Finally, the domain of the η function starts from a negative value so that the discharge process is also modeled.

The maximum charge $\hat{q}_k$ for each device k is a random variable with uniform distribution $U(0, 1000)$. The upper limit of the distribution domain is arbitrary since the time variation Δt is further defined to guarantee the feasibility regarding the image of η. Let ϵ be a very small arbitrary real positive value and α_{in}, β_{in} be the parameters of a beta probability distribution, which are used to control the difficulty of the generated instances. The threshold charges, the final charges, the initial charges, and the minimum charges are respectively defined as follows

$$\underline{Q}_k = q_k(t) = (\hat{q}_k - \epsilon) \cdot b_1 + \epsilon,\ b_1 \sim \beta(\alpha_{in}, \beta_{in})$$

$$q_k(0) = (\hat{q}_k - \epsilon) \cdot u_1 + \epsilon,\ u_1 \sim U(0,1)$$

$$\underline{q}_k = (min\{q_k(0), q_k(t)\} - \epsilon) \cdot u_2 + \epsilon,\ u_2 \sim U(0,1)$$

Despite the flexibility of its shape, the beta distribution family was chosen due to its domain be between 0 and 1, which makes it easier to ensure the relaxed restrictions are still valid.

Next, the other charge vectors of the solution are defined using linear interpolation. Then, each charge value $q_k(\tau)$ is updated to $min\{max\{q_k(\tau) + z, \underline{q}_k\}, \hat{q}_k\}$, where $z \sim N\left(0, \frac{\lambda}{4} \cdot (\hat{q}_k - \underline{q}_k)\right)$. The value of the standard deviation is calculated based on the *68-95-99.7 rule*. In this case, the rule states that the probability of z to be inside the interval $[-2\sigma, 2\sigma]$ is 95%. For the experiments, we consider $\lambda = \frac{2}{t}$, which means the new value of $q_k(\tau)$ has a 95% chance to stay between the former values of $q_k(\tau - 1)$ and $q_k(\tau + 1)$, as demonstrated below. Notice that since the former values of the charges are obtained by linear interpolation between $\hat{\vec{q}}$ and $\underline{\vec{q}}$, the difference between two time-adjacent charges $q_k(\tau)$ and $q_k(\tau + 1)$ is $\frac{1}{t} \cdot (\hat{q}_k - \underline{q}_k)$.

$$\sigma = \frac{\lambda}{4} \cdot (\hat{q}_k - \underline{q}_k) \Leftrightarrow \sigma = \frac{2}{4t} \cdot (\hat{q}_k - \underline{q}_k) \Leftrightarrow 2\sigma = \frac{1}{t} \cdot (\hat{q}_k - \underline{q}_k)$$

The final charge-vector sequence of the solution is then obtained using the discretization mechanism described by Section 3.4. The charge vectors are replaced by the *centroids* of their *polytopes*, which does not invalidate $\hat{\vec{q}}$ either $\underline{\vec{q}}$. However, both initial and threshold vectors must be updated.

The time-variation constant Δt is defined in a way to ensure that all charge variations are feasible regarding the domains of the η conversion functions. So, let the interval $[\eta_k^<, \eta_k^>]$ be the image of the conversion function of the k-th device. The Δt constant is defined as follows.

$$\Delta t = max_{k,\tau} \left\{ max \left\{ \frac{1}{\eta_k^>}(q_k(\tau) - q_k(\tau - 1)); \frac{1}{\eta_k^<}(q_k(\tau) - q_k(\tau - 1)) \right\} \right\}$$

After that, the parameters within each time-slot τ are defined in a way that the charge progression of the solution is feasible. The load resistances are calculated using the $\mathcal{R}$ lookup table, $\vec{q}(\tau - 1)$, and $\hat{\vec{q}}$. The fixed resistances $\vec{r}^p$ of the passive devices are calculated *a priori* and kept constant for all slots so that the load resistances are the only resistive components that vary over time. Thus, the impedance sub-matrix $Z_R(\tau)$ is generated as in Equation 7.1. Notice that Z_R is guaranteed to be symmetrical and imaginary except by the main diagonal, as defined in Section 3.3. The beta distribution is used again due to its domain limits are between 0 and 1, which avoids very extreme and unrealistic couplings.

$$Z_R = diag(\vec{r}^p + \vec{\mathcal{R}}(\vec{q}(\tau))) - \frac{\sqrt{-1}}{2}(B + B^T), \qquad B_{ij} \sim \beta(2,2) \tag{7.1}$$

Let $y_k^<(\tau) \leq \|i_k^p\|(\tau) \leq y_k^>(\tau)$ be the interval such that $\eta(C(\|i_k^p\|(\tau)) - i_k^d(\tau)) = q_k(\tau) - q_k(\tau - 1)$. In other words, for any receiving current inside this interval, there is at least one discharge current $0 \leq i_k^d(\tau) \leq \hat{i}_k^d$ such that the resulting charge variation is equal to $q_k(\tau) - q_k(\tau - 1)$. The impedance sub-matrices $Z_T(\tau)$ and $M(\tau)$ as well as the voltage vector $\vec{v}^a(\tau)$ are calculated following the procedure from Algorithm 7, which generates a sample of possible sets of parameters and then chooses the most efficient regarding dissipated power. All lower-bound constraints are then multiplied by a factor with beta distribution $\beta(\alpha_{in}, \beta_{in})$ and all upper-bound constraints are divided by a number with the same probability distribution. These operations work as relaxations of the constraints and determine the average difficulty level of the generated instances.

The next subsections aim at explaining some sub-algorithms used for the generation.

Algorithm 7 Algorithm for generating valid impedance sub-matrices and voltage vector

```
$best_\rho \leftarrow \infty$
while $TTL > 0$ do
  $TTL \leftarrow TTL - 1$
  $\vec{i^p} \leftarrow$ any current such that $\vec{y^<} \leq \left\|\vec{i^p}\right\| \leq \vec{y^>}$
  generate $M$ and $\vec{i^a}$ such that $M\vec{i^a} + Z_R\vec{i^p} = 0$
  generate $Z_T$ such that $V = Z_T\vec{i^a} + M^T\vec{i^p}$ is real
  $\rho \leftarrow (\vec{v^a})^T Re\{\vec{i^a}\}$
  if $\rho < best_\rho$ then
    $best_\rho \leftarrow \rho$
    $best_v \leftarrow \vec{v^a}$
    $best_{zt} \leftarrow Z_T$
    $best_m \leftarrow M$
  end if
end while
return $best_\rho$, $best_v$, $best_{zt}$, $best_m$
```

7.1.1 Generation of the coupling matrix

The generation of $M(\tau)$ occurs as follows. If $n_a \geq n_p$, it is randomly generated as $M(\tau) = \sqrt{-1}\mathcal{M}$ such that $\mathcal{M}_{ij} \sim \beta(2,2)$ and all rows are linearly independent. If $n_a < n_p$, on the other hand, there are more rows than columns and, therefore, they cannot be linearly independent. Thus, $M(\tau)$ must be chosen in a way that $M(\tau)M(\tau)^+ \, Z_R(\tau) \, \vec{i^p}(\tau) = Z_R(\tau)\vec{i^p}(\tau)$ (see Equation 3.6).

Let $\mathcal{M}$ be the first n_a rows of $M(\tau)$. Let R be the first n_a rows of $Z_R(\tau)$. $\mathcal{M}$ can also be generated by random beta sampling in a way to all rows to be linearly independent. Thus, $\mathcal{M}$ is invertible and the currents of the active circuits are obtained by

$$\vec{i^a}(\tau) = -\mathcal{M}^{-1} \, R \, \vec{i^p}(\tau)$$

Rows $n_a + 1$ to n_p, in turn, must be imaginary (see Section 3.3) and such that

$$m_i\vec{i^a} = -z_i\vec{i^p}$$

where m_i is the i-th row of $M(\tau)$ and m_i is the i-th row of $Z_R(\tau)$. Thus, the following equalities must be respected in order to m_i to be imaginary.

$$\begin{cases} m_i \, Re\{\vec{i^a}\} = -\sqrt{-1} \, Im\{z_i\vec{i^p}\} \\ \sqrt{-1} \, m_i \, Im\{\vec{i^a}\} = -Re\{z_i\vec{i^p}\} \end{cases}$$

Which is equivalent to

$$\begin{bmatrix} Re\{\vec{i^a}\} \\ Im\{\vec{i^a}\} \end{bmatrix} m_i^T = \sqrt{-1} \begin{bmatrix} -Im\{z_i \vec{i^p}\} \\ Re\{z_i \vec{i^p}\} \end{bmatrix}$$

Assuming $n_a \geq 2$ and $Re\{\vec{i^a}\}$ and $Im\{\vec{i^a}\}$ are linearly independent, this equation has infinite solutions. These can be obtained using the *pseudo-inverse*, which is guaranteed to be real. Since the right side is guaranteed to be imaginary, so is m_i.

7.1.2 Generation of the transmitting-impedance matrix

The generation of $M(\tau)$ creates $\vec{i^a}$ as by-product. Thus, Z_T must be generated so that $\vec{v^a} = Z_T \vec{i^a} + M^T \vec{i^p}$ be real. Moreover, according to Section 3.3, Z_T must be symmetrical and imaginary except by the main diagonal. The method described in the following paragraphs is summarized by Algorithm 8.

Algorithm 8 Algorithm for generating a valid transmitting-impedance matrix

```
$\vec{r^a} \leftarrow -\vec{1}$
$M^a \leftarrow \frac{\sqrt{-1}}{2}(B + B^T), \qquad B_{ij} \sim \beta(2,2)$
while true do
  $\vec{x} \leftarrow \sqrt{-1}\, M^a\, Re\{\vec{i^a}\} - Im\{M^T\, \vec{i^p}\}$
  $\vec{r^a} \leftarrow \vec{x} \oslash Im\{\vec{i^a}\}$
  if $r_i^a > 0 \;\forall\; i$ then
    break
  else
    $k \leftarrow arg\, min\{\vec{r^a}\}$
    $\Delta m \leftarrow \sqrt{-1}\frac{x_k}{Re\{i_k^a\}} - \sqrt{-1}\, sign(Im\{i_k^a\}) \cdot sign(Re\{i_k^a\}) \cdot (g + \epsilon) \quad g \sim \gamma(1,2)$
    $M_{k,k}^a \leftarrow M_{k,k}^a + \Delta m$
  end if
end while
$Z_T \leftarrow diag(\vec{r^a}) + M^a$
return $Z_T$
```

According to Equation 3.5, the voltage vector $\vec{v^a}$, the transmitting impedance submatrix Z_T, the coupling matrix M, the transmitting currents $\vec{i^a}$, and the receiving currents $\vec{i^p}$ are related as follows.

$$\vec{v^a} = Z_T \vec{i^a} + M^T \vec{i^p}$$

Let Z_T be decomposed as $Z_T = diag(\vec{r^a}) - M^a$, where $\vec{r^a}$ is the resistance vector, which is real, and M^a is the matrix with the couplings between transmitting coils, which are all imaginary. Thus, the imaginary part of $\vec{v^a}$ is expressed by

$$Im\{\vec{v^a}\} = diag(\vec{r^a}) Im\{\vec{i^a}\} - \sqrt{1} M^a Re\{\vec{i^a}\} + Im\{M^T \vec{i^p}\}$$

Let $\vec{x} \triangleq \sqrt{1}M^a Re\{\vec{i^a}\} - Im\{M^T\vec{i^p}\}$. For $Im\{\vec{v^a}\}$ to be zero, $diag(\vec{r^a})Im\{\vec{i^a}\} = \vec{x}$. In other words, admitting $Im\{i_k^a\} \neq 0$, the following must be respected.

$$\vec{r^a} = \vec{x} \oslash Im\{\vec{i^a}\}$$

The resulting $\vec{r^a}$ is guaranteed to be real, but its values must be also positive since it represents resistances. Thus, let r_k^a be such that $r_k^a \leq 0$. One might add an imaginary Δm to M_{kk}^a so that (i) M^a stay imaginary, (ii) M^a stay symmetrical, and (iii) only r_k^a is affected in $\vec{r^a}$. Thus, r_k^a becomes

$$r_k^a = \frac{x_k + \sqrt{-1}\Delta m Re\{i_k^a\}}{Im\{i_k^a\}}$$

Thus, for r_k^a to be greater than zero, admitting $Re\{i_k^a\} \neq 0$, there are two cases, depending on the signal of $\frac{Re\{i_k^a\}}{Im\{i_k^a\}}$.

$$\begin{cases} \Delta m < \sqrt{-1}\dfrac{x_k}{Re\{i_k^a\}}, & \dfrac{Re\{i_k^a\}}{Im\{i_k^a\}} > 0 \\ \Delta m > \sqrt{-1}\dfrac{x_k}{Re\{i_k^a\}}, & \dfrac{Re\{i_k^a\}}{Im\{i_k^a\}} < 0 \end{cases}$$

And, since $g \sim \gamma(1,2)$ is always real and non-negative, the following formula is guaranteed to create a valid r_k^a without disturbing the other active resistances. Consider that ϵ is a very small real positive constant.

$$\Delta m = \sqrt{-1}\frac{x_k}{Re\{i_k^a\}} - \sqrt{-1}\, sign(Im\{i_k^a\}) \cdot sign(Re\{i_k^a\}) \cdot (g + \epsilon)$$

7.2 Statistical Analysis Description

This section aims to describe the statistical tools employed in the experiments. In short, we use

- *Pearson Correlation* [Benesty et al., 2009] to test linear dependency between a measured quantity and an input parameter, when both are continuous or the discretization interval is small enough. The correlation varies between -1 and 1, where -1 means complete inverse correlation, 0 means independence and 1 means complete correlation. This statistic is used to testify linearity between the number of time-slots and the execution time.

- *Kruskal-Wallis H Test* to test overall dependency between a continuously measured quantity and a discrete input parameter. The null hypothesis of the test states that the probability distributions of the measured data for all values of the discrete variable are all the same. Thus, a small *p-value* means probable dependence between the quantities of interest. This statistic is used to testify the dependence between the input parameters and the execution time.

- *Chi-Squared Test for Independence* to testify independence between a discrete measured quantity and a discrete input parameter. Analogously to *Kruskal-Wallis H Test*, a small p-value means probable dependence between the quantities of interest. This statistic is used to testify the dependence between the input parameters and effectiveness.

- *Binomial confidence interval with normal approximation*, as known as the *Wald* method for a binomial confidence interval, to compare the effectiveness between algorithms for the *No-Starvation MIMO Charging Problem.* Let p be the sample probability of success, that is, the ratio between the number of successes n and the number of experiments m. The confidence interval for a given significance level α is as follows.

$$p \pm z_{(1-\alpha/2)}\sqrt{\frac{p\cdot(1-p)}{m}}$$

- *Hypothesis test for comparison between two binomial random variables.* Let n_1 be the number of successes for m_1 instances using algorithm 1. Let n_2, m_2 be analogous to algorithm 2. Let p_1, p_2 be respectively the success probabilities of algorithms 1 and 2. The *null-hypothesis* states that $p_1 = p_2$ while the *alternative-hypothesis* states that $p_1 > p_2$. Thus, rejecting the *null-hypothesis* means that Algorithm 1 can be considered as more effective than Algorithm 2. Let $p_1' = \frac{n_1}{m_1}$ and $p_2' = \frac{n_2}{m_2}$ be maximum likelihood estimators for p_1 and p_2. The probability distributions of p_1' and p_2' are approximately normal and are described as follows.

$$p_1' \sim N\left(p_1, \frac{p_1\cdot(1-p_1)}{m_1}\right) \qquad p_2' \sim N\left(p_2, \frac{p_2\cdot(1-p_2)}{m_2}\right)$$

Thus, the probability distribution of their difference is as follows

$$p_1' - p_2' \sim N\left(p_1 - p_2, \frac{p_1\cdot(1-p_1)}{m_1} + \frac{p_2\cdot(1-p_2)}{m_2}\right)$$

By standardizing the distribution, it follows that

$$z = \frac{p_1' - p_2'}{\sqrt{\frac{p_1'\cdot(1-p_1')}{m_1} + \frac{p_2'\cdot(1-p_2')}{m_2}}} \sim N(0,1)$$

And, therefore, the corresponding *p-value* is the probability

$$\mathcal{P}(Z > z \,|\, Z \sim N(0,1))$$

- *t-Student Confidence Interval* is used to compare execution times between different algorithms.

Besides the statistical methods used for comparing the results, let us also define the *Expected Relative Lifetime* (ERL) score, which measures the difficulty of a given problem instance to be solved using a randomized algorithm. The score varies between 0 (very difficult) and 1 (very easy).

Definition 7.2.1. *The Expected Relative Lifetime (ERL) score of a problem instance is the normalized expected index of the time-slot where the charge of one of the devices will drop below the minimum $\underline{\vec{q}}$ for the first time, considering the Random-Feasible-Future Algorithm (see Section 5.4.2.4). Thus, let τ_j be the time-slot where the j-th execution of the RFF Algorithm failed due to the minimum charge constraint. Let λ_j be an indicator variable which is 1 if the solution solves the problem and 0 otherwise. Let t be the total number of time-slots of the instance. The ERL score* **e** *considering 1000 repetitions is defined as*

$$e = \frac{1}{1000 \cdot (t+1)} \left(\sum_{j=1}^{1000} (\tau_j + \lambda_j) \right)$$

Chapter 8

Experimental Results

We employed the six proposed algorithms to solve randomly generated instances of both *No-Starvation MIMO Charging* and *Minimum-Time MIMO Charging* problems. The parameters chosen for each algorithm are summarized in Table 8.1. The time-to-leave value for the greedy algorithms is a large arbitrary number. The parameters chosen for the Dynamic-Programming-based algorithms were chosen in a way to approach their execution times. n_k is the number of samples of the k multiplier for each pair $(\vec{q}_0, \vec{\mathbf{v}}^a)$ in the exploration method.

8.1 No-Starvation MIMO Charging Problem: easy instances

The first set of experiments used "easy instances", that is, instances generated using $\alpha_{in} = 3$ and $\beta_{in} = 2$ input parameters. Figure 8.1 shows the histogram of a sample of 10000 values generated using the beta distribution and the aforementioned parameters. The instance generator uses these values to relax the constraints, resulting in a set of problem instances whose *ERL scores* are described by the second histogram.

Figure 8.2 shows the success ratio of each considered algorithm for the generated instances, that is, the ratio between the number of successes and the total number of experiments. Each algorithm attempted to solve the same 100 instances. The confidence intervals were calculated using the *binomial distribution with normal approximation* (see Section 7.2) with a 10% significance level.

For almost all cases, *Simple*, *Pareto*, *Max-Sum*, and *Max-Min* algorithms remained statistically equivalent. However, considering all instances, the *Pareto* algorithm achieved a 97% success ratio against the 92% success ratio achieved by the *Max-Sum* algorithm. Thus, the corresponding *p-value* for equivalence against *Pareto* superiority results in

Table 8.1. The considered parameters for each algorithm.

	TTL_1	TTL_2	TRH	n_k
Simple	250	80	100	5
Pareto	750	60	250	-
Fly-Weight	250	40	100	5
Max-Sum	10000	-	-	-
Max-Min	10000	-	-	-
Max-Sum-Of-Currents	10000	-	-	-

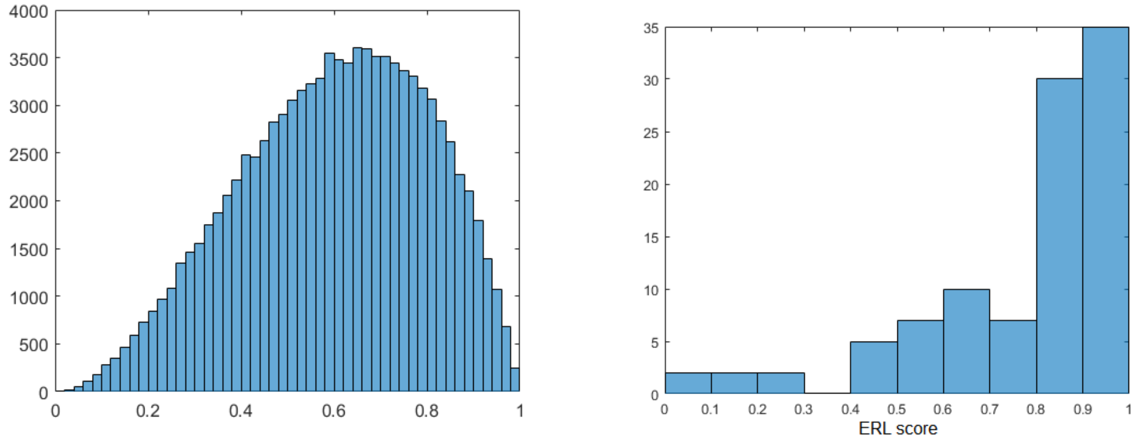

Figure 8.1. Beta distribution with $\alpha_{in} = 3$ and $\beta_{in} = 2$ and the corresponding ERL score histogram.

0.059354, which can be rejected for a 10% significance level. Consequently, the *Pareto* algorithm is more effective than *Max-Min* and *Fly-weight* algorithms for considered instances.

The *Fly-weight* algorithm, in turn, achieved a significantly less success ratio than the other ones. In particular, *Simple* and *Pareto* algorithms achieved a larger success ratio than the greedy ones for instances with two active circuits. Besides that, the sample means of *Simple* and *Pareto* algorithms moved away from the greedy ones with the increase in the number of passive devices. Indeed, both statistically surpassed the *Max-Min* algorithm for instances with four passive devices.

Table 8.2 summarizes the calculated *p-values* of the *chi-squared tests for independence* (see Section 7.2) between each input parameter and the successes of the considered algorithms. For a 10% significance level, the success of *Max-Sum*, *Max-Min*, and *Fly-Weight* algorithms can be considered as being dependent on the number of passive circuits. Roughly, their success ratio decreases as the number of passive devices increase, which might be resulting from the enlargement of the charge vector space. The other algorithms do not show sufficient evidence of dependence on the number of passive circuits.

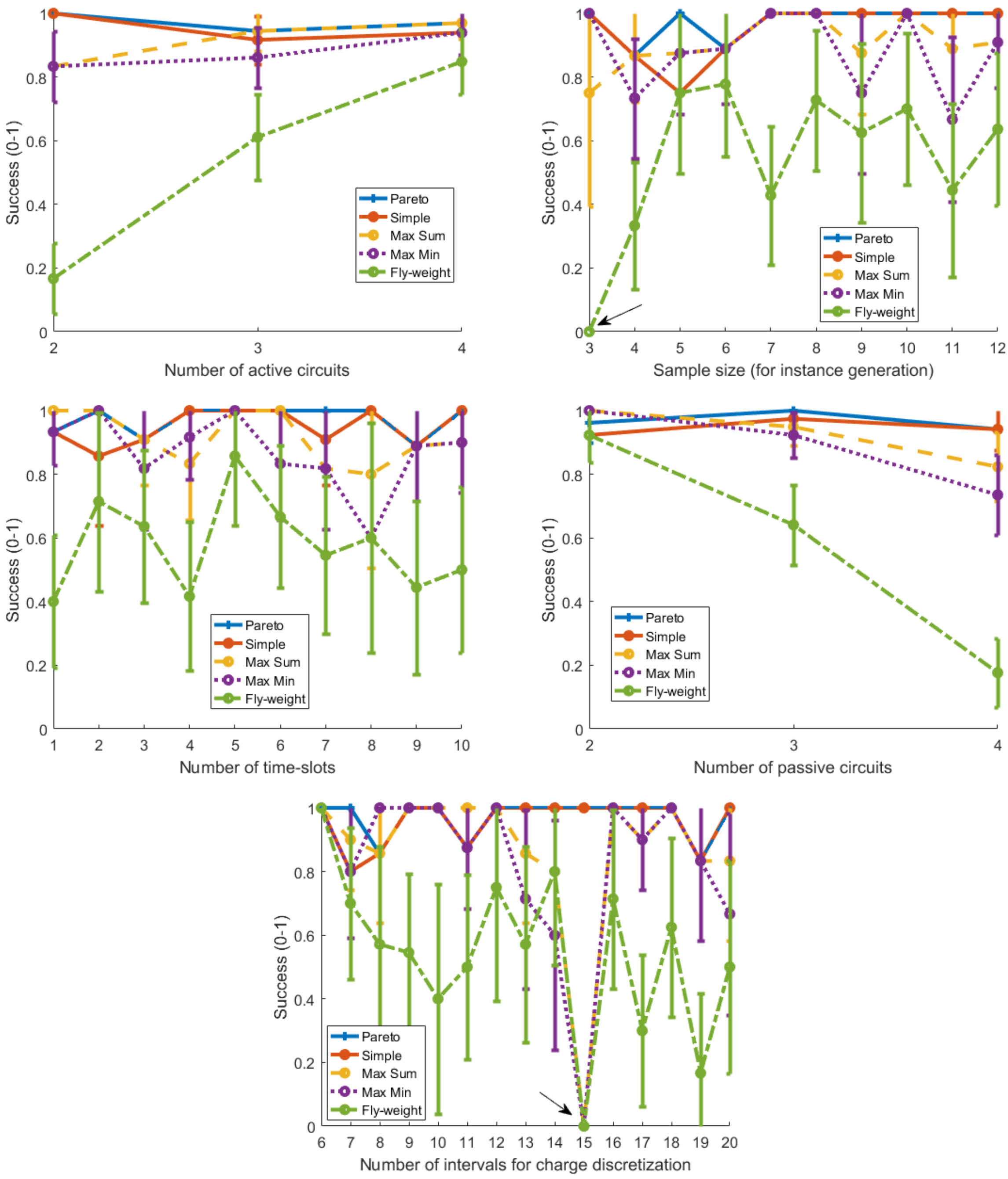

Figure 8.2. The success ratio for some input parameters with $\alpha_{in} = 3, \beta_{in} = 2$.

In addition, the *Fly-Weight* algorithm can be considered dependent on the number of active circuits. Roughly, its success ratio increases as the number of passive devices increase, which might result from the enlargement of the active-voltage vector space. Thus, more voltage vectors generate the same receiving-current vector and, therefore, the exploitation method is facilitated.

The *Max-Min* algorithm can also be considered dependent on the number s of charge

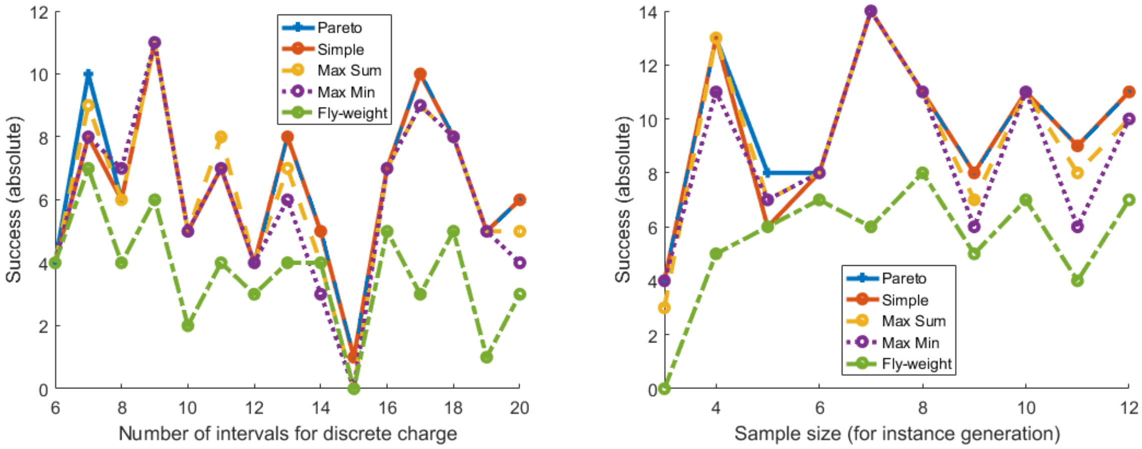

Figure 8.3. Absolute number of successes for some input parameters with $\alpha_{in} = 3, \beta_{in} = 2$.

Table 8.2. P-values for the *Chi-Squared* independence test between input parameters and the number of successes of each considered algorithm.

	Pareto	**Simple**	**Max-Sum**	**Max-Min**	**Fly-weight**
n_p	0.34	0.63	0.036	0.005	2.1e-8
n_a	0.42	0.29	0.1	0.38	2.5e-8
t	0.72	0.81	0.58	0.62	0.69
s	0.61	0.58	0.18	0.08	0.21
nSamples	0.36	0.15	0.74	0.13	0.14

Table 8.3. P-values for the *Kruscal-Wallis* independence test between input parameters and the execution times of each considered algorithm.

	Pareto	**Simple**	**Max-Sum**	**Max-Min**	**Fly-weight**
n_p	0.45	0.29	0.42	0.50	0.05
n_a	0.22	0.39	0.28	0.29	0.42
t	3e-14	1e-13	1e-13	1e-13	5e-4
s	0.07	0.11	0.11	0.09	0.51

discretization intervals, although this statistic may be negatively influenced by the degenerated point $s = 15$ (indicated with the arrow in Figure 8.2). The same degeneration also appears in the *sample size* chart from Figure 8.2, which is also indicated by an arrow. Figure 8.3 shows the absolute number of successes instead of the success ratio. Notice that the number of instances with $s = 15$ and with $nSample = 3$ is very small when compared to other scenarios.

Figure 8.4 shows the execution times towards the same experiments from Figure 8.2. The confidence intervals are based on t-student distribution with 99 degrees-of-freedom and a 10% significance level. As expected, the dynamic-programming-based algorithms are clearly more time-expensive than the greedy ones since the last ones consider only

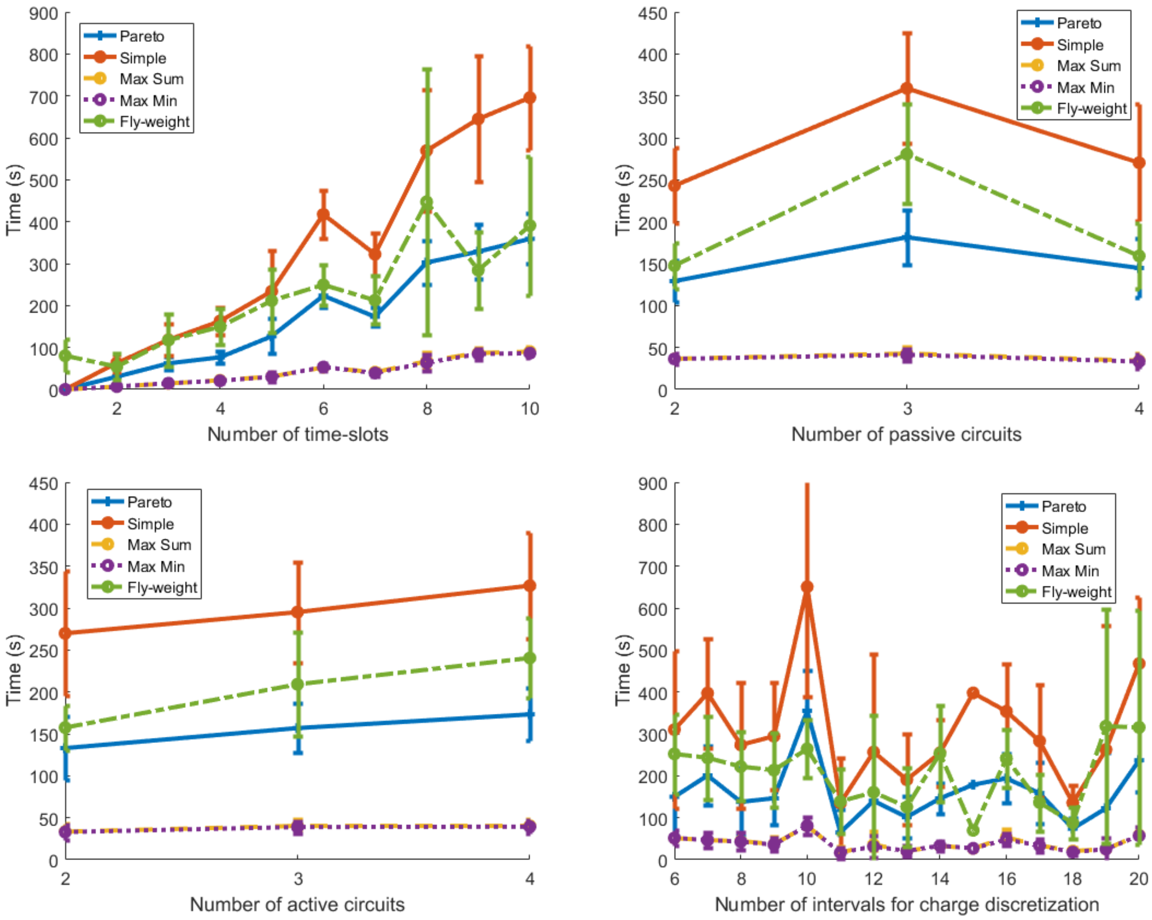

Figure 8.4. Execution times for some input parameters with $\alpha_{in} = 3, \beta_{in} = 2$.

Table 8.4. *Pearson* correlation values between input parameters and the execution times of each considered algorithm.

	Pareto	Simple	Max-Sum	Max-Min	Fly-weight
n_p	0.03	0.02	-0.03	-0.04	-0.003
n_a	0.11	0.08	0.08	0.07	0.15
t	0.84	0.81	0.85	0.84	0.46
s	-0.05	-0.06	-0.1	-0.1	-0.02

one previous-charge vector for each time-slot. From Table 8.3 and considering a 10% significance level, one might conclude that the execution-time for all considered algorithms is highly dependent on the number of time-slots. Furthermore, the values present in Table 8.4 state that the *pearson correlation* between the execution-times of all algorithms and the number of time-slots is significant, which indicates a probable linear relation. The only exception is the *Fly-Weight* algorithm, whose correlations are probably prejudiced by the high number of failures.

8.2 No-Starvation MIMO Charging Problem: hard instances

The next set of experiments used “hard instances”, that is, instances generated using $\alpha_{in} = 5$ and $\beta_{in} = 0.5$ input parameters. Figure 8.5 shows the histogram of a sample of 10000 values generated using the beta distribution and the aforementioned parameters. Similarly to the “easy instances”, the second histogram describes the *ERL scores* resulting from the new distribution of the relaxation factors.

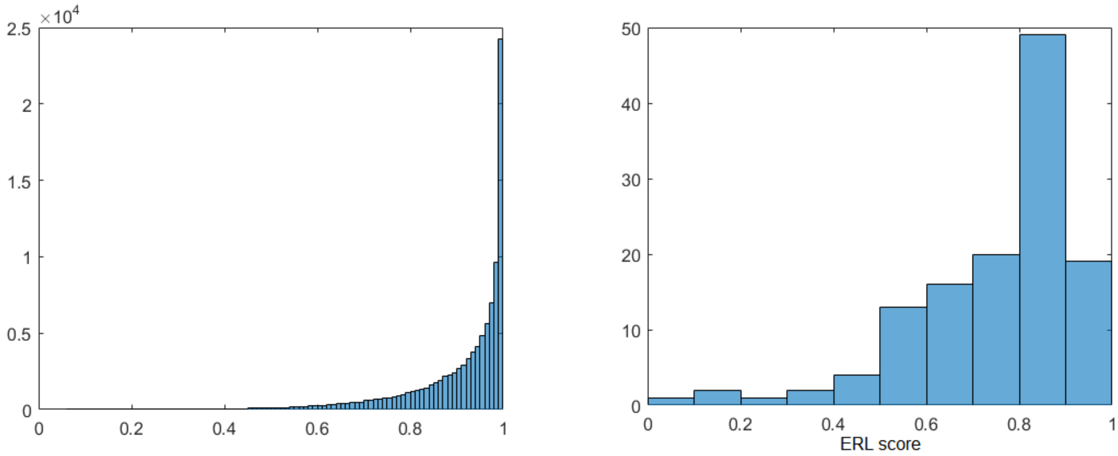

Figure 8.5. Beta distribution with $\alpha_{in} = 5$ and $\beta_{in} = 0.5$ and the corresponding ERL score histogram.

We chose the *Pareto* algorithm as a representative of the dynamic-programming-based methods due to the good balance between execution-time and effectiveness. The resulting success ratios of the *Pareto*, *Max-Sum* and *Max-Min* algorithms for the same 128 instances are shown in Figure 8.6. Over again, the error bars are based on *binomial confidence-intervals with normal approximation* and 10% significance level.

Notice that, unlike for the “easy instances”, the *Max-Sum* algorithm is often more effective than the *Pareto* algorithm. However, the mean difference occurs because of the instances with a small number of intervals for charge discretization. The p-value of the chi-squared test for independence between the successes of *Pareto* and the number of intervals is 0.092652, which indicates dependence for a 10% significance level. Indeed, the mean effectiveness increases as the number of intervals increases, since the assumption of equivalence between states within the same *polytope* becomes more accurate.

Thus, considering all instances where the number of intervals is more than 15, the *Pareto* algorithm achieved an 89% success ratio against the 77% success ratio achieved by the *Max-Sum* algorithm. Thus, the corresponding *p-value* for equivalence against *Pareto* superiority results in 0.09968, which can be rejected for a 10% significance level.

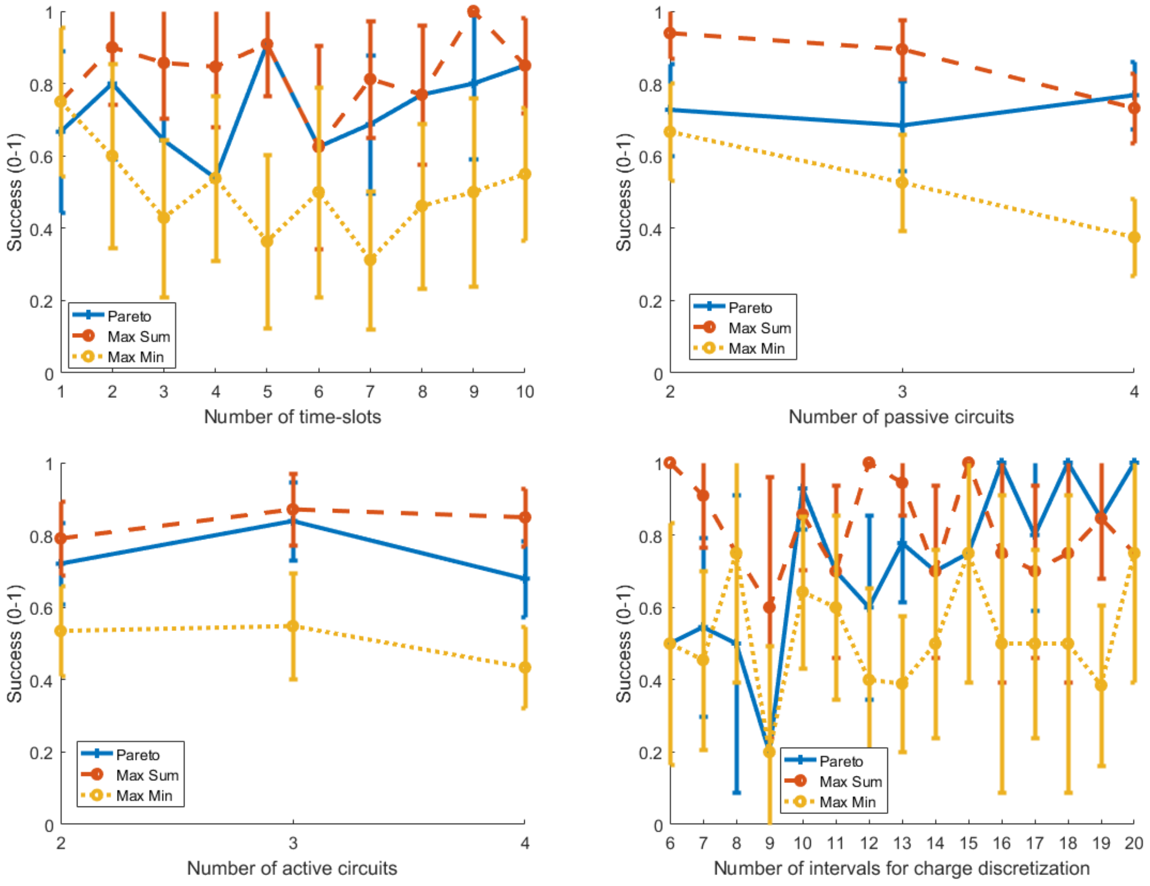

Figure 8.6. Normalized number of successes for some input parameters with $\alpha_{in} = 5, \beta_{in} = 0.5$.

Consequently, for the chosen significance level, the *Pareto* algorithm is also more effective than the *Max-Min* algorithm.

Figure 8.7 shows the execution times for the same experiments from Figure 8.6. *Kruskal-Wallis tests for independence* indicated that the execution time of all algorithms is dependent on the number of time-slots since all calculated p-values resulted in values of less than 10^{-10}. Moreover, the Pearson correlations for *Pareto*, *Max-Sum*, and *Max-Min* algorithms are respectively 0.76, 0.78, and 0.75, which indicated a linear tendency.

The other parameters do not have enough evidence to be considered as influencing the execution time. This indicates that the generation of feasible futures using the *Pareto* algorithm usually ends due to time-to-leave conditions. Thus, the algorithm would require milder time limitations to fill sufficient *polytopes* in the charge space.

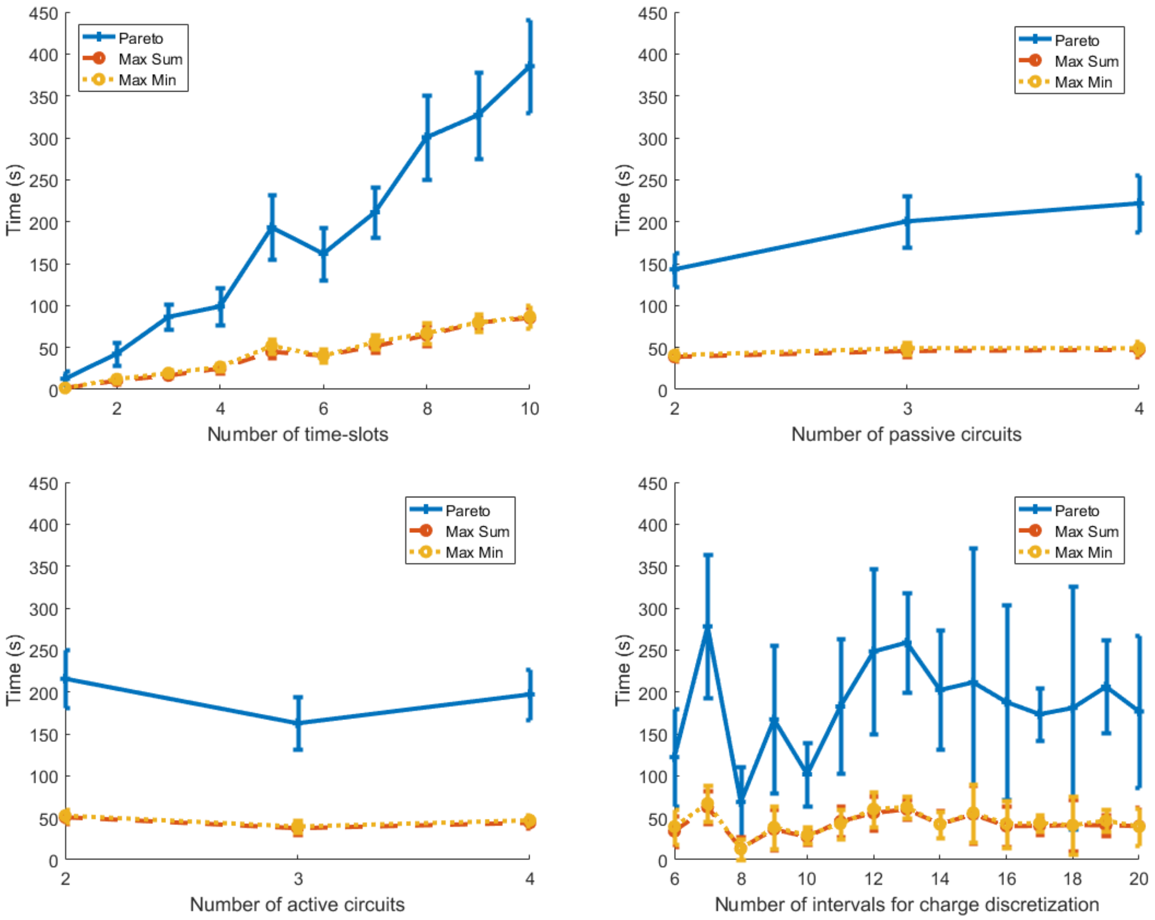

Figure 8.7. Execution times for some input parameters with $\alpha_{in} = 5, \beta_{in} = 0.5$.

8.3 No-Starvation MIMO Charging Problem: Comparison with baselines

Figure 8.8 evaluates the greedy algorithms using 350 "hard instances". Analogously to the other experiments involving the success ratio, we calculated the confidence intervals using a *binomial distribution with normal approximation*. The mean success ratio of the *Max-Sum-Of-Currents* and the *Max-Power* algorithms achieved the midterm between the *Max-Sum* and the *Max-Min*, though they was statistically equivalent to the *Max-Sum* algorithm in each case separately and to *Max-Min* algorithm for some punctual cases. Considering all instances, the *Max-Sum* algorithm achieved a 74% success ratio against the 69% success ratio achieved by the *Max-Sum-Of-Currents* algorithm and 70% achieved by the *Max-Power* algorithm.

Thus, the corresponding *p-value* for equivalence between *Max-Sum* and *Max-Sum-Of-Currents* against *Max-Sum* superiority results in 0.065137, which can be rejected for a 10% significance level. Furthermore, the *p-value* for equivalence between *Max-Sum* and

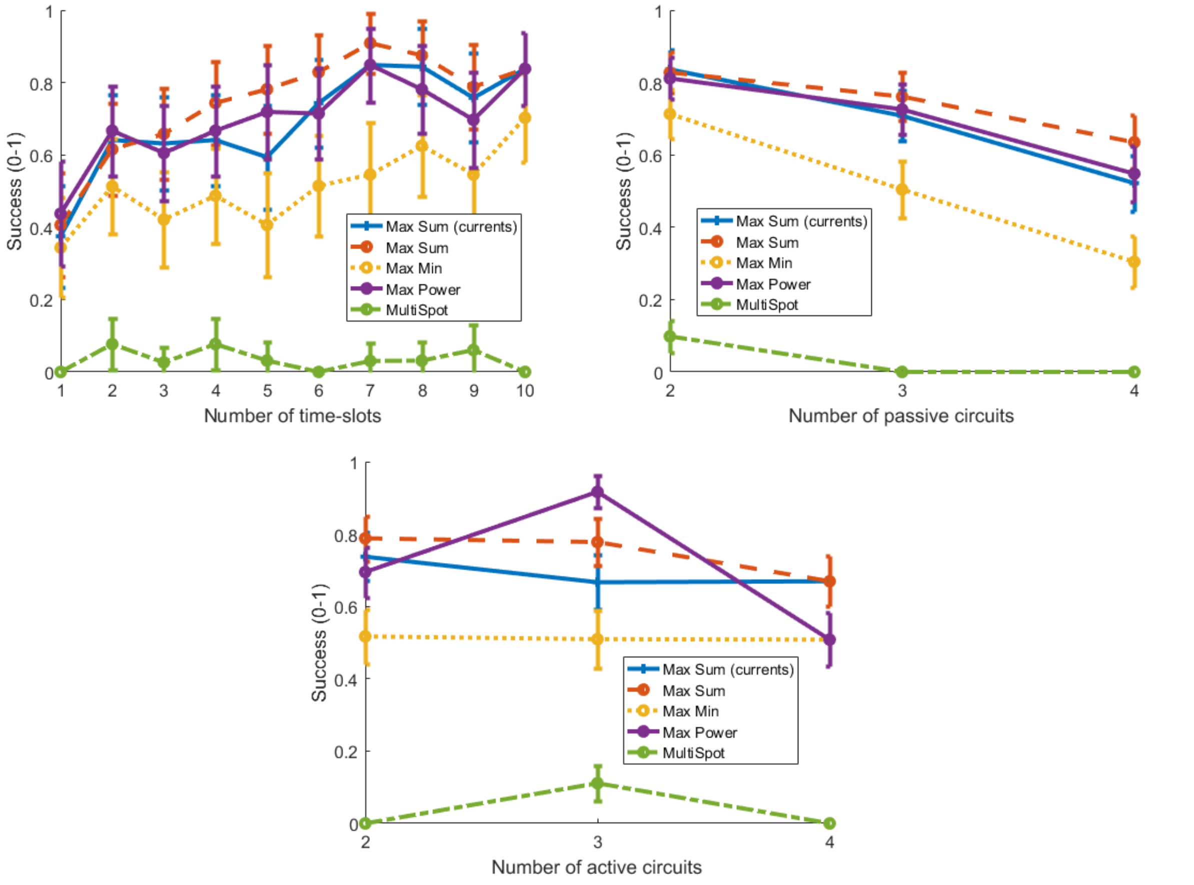

Figure 8.8. Success ratios for some input parameters with $\alpha_{in} = 5, \beta_{in} = 0.5$, and only the greedy algorithms.

Max-Power against *Max-Sum* superiority results in 0.08873, which can also be rejected for a 10% significance level. Consequently, for the chosen significance level, the *Max-Sum* algorithm is also more effective than the *Max-Sum-Of-Currents* and *Max-Power* algorithms. The *MultiSpot* algorithm, in turn, achieved poor performance, as it disregards the maximum signal amplitude restrictions.

Table 8.5 shows the p-values for *chi-squared* independence tests between the input parameters and the successes of the algorithms. Even for a 1% significance level, the success of all algorithms can be considered dependent on the number of passive devices. There was no evidence enough to state dependence concerning the number of active devices. Moreover, only *Max-Sum*, *Max-Sum-Of-Currents* and *Max-Power* can be considered statistically as dependent on the number of time-slots.

Table 8.5. P-values for the *chi-squared* independence test between input parameters and the successes of each considered algorithm.

	n_a	n_p	t
Max-Sum	0.27	1e-6	0.0004
Max-Min	0.49	2e-9	0.13
Max-Sum-Of-Currents	0.61	0.002	6e-5
Max-Power	0.29	4e-5	0.02
MultiSpot	0.12	9e-6	0.46

Table 8.6. P-values for the Kruskal-Wallis independence test between input parameters and the normalized charging times of each considered algorithm.

	Pareto	**Simple**	**Max-Sum**	**Max-Min**	**Fly-weight**
n_p	0.03	0.03	0.01	0.02	0.81
n_a	0.78	0.91	0.67	0.84	2e-3
t	0.02	0.03	0.01	0.02	0.22
s	0.26	0.22	0.31	0.30	0.19
nSamples	0.55	0.52	0.57	0.48	0.59

8.4 Minimum-Time MIMO Charging Problem

The main metric chosen to evaluate the algorithms for the *Minimum-Time MIMO Charging Problem* was the normalized charging time, that is, the number of time-slots each solution took to reach the threshold charge vector divided by the maximum number of time-slots of the problem instance. Thus, the resulting values vary between 0 and 1, where 0 means the initial state of the problem instance is already a valid solution and 1 means all time-slots were required to charge the devices.

Figure 8.9 shows the means of the normalized charging times and the *t-Student* confidence intervals calculated similarly to the ones from Figure 8.4. The results for all considered algorithms were statistically equivalent, except by some cases where the *Fly-Weight* algorithm achieved worse charging times when compared to the others. The *Kruskal-Wallis* independence test indicates that, even for a 5% significance level, the charging times are dependent on the number of passive devices and the maximum number of time-slots, as illustrated by Table 8.6. The effectiveness of the *Fly-weight* algorithm is again affected by the small success ratio. The only input parameter which has enough evidence to be considered correlated to the charging time is the number of active circuits. Indeed, the increasing of the number of active circuits favors the exploitation method.

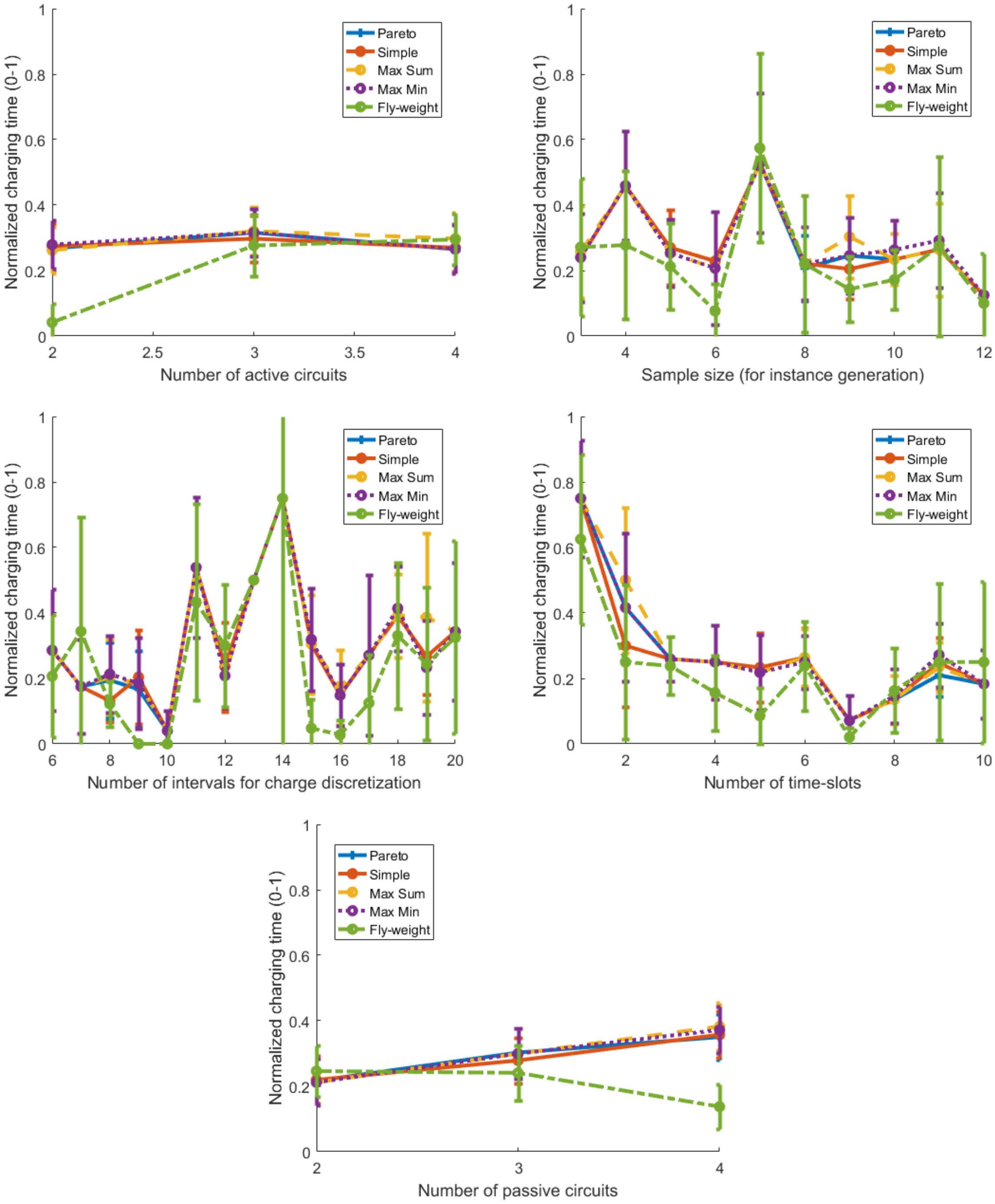

Figure 8.9. Normalized charging times.

8.5 Minimum-Time MIMO Charging Problem: Comparison with baselines

Since the previous experiments indicated no evidence of a difference between the proposed algorithms regarding the normalized charging times, some were omitted in the next

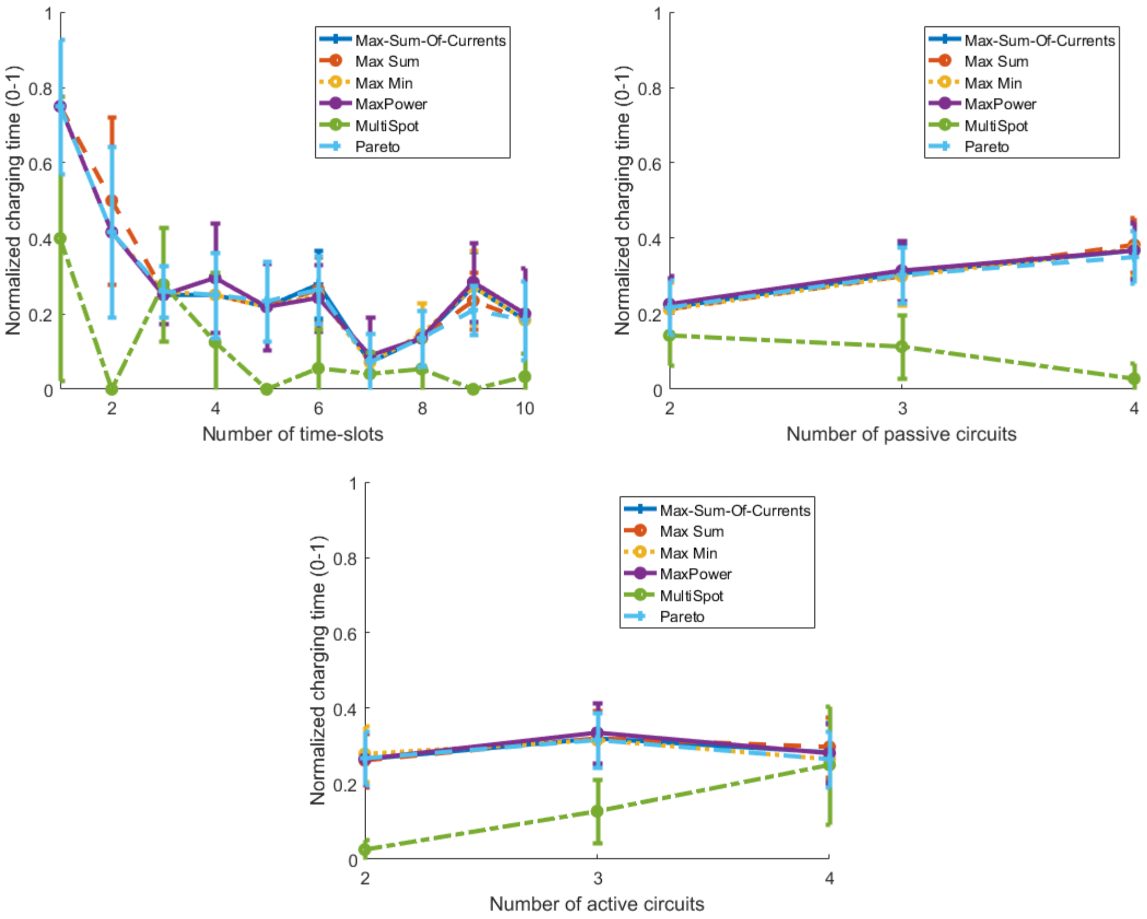

Figure 8.10. Comparison of the normalized charging times considering the proposed algorithms and the baselines.

experiments for sake of clarity. Thus, besides the greedy algorithms *Max-Sum* and *Max-Min* and the baseline algorithms *Max-Power*, *MultiSpot* and *Max-Sum-Of-Currents*, we considered only the *Pareto* algorithm. Indeed, the *Pareto* algorithm achieved the best results among the dynamic-programming ones.

Figure 8.10 shows the normalized charging time results considering only "easy instances". Most algorithms can be considered equivalent regarding the charging time, but *MultiSpot* required significantly less time to charge the devices in the average case. However, this does not mean that *MultiSpot* was more effective than the other algorithms. *MultiSpot* obtained a solution in 45 out of 99 tested instances, while *Max-Sum-Of-Currents* obtained 95 solutions, *Max-Power* obtained 94, *Max-Sum* obtained 97, *Max-Min* obtained 96, and Pareto obtained 97. Thus, Figure 8.10 counts only the instances for which *MultiSpot* obtained success, that is, the easiest instances. Figure 8.11 shows the normalized charging time data considering only the instances where *MultiSpot* was successful.

Analogously to the experiments from Section 8.5, *MultiSpot* is prejudiced by not

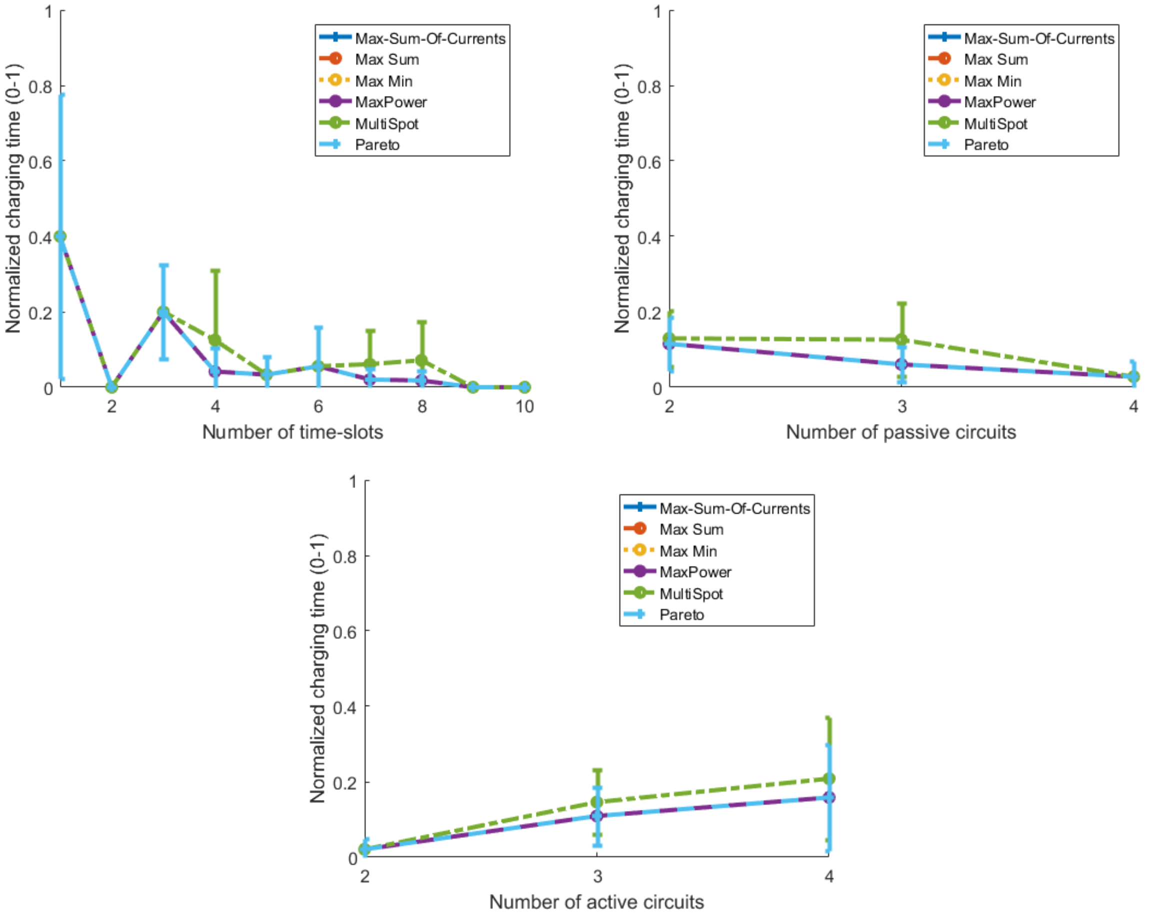

Figure 8.11. Comparison of the normalized charging times considering only instances for which *MultiSpot* was successful.

considering amplitude limitations. Indeed, the test for equivalence between *MultiSpot* and *Max-Power*, which has the second worst average, has a *p-value* of $7 \cdot 10^{-20}$, which can be rejected even for a extremely low significance level. The other algorithms cannot be considered different regarding effectiveness since the corresponding p-values are all superior to 10%. Notice that this does not disaffirm the results obtained for the *No-Starvation MIMO Charging Problem* because a valid solution for the *Minimum-Time MIMO Charging Problem* may be smaller than the time horizon.

Chapter 9

Conclusions and Future Work

In this work, we proposed two novel computational problems related to Wireless Charging using Inductive Power Transfer. The *No-Starvation MIMO Charging Problem* aims at finding a feasible voltage temporal-series for all power transmitting elements in the network which enables all power receivers to stay alive for a determined period of time. The *Minimum-Time MIMO Charging Problem*, in turn, aims at finding a feasible voltage temporal-series which minimizes the time required to recharge all elements in the network.

We proved the decision version of both problems as being NP-Hard and provided a dynamic-programming approach to solve both problems with linear time-complexity towards the duration of the period of time, which is the most expressive input parameter. Furthermore, the method is exponential towards the number of devices in the network. We proposed three algorithms based on the dynamic-programming approach. The *Simple* algorithm populates the *Feasible Future* sets with all reachable states and uses floating-point representation. The *Pareto* algorithm uses only the reachable states generated with voltage vectors that lead to *Pareto-optimal* charge vectors. The *Fly-Weight* algorithm, in turn, represents the states as vectors of bytes and, therefore, requires special treatment to avoid excessive error propagation, although the employed data structure is 5 to 6 times more efficient regarding memory-usage.

We also proposed three greedy algorithms as heuristics for the problems. The *Max-Sum* algorithm chooses at each moment the voltage vector which maximizes the sum of the state-of-charges of all device batteries. The *Max-Min* algorithm maximizes the minimal state-of-charge. Finally, the *Max-Sum-Of-Currents* maximizes the sum of the magnitudes of the receiving currents.

We created an algorithm to generate problem instances with a guaranteed solution and different difficulty levels. We established a methodology for testing and comparing the proposed algorithms using the generated instances. The collected results support the following conclusions.

1. Problems become more difficult as the number of passive devices increases because the charge vector space becomes larger.

2. The *Fly-Weight* algorithm becomes more effective as the number of active devices increases. The exploitation algorithm is favored by the enhancement of the multiplicity of voltage vectors which lead to the same receiving currents.

3. Increasing the number of charge segments makes dynamic-programming algorithms more effective as long as they have a high enough time-to-leave. Indeed, smaller *polytopes* fortify the assumption that all internal states are equivalent to each other.

4. The search space at each time-slot is large enough that the dynamic-programming algorithms cannot fill all feasible *polytopes* using a reasonable time limit.

5. For the *No-Starvation MIMO Charging Problem* and with a sufficient number of charge segments, the *Pareto* algorithm is more effective than all greedy algorithms and the *Fly-Weight* algorithm, achieving 97% effectiveness in “easy instances” and 89% in “hard instances”.

6. For the *No-Starvation MIMO Charging Problem*, the Max-Sum algorithm is the most effective among the proposed greedy algorithms, achieving 92% effectiveness in “easy instances” and 74% in “hard instances”.

7. For the considered limitations of the number of iterations and the *ERL scores* of the instances, all proposed algorithms are equivalent regarding the *Minimum-Time Charging Problem.*

Thus, the results lead us to the following overall conclusions.

- If enough computational power is provided as well as omniscient information about the behavior of the powered devices for a whole time horizon, the Pareto algorithm is indicated to solve the *No-Starvation MIMO Charging Problem.*

- Otherwise, the *Max-Sum* algorithm is indicated to solve both proposed problems.

Future work include the creation of an *dynamic-programming* algorithm that improves the representation of feasible futures, for example, using a *KD-tree* for grouping regions of the charge hyper-space. They also include the development of a more efficient algorithm for the *Max-Sum* heuristic using, for example, a branch-and-bound approach.

Bibliography

Bekmezci, I., Sahingoz, O. K., and Temel, Ş. (2013). Flying ad-hoc networks (fanets): A survey. *Ad Hoc Networks*, 11(3):1254--1270.

Benesty, J., Chen, J., Huang, Y., and Cohen, I. (2009). Pearson correlation coefficient. In *Noise reduction in speech processing*, pages 1--4. Springer.

Bi, Z., Keoleian, G. A., Lin, Z., Moore, M. R., Chen, K., Song, L., and Zhao, Z. (2019). Life cycle assessment and tempo-spatial optimization of deploying dynamic wireless charging technology for electric cars. *Transportation Research Part C: Emerging Technologies*, 100:53--67.

Bulut, E., Hernandez, S., Dhungana, A., and Szymanski, B. K. (2018). Is crowdcharging possible? In *2018 27th International Conference on Computer Communication and Networks (ICCCN)*, pages 1--9. IEEE.

Cao, G., Zhou, H., Zhang, H., Xu, J., Yang, P., and Li, X.-Y. (2018). Requirement-driven magnetic beamforming for mimo wireless power transfer optimization. In *2018 15th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*, pages 1--9. IEEE.

Cederbaum, I. (1956). Analysis of linear n-port networks. *Proceedings of the IEE-Part C: Monographs*, 103(4):267--271.

Chang, S.-Y., Kumar, S. L. S., and Hu, Y.-C. (2017). Cognitive wireless charger: Sensing-based real-time frequency control for near-field wireless charging. In *Distributed Computing Systems (ICDCS), 2017 IEEE 37th International Conference on*, pages 2302--2307. IEEE.

Chen, R., Zheng, C., Zahid, Z. U., Faraci, E., Yu, W., Lai, J.-S., Senesky, M., Anderson, D., and Lisi, G. (2014). Analysis and parameters optimization of a contactless ipt system for ev charger. In *Applied Power Electronics Conference and Exposition (APEC), 2014 Twenty-Ninth Annual IEEE*, pages 1654--1661. IEEE.

Dai, H., Liu, Y., Chen, G., Wu, X., and He, T. (2014). Safe charging for wireless power transfer. In *INFOCOM, 2014 Proceedings IEEE*, pages 1105--1113. IEEE.

Gatilov, S. Y. (2014). Using low-rank approximation of the jacobian matrix in the newton–raphson method to solve certain singular equations. *Journal of Computational and Applied Mathematics*, 272:8--24.

Huang, L., Rieutort-Louis, W., Hu, Y., Sanz-Robinson, J., Wagner, S., Sturm, J. C., and Verma, N. (2012). Integrated all-silicon thin-film power electronics on flexible sheets for ubiquitous wireless charging stations based on solar-energy harvesting. In *2012 Symposium on VLSI Circuits (VLSIC)*, pages 198--199. IEEE.

IEEE (2010). Ieee standard definitions for the measurement of electric power quantities under sinusoidal, nonsinusoidal, balanced, or unbalanced conditions. *IEEE Std 1459-2010 (Revision of IEEE Std 1459-2000)*, pages 1–50. ISSN .

Jadidian, J. and Katabi, D. (2014). Magnetic mimo: How to charge your phone in your pocket. In *Proceedings of the 20th annual international conference on Mobile computing and networking*, pages 495--506. ACM.

Jiang, Q., Qin, Y., Zhao, Y., Xu, C.-Z., and Wang, X. (2017). Maximum power transfer scheme for magnetic resonance charging system. In *Wireless Power Transfer Conference (WPTC), 2017 IEEE*, pages 1--4. IEEE.

Jung, H. and Lee, B. (2019). Optimization of magnetic field focusing and null steering for selective wireless power transfer. *IEEE Transactions on Power Electronics*.

Kim, K., Kim, H.-J., and Choi, J.-W. (2017). Magnetic beamforming with non-coupling coil pattern for high efficiency and long distance wireless power transfer. In *Wireless Power Transfer Conference (WPTC), 2017 IEEE*, pages 1--4. IEEE.

Kisseleff, S., Akyildiz, I. F., and Gerstacker, W. (2015). Beamforming for magnetic induction based wireless power transfer systems with multiple receivers. In *Global Communications Conference (GLOBECOM), 2015 IEEE*, pages 1--7. IEEE.

Lin, C., Guo, C., Dai, H., Wang, L., and Wu, G. (2019). Near optimal charging scheduling for 3-d wireless rechargeable sensor networks with energy constraints. In *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, pages 624--633. IEEE.

Ma, C., Fu, M., and Zhu, X. (2011). Wireless charging of electric vehicles: A review and experiments. In *ASME 2011 International Design Engineering Technical Conferences* *and Computers and Information in Engineering Conference*, pages 881--887. American Society of Mechanical Engineers Digital Collection.

Madhja, A., Nikoletseas, S., Raptopoulos, C., and Tsolovos, D. (2016). Energy aware network formation in peer-to-peer wireless power transfer. In *Proceedings of the 19th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, pages 43--50. ACM.

Madhja, A., Nikoletseas, S., Tsolovos, D., and Voudouris, A. A. (2018). Peer-to-peer energy-aware tree network formation. In *Proceedings of the 16th ACM International Symposium on Mobility Management and Wireless Access*, pages 1--8. ACM.

MarketsandMarkets (2017). Wireless power transmission market. https://www.ecmag.com/section/your-business/wireless-power-transmission-market-worth-1704-billion-2020
Accessed: 2018-11-24.

Mohanti, S., Bozkaya, E., Naderi, M. Y., Canberk, B., and Chowdhury, K. (2018). Wifed: Wifi friendly energy delivery with distributed beamforming. In *IEEE INFOCOM 2018-IEEE Conference on Computer Communications*, pages 926--934. IEEE.

Monti, G., Dionigi, M., Mongiardo, M., and Perfetti, R. (2017). Optimal design of wireless energy transfer to multiple receivers: power maximization. *IEEE Transactions on Microwave Theory and Techniques*, 65(1):260--269.

Nikoletseas, S., Raptis, T. P., and Raptopoulos, C. (2017). Wireless charging for weighted energy balance in populations of mobile peers. *Ad Hoc Networks*, 60:1--10.

Patil, A. (2020). Wireless charging market outlook – 2027. www.alliedmarketresearch.com/wireless-charging-market/ Accessed: 2020-05-16.

Peng, Y., Li, Z., Zhang, W., and Qiao, D. (2010). Prolonging sensor network lifetime through wireless charging. In *Real-Time Systems Symposium (RTSS), 2010 IEEE 31st*, pages 129--139. IEEE.

Pudur, R., Hanumante, V., Shukla, S., and Kumar, K. (2014). Wireless power transmission: A survey. In *International Conference on Recent Advances and Innovations in Engineering (ICRAIE-2014)*, pages 1--6. IEEE.

Shi, L., Kabelac, Z., Katabi, D., and Perreault, D. (2015). Wireless power hotspot that charges all of your devices. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, pages 2--13. ACM.

Tech, I. (2018). Half a billion smartphones and other devices with wireless power technology shipped in 2017, ihs markit says. technology.informa.com/600120/half-a-billion-smartphones-and-other-devices-with-wireless-power-technology-shipped-in-2017-ihs-markit-says/ Accessed: 2020-05-16.

Wang, H., Li, Q., Lin, Z., Cai, C., Wang, W., and Meng, T. (2019). Optimization design of drone wireless charging system based on asymmetric coupling. In *Proceedings of the 2019 International Conference on Artificial Intelligence and Computer Science*, pages 258--264.

WPC (2008). Qi standard. www.wirelesspowerconsortium.com/ Accessed: 2018-11-22.

Xie, L., Shi, Y., Hou, Y. T., and Sherali, H. D. (2012). Making sensor networks immortal: An energy-renewal approach with wireless power transfer. *IEEE/ACM Transactions on networking*, 20(6):1748--1761.

Yang, G. (2017). Magnetic mimo signal processing and optimization for wireless power transfer. *IEEE Transactions on Signal Processing*, 65(11):2860--2874.

Yang, G., Moghadam, M. R. V., and Zhang, R. (2016). Magnetic beamforming for wireless power transfer. In *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, pages 3936--3940. IEEE.

Zhao, C., Zhang, H., Chen, F., Chen, S., Wu, C., and Wang, T. (2020). Spatiotemporal charging scheduling in wireless rechargeable sensor networks. *Computer Communications*, 152:155--170.