Mutaz Tuffaha

On the Management and Control of Isolated Power Systems

Mutaz Tuffaha

Doctoral Thesis

NTNU
Norwegian University of
Science and Technology
Faculty of Information Technology,
Mathematics and Electrical Engineering
Department of Engineering Cybernetics

◼ NTNU
Norwegian University of
Science and Technology

◼ NTNU

Mutaz Tuffaha

On the Management and Control of Isolated Power Systems

Thesis for the degree of Philosophiae Doctor

Trondheim, 2015:261

Norwegian University of Science and Technology
Faculty of Information Technology,
Mathematics and Electrical Engineering
Department of Engineering Cybernetics

NTNU
Norwegian University of
Science and Technology

To the soul of the woman who brought me to life
To the woman who helps me live it and understand it

Summary

Power systems have been a rich arena for research and study since the very beginning of their use commercially in the late 19th century. At the beginning, the objectives of the research were focused on the stability and reliability of the power systems, especially with the fast growth in their size and in the demand on the electric power. Later, due to the need to save losses and fuel consumption, the increasing awareness of the environmental challenges, the growing desire to integrate renewable energy resources, and many other reasons, the attention of the research was drawn to another essential topic, that is the management and planning of power systems. Isolated power systems, however, did not receive similar attention until recently. Isolated power systems are indispensable in many applications such as, but not limited to, island power systems and marine vessels. This work addresses two different topics in the study of power systems that are intended for isolated power systems, but the results provided can also be expanded to regular ones. Thus, the thesis is divided into two parts: the management and scheduling of the power systems, and the control of Gensets.

The first part contains our contributions in the field of operations research and the management of power systems after providing a brief introduction to *integer programming* (IP), *mixed-integer programming* (MIP), and propositional calculus. This introductory chapter is fundamental to understand the basic concepts that were used to develop the models and techniques throughout this work. Then, we describe a new technique to represent *piecewise linear* (PWL) functions in optimization problems based on mixed-logical inequalities. The proposed technique is best suited for special class of discontinuous functions that cannot be handled by the regular SOS method. Finally, we introduce our main contribution in this thesis in the fourth chapter, that discusses the *unit commitment* (UC) and *economic dispatch* (ED) problems. In this chapter, we present a state-space model in discrete time that can be used to solve both the UC and ED, simultaneously. Such models can be useful when *model predictive control* (MPC) philosophy is considered to make the scheduling and planning of power systems more reliable and adaptive to changes in the demand side. Further, we show by simulations that the proposed model could be more accurate than the commonly-used ones. We believe that the proposed model can be the core model for all kinds of power systems, not only the isolated one.

In the second part, we present our results regarding the control of the *generating set* (Genset) that comprises a Diesel engine and a synchronous generator. In chap-

ter 5, we present a control-oriented model of the Genset, and design a controller by feedback linearisation to regulate the shaft speed and the terminal voltage, simultaneously, through two control inputs: the fuel mass and the voltage of the field excitation circuit. We provide simulations to show that the proposed controller make the two manipulated control inputs interact with each other, i.e. they both respond to any change of the terminal voltage or the load. In addition, we discuss the robustness of the proposed controller to unmeasured disturbances, uncertainties, and time delays imposed by the Diesel engine, if they are not so large.

Chapter 6 discusses special class of marine Gensets or shaft generators. In this type of marine Gensets, the Diesel engine is connected through a clutch and a gear box to a synchronous machine, and the main propeller. Such Gensets can be operated in different modes. Thus, we extend the model proposed in Chapter 5 to include the main propeller. Then, we design a controller to regulate the shaft speed and the terminal voltage in one mode of operation. Also, we provide some simulations to show that the proposed controller can be considered robust to small uncertainties and time delays.

The last chapter summarizes the main contributions of this work, and discusses recommendations for possible future work.

Preface

Acknowledgments

Contents

List of Figures

List of Tables

List of Abbreviations

MILNP *Mixed Integer Non-Linear Program.* 11, 22, 23, 45

MILP *Mixed Integer Linear Program.* 3, 11, 14, 16–20, 45, 47, 49, 50, 52–54, 59, 63, 68, 69, 71, 72, 74, 77

MIP *Mixed Integer Program.* 3, 6, 11, 13–15, 17–20, 22, 23, 25, 29, 31, 33, 34, 38, 41, 42, 45, 46, 50, 52, 56, 57, 59, 61, 63, 64, 66, 68, 71, 72, 74–76, 78, 81, 82, 122

MIQP *Mixed Integer Quadratic Program.* 12, 49, 63, 64, 68, 69, 71, 72, 75, 78

MLD *Mixed Logical Dynamic.* 24, 33, 56

MPC *Model Predictive Control.* 6, 7, 54–56, 80–82, 122

NN *Neural Network.* 17

OA *Outer Approximation.* 22, 45

OC *Optimal Control.* 53, 55, 56, 79

OPF *Optimal Power Flow.* 1, 2, 43, 48, 49, 54

OR *Operations Research.* 11, 28, 32

PL *Priority List.* 3, 4, 6, 45, 59

PMS *Power Management System.* 2, 4, 6

PSO *Particle Swam Optimization.* 45

PTI *Power Take In.* 5, 6, 111, 118, 123

PTO *Power Take Out.* 5–7, 111, 112, 115, 118, 123

PWL *Piecewise Linear.* 6, 27–39, 49, 50, 63, 64, 66, 68, 71, 72, 74, 76, 77, 79, 82, 121, 122

SA *Simulated Annealing.* 17, 45

self-UC *Self Unit Commitment.* 42–45

SI *Spark Ignition.* 103

SOS *Special Order Set.* 6, 27, 33, 35, 36, 121

SOS2 *Special Order Set of type 2.* 30, 32, 33, 36–39, 64, 72

SOSD *Special Order Set of type 2 for Discontinuous functions.* 32

UC *Unit Commitment.* 2–4, 6, 11, 28, 41, 42, 44–47, 49, 50, 52–57, 59, 66, 69, 79–82, 121

Chapter 1

Introduction

1.1 Historical Perspective

In 1882 Thomas Edison established the first complete power system in New York city. It was a small dc system that was used mainly to light incandescent lamps [56]. Since then, the power industry has been growing to be one of the most important achievements of humanity all over the history of mankind. With the increasing size of the power systems, and the large drops of the dc voltage when transmitted over long distances, dc systems were replaced with ac systems, because the voltages at the generation level had to be increased, and thus transformers were required. Besides, the ac generators and motors were easier to deal with than the dc ones. Then, Nikola Tesla introduced multi-phase ac systems [56], which by common consent prevailed till our days.

Thus, the early power systems were *isolated*. Later, interconnecting those isolated power systems was advocated to increase the reliability of the power system. Because the interconnection of the power systems necessitated standard frequency and voltages through the network, the study of *power system dynamics* broke through. Then, the *power system management* emerged as a standalone branch of the research topics, and is still ongoing due to the following reasons:

1. the need to reduce transmission and generation losses throughout large systems

2. saving fuel consumption for economic and environmental reasons, especially at that time when storage units were not available

3. privatization of public utilities which lead to more greedy scheduling of power systems by the private companies that aimed to maximize the profits

4. the urging demand to exploit, to the full extent, renewable resources of energy that are intermittent by nature.

Recently, the branch of electrical engineering concerning large power systems, which we refer to as *on-land* power systems from here on, covers many fields, whether on the *generation* level, or the *transmission*, and *distribution* levels. The most important topics include, among many others, *power systems stability* and reliability, *Optimal Power Flow* (OPF), *Automatic Generation Control* (AGC), and *Economic*

Dispatch (ED).

More recently, the concept of *smart grids* erupted in research and industry as the targeted culmination of power engineering. For references on smart grids, the reader is referred to [23]. Basically, a smart grid is a power grid in which the generation is not centralised, that is to say anyone can contribute to the generation process. Then, communication channels between the producers and the consumers are provided. Naturally, this will create a huge market in which real-time pricing, bidding and auctions can be used to accommodate the generation to the demand more tightly. Thus, in smart grids, electrical engineering, communication, software engineering, economics, and logistics are commonly utilised to achieve the targets. In smart grids, isolated power systems can also be integrated into the large grid or *macrogrid*. Smart grids are supposed to be capable of connecting or disconnecting such isolated systems upon request. The isolated power system with these merits is referred to, in smart grid jargon, as a *microgrid*.

1.2 Between On-land and Isolated Power Systems

Isolated power systems are indispensable in many applications such as, among many others, far rural areas, small islands, marine vessels, oil rigs and platforms. Besides, as mentioned above, the targeted microgrids in smart grids are required to operate in two modes: connected to the macrogrid, and *islanding* mode, i.e. disconnected from the grid. Hence, the research on the management and control of the isolated power systems is necessary.

The author in [83] held a comparison between the generation in on-land power systems and marine power systems. Table 1.1 generalizes this comparison.

In spite of the discrepancies given in Table 1.1, the isolated systems can be considered as on-land systems but on much smaller scale. Thus, there is no need for complicated hierarchical control structure to achieve the targets of a reliable power system, especially for smaller systems.

1.2.1 Management of Power systems

In any power system, energy or *Power Management System* (PMS) is an essential unit that monitors and controls the generation and transmission of the power to match the demand on the network. The most important objective of this unit was to decide which units must be turned on and which must be turned off, that problem was later called the *Unit Commitment* (UC). Besides, in case the unit was turned on, the PMS had to solve another problem, that is to decide the output power level of the unit. That problem was later referred to as ED. The problems got more complicated with the increasing size of the power grids with increasing number of generation units, buses, transmission and distribution subsystems. Thus, the study of the OPF was introduced to ensure that each bus in the power grid receives the optimal amount of the *real* and *reactive* power, at the accepted voltage and frequency levels, by controlling many parameters such as the voltage level, transformers tap ratios, and the power generation levels. For comprehensive introduction to this problem, the reader is referred to [22].

		On-land	Isolated
Transmission & Distribution	Network	Complicated	Simple
	Losses	High	Negligible
Generating Units	Number	High	Low
	Size	Large	Small to medium
	Start-up	Slow and costly	Faster and Cheaper
	Variety	Many: fossil fuel, nuclear, hydraulic, solar, wind, ...	A few
Demand & Loads	Nature	Stochastic	Could be deterministic in some applications
	Predictability	Highly accurate	inaccurate and sometimes unavailable
	Controllability	Low but increasing with smart grids	High with possible load limiting
	Importance	Outages must be avoided	Depends on the application

Table 1.1: Comparison between on-land power systems and isolated ones

The engineers of the early systems used to depend on *Priority List* (PL), or *start/stop* tables, prepared a priori [29]. This technique is still in use these days, especially for isolated power systems like marine systems, see e.g. [83] and the references therein. Thanks to the *Dynamic Programming* (DP) technique, the research on the PL developed rapidly and is still ongoing. Due to the progress in the theory and application of statistics, prediction of the demand on the power grid became easier. Correlations between the demand and the weather condition, the time of the day, and the season were apparent. Thus, scheduling and planning of power systems was an eminent possibility in contrast to PL. Later, *Integer Program* (IP) and *Mixed Integer Program* (MIP) was suggested to solve the UC problem whose solution involve solutions of the ED problem.

On the other hand, in isolated power systems PL is still preferred in the industry, and hence the research. One reason for that could be the difficulty of the demand prediction, in general. However, some attempts have been done to manage the isolated systems by techniques based on the UC or ED problems. For example, the authors in [76] proposed a *Mixed Integer Linear Program* (MILP) that can be used for the design and planning of oil platforms. To overcome the difficulty of the demand prediction, the authors in [76] divided the demand into two parts: the constant loads of the crew facilities, and the variable load of the processing facilities that is the oil production. Then, they assumed that the variable load is proportional to the production rate that can be predicted from the given profiles. Another example can be found in [17] where the authors described a scheduling

algorithm of small *autonomous* systems that comprise Diesel engines and wind generators. The focus of the authors in [17] was focused on describing a model to predict the wind speed and load demand. Then, they proposed an algorithm to solve the UC by feeding the short-term predictions and solving the problem for short-term planning horizon [17]. However, in order to reduce the computational time, they exploited the PL for the Diesel engines.

1.2.2 Control of Power Systems

The main target of power systems is to deliver electrical energy that meets the demand, at the minimum cost, at the desired standard level of frequency, voltage, and reliability. That is why a complicated hierarchical control structure is usually used. While PMS takes care of the scheduling and planning of power systems operations, other control devices and units are usually deployed over all levels of the power grid not only to achieve the previous criteria, but also to react to contingencies. The low-level controllers represent the corner stone of that complex hierarchical control structure, specifically, the *speed governor*, and *Automatic Voltage Regulator* (AVR). The generator, that is usually a 3-phase synchronous machine, is driven by a prime mover that could be a steam turbine, hydraulic turbine, wind turbine, Diesel engine, or any other kind. The generator is connected mechanically to the prime mover through the shaft. Now, to keep the frequency stable the rotor speed of the generator, and thus the shaft speed must be kept stable, and this is taken care of by the speed governor. On the other hand, the AVR regulates the terminal voltage by controlling the field excitation circuit in the *rotor* of the synchronous machine.

Speed Governor and Frequency Control

When the electrical load on the generator increases, the rotor speed will drop. Thus, the frequency drops to a new steady-state value depending on the size of the step and the parameters of the generator. It is worth mentioning here that, the frequency control is related to the real power control, because any change in the load will make the speed change. Hence, the studies on *load-frequency control* includes, in addition to the frequency control, the generation control.

In order to compensate for the drop in speed when the load increases, the torque of the prime mover must be increased. The speed governor is a PID controller that uses the integral of the signal of the error between the measured shaft speed and the reference speed or the set point, to control the speed of the shaft by controlling the torque of the prime mover through, e.g the gate of the hydraulic turbine, the valve opening in the case of the steam turbine, or the fuel rack position in the case of the Diesel engine. Speed governors are divided into two main types:

1. The *Isochronous* governor. As the name suggests, this governor ensures that the rotational speed is constant. However, stability studies show that isochronous governors are best suited for isolated units, because if more than one unit is connected to the system, they would tend to "fight each other, each trying to control system frequency to its own settings [56]."

2. The *Droop* governor. This governor allows the reference speed to drop to a new set point, when the load increases. Thus, when several units work in parallel, and the load on the system increases, the step of the load will be distributed over them according to their sizes and the reference speed will change. In this case, all units will rotate with this new speed, and thus the frequency will be stable again, but around a different set point.

AVR and Voltage Control

The synchronous machine consists of two parts; a rotating part called *rotor*, and a stationary part called *stator*. When the machine operates as a generator, the rotor is driven by the prime mover. A magnetic field is produced from the rotor either by a permanent magnet or a winding around the rotor fed by a dc or ac circuit called *field excitation circuit*. When the rotor moves, the magnetic field produced from it will also move. Hence, an emf will be induced in the *armature* winding in the stator with a frequency proportional to the rotational speed, as will be explained in Chapter 5.

The AVR is part of the excitation systems that can take many forms and types. However, the AVR in general is a PID controller that uses the difference between the measured terminal voltage and the set point to control the excitation circuit. Many other functions are also included in the excitation systems, such as limiting functions, load compensation, and power stabilizer.

In large power systems, due to the huge network of the transmission lines, the voltage at each level is related to the reactive power delivered or consumed at each bus. Hence, in order to ensure good voltage stability the reactive power flow must be controlled. This is usually done by various types of reactive power *sources* and *sinks*, such as shunt capacitors, shunt reactors, synchronous condensers, and *static var compensators* (SVC) deployed all over the power grid [56], and [64]. Thus, the voltage stability is achieved by a multi-level control structure, in the bottom of which lies the AVR. Nevertheless, for isolated power systems the problem should be easier.

The problem

The AVR and speed governor act separately, i.e. the governor responds to the change of the shaft speed by changing the amount of the input whether fuel, steam, or water in hydraulic units, whereas the AVR responds to any change of the terminal voltage by changing the field excitation circuit. This may result in undesired oscillations or even instability as explained in [70], especially in isolated units that use isochronous speed governor, that is called a *Generating Set* (Genset). This problem becomes even worse for special class of marine shaft generators in which a Diesel engine is connected through a clutch and a gear box to a synchronous machine connected to the main switch board, and the main propeller. This Genset can be operated in different modes, and hence the manufacturers usually refer to such systems as *Power Take In* (PTI)/*Power Take Out* (PTO) shaft generators.

1.3   Thesis Main Scope

When this project started the objective was defined, in a very broad sense, to propose any method that improves the action of the PMS on marine vessels in regard to units scheduling, especially with these PTI/PTO shaft generators. Besides, the author was directed to find a replacement to the common PL strategy, and preferably try MIP. The resulting work may be considered as a first step in that direction.

1.4   Thesis Outline and Main Contributions

We divide the thesis into two parts. The first part concerns the MIP to solve the UC. Since isolated power systems are small in size, we thought that a state-space model in discrete time of the power generation process, to solve the UC problem, could be useful. The second part focuses on the control of the Genset and the coordination of the AVR and speed governor, especially for marine Genset.
The main contributions in each chapter of this thesis can be summarized as follows:

1. **Part I** consists of three chapters:

   - **Chapter 2** is an introductory chapter on IP and MIP. In order to make it easier for the reader, we present the basic mathematical definitions and results from IP theory. Terms like *polytope*, *convex-hull*, *facet*, and *relaxation* will be explained. In addition, a brief description of the most common deterministic methods to solve IP is introduced, including *Branch-and-Bound* (B&B) and *Cutting Plane* (CP). Finally, some basic results on *propositional calculus* is provided with a description of the so-called *mixed-logical* inequalities. If the reader is familiar with this theory, this chapter can be skipped.

   - **Chapter 3** describes the methods used for optimization over *Piecewise Linear* (PWL) functions. A new technique to represent PWL functions in optimization problems based on mixed-logical inequalities. The proposed technique is best suited for special class of discontinuous functions that cannot be handled by the regular *Special Order Set* (SOS) method. Besides, we propose strong inequalities to reduce the computational complexity.

   - **Chapter 4** discusses the UC and ED problems. The main contribution of this chapter is a state-space model in discrete time that can be used to solve both the UC and ED, simultaneously. The numerical results show that the proposed model could be more accurate than the commonly used ones, especially for small-sized power systems. Then, based on the proposed model, a technique is suggested to capture the start-up cost that is proved to be more accurate than the commonly used ones. Further, strong inequalities are proposed to reduce the computational time, and shown to be effective for both the proposed model and the commonly used ones. Finally, we discuss the *Model Predictive Control* (MPC) strategy utilisation for the purposes of power systems planning and scheduling to adapt with changes in the demand side. Further, we

motivate the preference of the proposed model over the commonly used ones to apply the MPC philosophy.

2. **Part II** consists of two chapters:

   - **Chapter 5** is about the control of the Genset that comprises a Diesel engine and a synchronous generator. Usually, the Genset is controlled by low-level PID controllers, specifically the speed governor for the engine, and the AVR for the generator. In order to coordinate these controllers, we present a non-linear controller of the Genset, that combines the action of the AVR and the governor. First, a model is proposed for the Genset based on the *mean-value* model of the Diesel engine and the simplified model of the synchronous machine. Then, feedback linearisation theory is used to design a controller to regulate the shaft speed and the terminal voltage, simultaneously, through two control inputs: the fuel mass and the voltage of the field excitation circuit. The simulations provided show that the proposed controller make the two manipulated control inputs interact with each other. Thus, if e.g. the load increases on the shaft and the speed drops, both controllers react to retain the speed to its nominal value, and hence the terminal voltage will not be affected. In addition to the time delay imposed by the Diesel engine, the *air/fuel* ratio is modelled as unmeasured uncertain disturbance. So, a primitive sensitivity analysis is provided to discuss the robustness of the proposed controller to the unmeasured disturbances, and time delay. The simulations are provided to show that for small time delays, and small values of the uncertainty, the proposed controller can be considered robust.

   - **Chapter 6** discusses special class of marine Gensets or shaft generators that is connected to the main propeller. Thus, we extend the model proposed in Chapter 5 to include the main propeller. Then, the proposed controller to regulate the shaft speed and the terminal voltage, is modified to cope with this class of shaft generators in one mode of operation, namely PTO. The added mass around the propeller from the hydraulic forces is modelled as another unmeasured uncertain parameter. The simulations presented in this chapter show that the proposed controller can still be considered robust to small uncertainties and time delays.

3. **Chapter 7** wraps up the final conclusions of this work, and discusses recommendations for possible future work.

1.4.1 Publications

The following is a list of the publications that have been written through the course of this thesis:

- M. Tuffaha and J. T. Gravdahl. Control-Oriented Model of a Generating Set comprising a Diesel Engine and a Synchronous Generator. *Modelling, Identification and Control (MIC)*, Vol.36, No.4, pp.199-214. 2015.

- M. Tuffaha and J. T. Gravdahl. Discrete State-Space model to solve the Unit Commitment and Economic Dispatch problems. Submitted to *Energy Systems (ENSY)*.

- M. Tuffaha and J. T. Gravdahl. Dynamic Formulation of the Unit Commitment and Economic Dispatch problems. *IEEE Conference on Industrial Technology (ICIT)*, pages 1294-1298, Seville-Spain, March 17-19, 2015.

- M. Tuffaha and J. T. Gravdahl. Modeling and Control of a Marine Diesel Engine driving a Synchronous machine and a Propeller. *2014 IEEE International conference on Control Applications (CCA) Part of 2014 IEEE Multiconference on Systems and Control*, pages 897-904, Antibes-France, Oct 8-10, 2014.

- M. Tuffaha and J. T. Gravdahl. Mixed-Integer Minimization of the Cost function of the Unit Commitment problem for Isolated power systems. *52nd IEEE Conference on Decision and Control (CDC)*, pages 421-428, Florence-Italy, Dec 11-13, 2013.

- M. Tuffaha and J. T. Gravdahl. Mixed-Integer Formulation of Unit Commitment problem for power systems: Focus on start-up cost. *IEEE 39th Annual Conference of the Industrial Electronics Society (IECON)*, pages 8160-8165, Vienna-Austria, Nov 2013.

Part I

Scheduling of Power Systems

Chapter 2

Preliminaries of Integer Programming and Mixed-Logic

> If I were again beginning my
> studies, I would follow the advice of
> Plato and start with mathematics.
>
> Galileo Galilei

IP or MIP represent core topics in combinatorial optimization theory, *Operations Research* (OR) and discrete mathematics. Many practical problems from those fields can be formulated as IP or MIP. Of those, scheduling problems appear to be the most attractive examples such as: trains, buses or aircraft scheduling problems, jobs assignment problems, travelling salesman problem, and of course power systems scheduling problems or the UC problem. In addition, *logic-based* methods can be very useful for IP and MIP. Hence, a brief summary of the basics of IP and *propositional calculus* could be necessary for this thesis. The reader is assumed familiar with the basics of the optimization theory and *Linear Programming* (LP).

2.1   Preliminaries

In this section, some basic definitions and results on *polyhedra*, that will be used throughout this work, are introduced.

In opposition to LP, IP is the optimization problem in which all of the decision variables are restricted to be integers or binaries. If some of the decision variables are restricted to be integers or binaries, the optimization problem is called MIP. Actually, MIP is usually treated as a special case of IP. Any optimization problem consists of an *objective function* to be minimized or maximized over some decision variables, and *constraints* that restrict the set in which those variables lie; the feasibility region. The objective then is to find the optimizers, that are the decision variables inside the feasibility region which optimize the objective function. This objective function can be linear, in this case the MIP is called MILP. In contrast, the *Mixed Integer Non-Linear Program* (MILNP) is the MIP which has a non-

linear objective function. One special case of the latter class is the *Mixed Integer Quadratic Program* (MIQP) that has a quadratic objective function. In this chapter, minimization problems only are discussed as examples, altough all of them can be extended to the maximization problems, easily.

The following definitions are essential in understanding and attacking optimization problems.

Definition 2.1 (Polyhedron [112]). A set $P \subseteq \mathbb{R}^n$ described by a finite set of linear constraints, as $P = \{\mathbf{x} \in \mathbb{R}^n : A\mathbf{x} \leq \mathbf{b}\}$ where A and b are of appropriate dimensions, is a *polyhedron*. A bounded polyhedron is a *polytope*.

Note that any inequality of the form $\mathbf{ax} \leq b$ defines a half-space. Hence, a polyhedron can alternatively be defined as the intersection of finitely many half-spaces [55]. Note also that feasibility regions of IP are not polyhedra, in general, because such sets are defined with restrictions on the variables to be integers not real. Now, let us introduce some geometrical properties of such sets.

Definition 2.2 (Convexity [55]). A set $X \subseteq \mathbb{R}^n$ is *convex* if the **convex combination** of any two points $\mathbf{x_1},\ \mathbf{x_2}\ \in X$, given by $\lambda\mathbf{x_1} + (1 - \lambda)\mathbf{x_2}$ for any $\lambda \in [0, 1]$, is also in X. The *convex-hull* of a set X, denoted by $conv(X)$, is defined as the set of all convex combinations of points in X.

Simply put, for a set in the 2-dimensional space, convexity means that the line connecting any two points in that set is also in it. It can be proved easily that the intersection of convex sets is also convex. Hence, **a polyhedron is always convex**. The convex-hull of the set X, however, is the smallest convex set containing X. So, if X is convex, then $X = conv(X)$, otherwise $X \subset conv(X)$. Usually, when the problem is complicated, finding the convex-hull of the feasibility region is more complicated than solving the problem itself, because enormous number of inequalities will be required to characterize it [112]. So, the convex-hull of a set cannot be known explicitly in most of the IP problems.

Another important property of polyhedra is *full-dimensionality*. We say that the polyhedron $P \subseteq \mathbb{R}^n$ is full-dimensional if its dimension, denoted by $dim(P)$, is equal to n. The meaning of the dimension is explained by the following definition.

Definition 2.3 (Dimension [112]). The points $\mathbf{x_1}, ..., \mathbf{x_k} \in \mathbb{R}^n$ are *affinely independent*, if the $k - 1$ directions $\mathbf{x_2} - \mathbf{x_1}, ..., \mathbf{x_k} - \mathbf{x_1}$ are linearly independent, i.e. none of them is a linear combination of some or all of the remaining directions. $dim(P)$ is one less than the maximum number of affinely independent points in P.

Note that if $P \subseteq \mathbb{R}^n$ is full-dimensional, i.e. $dim(P) = n$, then there are $n + 1$ affinely independent points in P. Full-dimensionality of a polyhedron is important because it ensures that there is an *interior point* [55]. To illustrate, consider a 2-dimensional space, if a polyhedron is not full-dimensional, all of the points would be co-linear, i.e. lying on one straight line, and thus the interior of the polyhedron would be empty. It can be proved that if a polyhedron is full-dimensional, then there exists a unique description with unique inequalities. However, as mentioned above, finding such characterization would be cumbersome.

The definitions above are used to describe the feasibility region of the problem

that is usually described by a set of constraints. Thus, the constraints or the inequalities that define such sets must be chosen wisely. The following definitions help us understand which inequalities can be more useful in the formulation of the problem.

Definition 2.4 (Face [112])**.** If an inequality $\mathbf{ax} \leq b$ is valid for the polyhedron $P \subseteq \mathbb{R}^n$, the *face* F is defined as the set $\{\mathbf{x} \in P : \mathbf{ax} = b\}$. In such case, the inequality $\mathbf{ax} \leq b$ is said to be *face-defining*.

The validity of the inequality $\mathbf{ax} \leq b$ for P means that **any $\mathbf{x} \in P$ satisfies the inequality**. Obviously, faces are polyhedra by themselves from the definition above. Thus, a face has a dimension that can be found by Definition 2.3, as well. A special class of faces is of great importance in any polyhedron, that is the *facet*.

Definition 2.5 (Facet [112])**.** A face F is called a *facet*, if $dim(F) = dim(P) - 1$. In such case, the inequality $\mathbf{ax} \leq b$ is said to be *facet-defining*.

From Definition 2.3, one can understand that the inequality $\mathbf{ax} \leq b$ defines a facet of the full-dimensional polyhedron $P \subseteq \mathbb{R}^n$, if and only if there exist n affinely independent points in P satisfying the inequality at equality, because $dim(P) = n$. Another special case of faces is called the *vertex*.

Definition 2.6 (Vertex [55])**.** A point $\mathbf{x}^*$, for which the set $\{\mathbf{x} \in P : \mathbf{x} = \mathbf{x}^*\}$ is a face, is called a *vertex*.

Basically, a vertex of a polyhedron is a "corner point" which results from the intersection of some *active* constraints. From LP theory, it can be easily proved that **any vertex of a polyhedron is an extreme point and thus a basic feasible solution**. Actually, this is the basic concept of *simplex method* used to solve LP, in which the vertices are examined until the optimal solution is found. One of the most basic concepts of optimization theory is the following proposition.

Proposition 2.1 ([112])**.** *All extreme points of conv(X) lie in X.*

This result means that the convex-hull of any set can not include any vertices except those contained in the set itself. When formulating IP or MIP, *redundant* inequalities should be avoided. Without going through proper mathematical definitions a redundant inequality is the one that lies outside the polyhedron but still valid because it is *dominated* by a stronger inequality or a facet-defining inequality. Examples of redundant inequalities include a multiplication of a facet-defining inequality. Consider, e.g. an inequality of the form $x < 3$, if another inequality like $x < 1$ is used, then the former is dominated by the latter because it is redundant. Thus, the first step in IP is to find a suitable *formulation* that simplifies the search for the optimizers. The word "formulation" in the jargon of combinatorial optimization has a special definition, and it is not to be confused with the regular meaning of the word.

Definition 2.7 (Formulation [112])**.** A polyhedron $P \subseteq \mathbb{R}^{n+p}$ is a *formulation* for the set $X \subseteq \mathbb{R}^n \times \mathbb{Z}^p$, if and only if $X = P \cap (\mathbb{R}^n \times \mathbb{Z}^p)$.

That is to say, if a feasibility region of an IP or MIP described by the set X, which is not a polyhedron because it contains some constraints on some variables to be integers, its formulation is the same set defined on real variables. Notice that the formulation is a polyhedron which implies that it is convex. However, the convex-hull of the feasibility set, denoted before by $conv(X)$, is not necessarily as same as its formulation.

Formulations are not unique. Actually, many formulations can be found to a specific IP or MIP. Thus, one formulation can be better or *tighter* than another. Mathematically speaking, formulation P_1 is said to be tighter than formulation P_2, if and only if $P_1 \subset P_2$, in other words smaller. In this case, it can be easily proved that the optimal solution obtained by P_1 is *better*.

A formulation can be considered as a special class of *relaxations*.

Definition 2.8 (Relaxation [112]). A problem $\{\min f(\mathbf{x}) : \mathbf{x} \in X'\}$ is a *relaxation* of the problem $\{\min c(\mathbf{x}) : \mathbf{x} \in X\}$ if:

1. $X \subseteq X'$, and
2. $f(\mathbf{x}) \leq c(\mathbf{x})$.

The most important relaxations include: LP relaxation, which is as same as the formulation explained in Definition 2.7 and illustrated in Example 2.1, and the *Lagrangian Relaxation* (LR) which will be discussed later. Relaxations are vital because they represent the first steps used to solve any IP and MIP, as will be explained later. Now, we need to discuss what formulations can be considered *ideal*.

2.1.1 Ideal Formulations

Again, the idealness here has a special meaning.

Definition 2.9 (Ideal Formulation [112], [78]). A formulation $P \subseteq \mathbb{R}^{n+p}$ for the set $X \subseteq \mathbb{R}^n \times \mathbb{Z}^p$ is called *ideal*, if all of its extreme points or vertices coincide with integer variables in $\mathbb{Z}^p$.

To elucidate, let us consider the following MILP example.

Example 2.1 (LP Relaxation of a MILP)**:**
For the MILP described by:

$$\min c_R\mathbf{x} + c_Z\mathbf{y}$$

$$s.t.$$

$$A_R\mathbf{x} + A_Z\mathbf{y} \leq \mathbf{b}$$
$$\mathbf{x} \in \mathbb{R}^n, \ \mathbf{y} \in \mathbb{Z}^p, \tag{2.1}$$

the LP relaxation is given by:

$$\min c_R\mathbf{x} + c_Z\mathbf{y}$$

$$s.t.$$

$$A_R\mathbf{x} + A_Z\mathbf{y} \leq \mathbf{b}$$
$$(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^{n+p}. \tag{2.2}$$

Now, if all of the extreme points or vertices in the polyhedron $\{(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^{n+p} : A_R\mathbf{x} + A_Z\mathbf{y} \leq \mathbf{b}\}$ satisfy the integrality condition that $\mathbf{y} \in \mathbb{Z}^p$, then the formulation in (2.2) is ideal. It is ideal because when simplex algorithm is used to solve the LP, it searches for optimal solutions by examining the vertices. Since all vertices are integers in $\mathbf{y}$, the search will stop once an optimizer is found. On the other hand, if the formulation is not ideal, the optimizer found by simplex may not be integer in $\mathbf{y}$, hence different techniques must be used.

Remark 2.1. Put differently, a formulation P of a feasibility set X is ideal iff $P = conv(X)$. To motivate, by Definition 2.2 and Proposition 2.1 $conv(X)$ is the smallest convex set containing all extreme points of X that satisfy the integrality constraints on the variables.

Checking whether a formulation is ideal or not is another difficult problem by itself. For a IP, consider a problem whose formulation is given by: $P = \{\mathbf{x} \in \mathbb{R}_+^n : A\mathbf{x} \leq \mathbf{b}\}$, then the following result can be proved.

Proposition 2.2 ([112]). *If $\mathbf{b}$ is a vector of integers, then P is ideal iff the matrix A is totally unimodular.*

The unimodularity can be defined as follows.

Definition 2.10 (Totally Unimodular Matrix [112]). Matrix A is said to be *totally unimodular*, if every square sub-matrix of A has determinant +1,-1, or 0.

Actually, from Definition 2.10 one can prove Proposiion 2.2 by using **Cramer's rule**. Although some sufficient conditions were proposed, checking the total unimodularity of a matrix is not an easy task and the checking algorithm may take long time.
Usually, ideal formulations are very hard to be obtained, especially for MIP. Thus, a weaker property is defined and it can be useful in many cases, that is *sharpness*. According to [108] sharp formulations are defined as follows.

Definition 2.11 (Sharp Formulation[108]). A formulation P of a set X is called *sharp*, if the *projection* of P on the *space* of X is $conv(X)$.

The *space* containing the variables in a feasibility set must be in real variables, because projections cannot be made on integer variables. This can be illustrated by the following definition.

Definition 2.12 (Projection [4]). The projection of the polyhedron

$$P = \{(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^n \times \mathbb{R}^p : A\mathbf{x} + B\mathbf{y} \leq \mathbf{b}\}$$

on the space of variables $\mathbf{x} \in \mathbb{R}^n$, denoted by $Proj_x(P)$, is given by:

$$Proj_x(P) = \{\mathbf{x} \in \mathbb{R}^n : \exists\ \mathbf{y} \in \mathbb{R}^p : (\mathbf{x}, \mathbf{y}) \in P\}. \tag{2.3}$$

Incidentally, the process opposite to projection in which more variables are included in the formulation is called *lifting*. For the polyhedron P in the definition

above, one can prove by using **Farkas' Lemma** that its projection on the $\mathbf{x}$ space is given by (see e.g. [5]):

$$Proj_x(P) = \{\mathbf{x} \in \mathbb{R}^n : \mathbf{u_t}(\mathbf{b} - Ax) \geq 0, \ \forall \ t = 1, ..., T\}, \quad (2.4)$$

where $\{\mathbf{u_t}\}_{\forall t}$ are the extreme rays of the cone $\{\mathbf{u} \in \mathbb{R}^m : \mathbf{u}B \geq 0\}$, if $B \in \mathbb{Q}^{m \times p}$. Thus, for Example 2.1, let the feasibility region of the MILP in (2.1) be $X = \{(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^n \times \mathbb{Z}^p : A_R\mathbf{x} + A_Z\mathbf{y} \leq \mathbf{b}\}$. Let its LP relaxation be P. We say that this formulation is sharp if $Proj_X(P) = conv(X)$.

2.2 Enumerations and Complexity

In this section, the hardness of the IP is discussed in order to give an insight of the computational complexity of such problems.

The most straight-forward approach one would think of to solve a IP is to *enumerate*. Differently put, since we are looking for integers in a specific feasibility set that minimize a certain cost function, we can calculate the cost function at all possible integers in the feasibility set. Actually, this may be so time-consuming, especially if the problem contains too many variables and constraints. This gave rise to the so-called *complexity theory* in mathematics which studies the "hardness" of the decision problems. Although this theory is beyond the scope of this thesis, some basic definitions from this field are worth mentioning. Avoiding proper mathematical definitions, decision problems are classified according to the time taken by the algorithm to solve them into two main categories: $\mathcal{P}$ and $\mathcal{NP}$ problems, which in turn can be divided into sub-categories.

Obviously, any computer algorithm used to solve a decision problem must be fed with a list of numbers, that are coded by binary representation, and instructions to be executed. The algorithm is said to be *polynomial-time* if: "it terminates after $\mathcal{O}(n^k)$, and all intermediate computations can be stored in $\mathcal{O}(n^k)$ bits, where n is the input size, k is an arbitrary integer, and $\mathcal{O}(x)$ is any function of the form $ax + b$, with $a, \ b > 0$ [55]." Now, $\mathcal{P}$ is defined as the class of decision problems for which there is a polynomial-time algorithm [55]. Further, $\mathcal{NP}$ is the class of all decision problems for which each *yes-instance* can be checked by a polynomial-time algorithm [55]. In fact, it is still not proved yet whether $\mathcal{NP} = \mathcal{P}$ or $\mathcal{NP} \neq \mathcal{P}$, and thus it is still one of the most important open problems in complexity theory as noted in [55]. However, it can be proved easily that $\mathcal{P} \subseteq \mathcal{NP}$. Then, the most difficult problems in class $\mathcal{NP}$ was defined as $\mathcal{NP} - hard$, or more generally $\mathcal{NP} - complete$. A problem P is said to be in $\mathcal{NP} - hard$ if all problems in $\mathcal{NP}$ can be *polynomially reduced* to P. That is to say, if all problems in $\mathcal{NP}$ can be converted in polynomial-time algorithm to an instance of P, then P is in $\mathcal{NP} - hard$. In fact it can be shown that IP is, in general, $\mathcal{NP} - hard$.

Complexity theory looks so interesting and important for combinatorial optimization because if one can prove that a problem is, e.g. of class $\mathcal{P}$, then the existence of a polynomial-time algorithm can be proved. Pragmatically speaking, this may not be so useful because even if the existence of such algorithm is proved, finding it would be a completely different story. Consider LP problems, for example. LP problems were proved to be of class $\mathcal{P}$, yet no polynomial-time algorithm has been

found as noted in [55]. Methods used to solve LP such as: simplex and *interior-point* methods are not polynomial-time, and the research is still ongoing to find such algorithms.

2.3 Solving Techniques and Solver

In this section, we present the deterministic techniques that are commonly used to solve IP and MIP. Non-deterministic techniques falling under the umbrella of *Artificial Intelligence* (AI), such as *Genetic Algorithm* (GA), *Neural Network* (NN) and *Simulated Annealing* (SA), are beyond our scope.

2.3.1 Relaxation

As mentioned before, relaxations are usually the first steps performed by any solver of IP or MIP. The reason behind that is the fact that relaxations provide *bounds* on the optimal solutions. To elaborate, we present the following result for IP which can be extended for MIP, as well.

Proposition 2.3 ([112]). *Let V_{IP}^* be the optimal solution of the IP given by:*

$$\min \ f_{IP}(\mathbf{x}) \ s.t. \ \mathbf{x} \in X_{IP}$$

Let also V_R^* be the optimal solution of the relaxation of the IP above given by:

$$\min \ f_R(\mathbf{x}) \ s.t. \ \mathbf{x} \in X_R,$$

that satisfies the conditions in Definition 2.8. Then, $V_R^* \leq V_{IP}^*$.

Proof. Let $\mathbf{x}^* \in X_{IP}$ be the minimizer of the IP such that $V_{IP}^* = f_{IP}(\mathbf{x}^*)$. Since $X_{IP} \subseteq X_R$, $\mathbf{x}^* \in X_R$. Hence, and from the second condition in Definition 2.8, $f_R(\mathbf{x}^*) \leq f_{IP}(\mathbf{x}^*) = V_{IP}^*$. Now, from the definition of optimal solution, $V_R^* \leq f_R(\mathbf{x})$, $\forall \ \mathbf{x}$, and the result follows. $\square$

Thus, relaxations are used to find the **lower (upper)** bounds on the minimum (maximum) solutions of the original IP or MIP. Note that the optimal solution of the relaxed problem must be found to determine the bound. While LP relaxation is constructed by relaxing the feasibility region as illustrated by Example 2.1, the LR is constructed by "relaxing" the objective function by including some or all of the constraint in the so-called *Lagrangian* function denoted by $\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda})$, usually. The following example illustrates the LR.

Example 2.2 (LR of the MILP in (2.1))**:**
For the MILP in (2.1), the LR is described by:

$$\min \ \mathcal{L}(\mathbf{x}, \mathbf{y}, \boldsymbol{\lambda}) = c_R \mathbf{x} + c_Z \mathbf{y} - \boldsymbol{\lambda}^T (\mathbf{b} - A_R \mathbf{x} - A_Z \mathbf{y})$$

s.t.

$$\mathbf{x} \in \mathbb{R}^n, \ \mathbf{y} \in \mathbb{Z}^p \ and \ \boldsymbol{\lambda} \geq 0, \tag{2.5}$$

where $\boldsymbol{\lambda}$ is a vector of appropriate dimension of unknown coefficients called Lagrangian multipliers.

Thus, the Lagrangian function is constructed for the minimization problems by subtracting the non-negative constraints, i.e. of the form $\mathbf{b} - A_R\mathbf{x} - A_Z\mathbf{y} \geq 0$. Indeed, the problem in (2.5) is a relaxation of (2.1) according to Definition 2.8. In order to see that, recall that the feasibility set of both problems is $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{y} \in \mathbb{Z}^p$. Besides, the objective function $\mathcal{L}(\mathbf{x}, \mathbf{y}, \boldsymbol{\lambda})$ in (2.5) is less than or equal to the objective function $c_R\mathbf{x} + c_Z\mathbf{y}$ in (2.1) because of subtracting the non-negative constraints. Hence, from Proposition 2.3, the minimum solution obtained from (2.5) is a lower bound of the minimum solution of (2.1). Notice that the decision variables in the LR include the Lagrangian multipliers themselves. For the minimization problems defined over the real numbers, *Karush-Kuhn-Tucker* (KKT) conditions are used to prove that such multipliers exist, and thus optimality can be investigated, see Chapter 12 in [74] and the references therein.

LR can be stronger than LP relaxation due to its importance for *Duality theory*. The duality theory is a very powerful tool in optimization as will be explained in the sequel after presenting the basic idea. Assume that we have a minimization problem defined over some real variables. Assume, also that the Lagrangian function $\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda})$ is defined. Then, the **dual function** is defined as [74]:

$$q(\boldsymbol{\lambda}) = \inf_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}), \tag{2.6}$$

and the **dual problem** of the original minimization problem can be described by [74]:

$$\max_{\boldsymbol{\lambda}} \; q(\boldsymbol{\lambda}) \text{ s.t. } \boldsymbol{\lambda} \geq 0. \tag{2.7}$$

Of course, for maximization problems the dual problem would be a minimization problem. It can be proved that the KKT conditions of the two problems are identical, and hence the dual problem can be used to solve the original problem, especially when it is easier to solve. The following example illustrates how we can formulate the dual problem of the MILP in (2.1).

Example 2.3 (Dual problem of the MILP in (2.1))**:**
By using the LR in (2.5), the dual function can be described by:

$$q(\boldsymbol{\lambda}) = \min_{\mathbf{x}, \mathbf{y}} \; \mathcal{L}(\mathbf{x}, \mathbf{y}, \boldsymbol{\lambda})$$

s.t.
$$\mathbf{x} \in \mathbb{R}^n, \; \mathbf{y} \in \mathbb{Z}^p. \tag{2.8}$$

Then, the dual problem of the MILP in (2.1) is given by:

$$\max_{\boldsymbol{\lambda}} \; q(\boldsymbol{\lambda}) \text{ s.t. } \boldsymbol{\lambda} \geq 0. \tag{2.9}$$

Duality theory and KKT conditions require differentiability of the objective functions and constraints of the optimization problem. Since in IP or MIP differentiability cannot be discussed (unless LP is used first) because some or all of the variables are not defined on real numbers, analogous conditions can be formulated. The following proposition is based on Proposition 10.2 in [112], but it is formulated here for general MILP as in Example 2.3.

Proposition 2.4. *Let $(\mathbf{x}^*, \mathbf{y}^*)$ be a minimizer of the dual function $q(\boldsymbol{\lambda})$ in (2.8). Let also $(\mathbf{x}^*, \mathbf{y}^*)$ satisfy the constraints in (2.1), i.e. $A_R\mathbf{x}^* + A_Z\mathbf{y}^* \leq \mathbf{b}$. Let further $\overline{\boldsymbol{\lambda}}$ be a vector of Lagrangian multipliers such that:*

$$\overline{\boldsymbol{\lambda}}^T (\mathbf{b} - A_R\mathbf{x}^* - A_Z\mathbf{y}^*) = 0. \tag{2.10}$$

Then, $(\mathbf{x}^*, \mathbf{y}^*)$ is a minimizer of the MILP in (2.1).

Proof. Let the cost function $c_R\mathbf{x} + c_Z\mathbf{y}$ be denoted by V_R. Note first that $q(\boldsymbol{\lambda}) \leq V_R$, for any $\mathbf{x}$, $\mathbf{y}$, $\boldsymbol{\lambda}$, and hence $q(\overline{\boldsymbol{\lambda}}) \leq V_R^*$, where V_R^* is the optimal solution we are looking for. From the definition of the dual function we have $V_{Dual}^* \leq V_R^*$, where V_{Dual}^* is the optimal solution of the dual problem in (2.9). Now, from the condition in (2.10), we obtain:

$$q(\overline{\boldsymbol{\lambda}}) = c_R\mathbf{x}^* + c_Z\mathbf{y}^*.$$

Since the point $(\mathbf{x}^*, \mathbf{y}^*)$ is feasible for the MILP, and since V_R^* is supposed to be the minimum solution of the MILP in (2.1), one can write $c_R\mathbf{x}^* + c_Z\mathbf{y}^* \geq V_R^*$. Thus,

$$V_{Dual}^* \geq q(\overline{\boldsymbol{\lambda}}) = c_R\mathbf{x}^* + c_Z\mathbf{y}^* \geq V_R^*,$$

which, with the inequality $V_{Dual}^* \leq V_R^*$, proves the result. $\square$

Remark 2.2. The condition in (2.10) is known as *complementarity* condition. It guarantees that either the constraint is active or the Lagrangian multiplier is zero, or both.

The proposition above says that dual problem can lead to the solution of the original IP or MIP under certain conditions, just like in the regular optimization theory. However, the most important aspect of the dual problem arises from the following fact. From the dual function in (2.8), it can be seen that any feasible solution $\overline{\boldsymbol{\lambda}}$ of the dual problem in (2.9) provides a lower bound for the original minimization problem. In the LP relaxation or LR one must find the optimal solution of the relaxation to determine the lower bound of the original optimization problem as mentioned earlier, whereas by duality a lower bound can be found by finding just a feasible solution for the dual problem and not necessarily the optimal solution. This could be an advantage over the LP relaxation for IP or MIP in many cases. Actually, there is no specific rule to decide which relaxation is better, namely, which one gives the closest lower bound in less time. This depends very much on the problem itself.

2.3.2 Branch-and-Bound

Perhaps because it is so intuitive, B&B represents the corner stone of all deterministic methods used to solve IP or MIP. Basically, B&B is an iterative algorithm that exploits enumeration but in a smarter and more selective way, in opposition to regular enumeration as explained earlier. The basic idea behind B&B algorithm is to divide the feasibility set into subsets based on the possible values of the integer variables, to form some kind of a tree of enumerations. It was mentioned before that a relaxation, whether LP or LR, must be used first because it determines

a lower (upper) bound for the minimization (maximization) problem. Since minimization problems are discussed here, the upper bound, sometimes called *primal* bound, is found by iterations, as will be shown in the sequel, but at the beginning it is assumed ∞. Unless the formulation is ideal (recall Definition 2.9), the decision variables which are supposed to be integers may have fractional parts, and thus an optimal solution is not reached yet. However, the lower bound on the optimal solution is now at hand at least (recall Proposition 2.3). This will be the first explored *node*, and this solution is called an *incumbent*. Then, one would think of *branching*, i.e. **partitioning** the original set into smaller subsets (if possible), that represent new active nodes, and solving the relaxation again over each subset.

Remark 2.3. Keep in mind that a family of sets $\{X_i, \ \forall i\}$ is said to be a partition of set X, if:

1. $\bigcup_i X_i = X$, and
2. $X_i \cap X_j = \emptyset, \ \forall i \neq j$.

Furthermore, it can be shown that with this partitioning, $\min_i\{\min_{x \in X_i} f(x)\} < \min_{x \in X} f(x)$.

For each node, the relaxation is solved again. According to the solutions, we can stop branching or *prune* from the new node in three cases:

1. If the new solution is achieved at integers. In this case, the upper bound and the incumbent are updated.

2. If the new subset is infeasible.

3. If the solution obtained is greater than the upper bound.

Otherwise, the node is branched by partitioning the subset into new subsets. To illustrate the concept of B&B, let us consider the MILP in (2.1). Let the feasibility set be denoted by $X = \{(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^n \times \mathbb{Z}^p : A_R\mathbf{x} + A_Z\mathbf{y} \leq \mathbf{b}\}$. Let the upper bound on the optimal solution be denoted by $\overline{V}$. Algorithm 2.1 illustrates the basic B&B algorithm.

Note that partition in Algorithm 2.1 was made by the floor ($\lfloor \rfloor$), and ceiling ($\lceil \rceil$) functions. Note also that $X_{i_j}^1 \cap X_{i_j}^2 = \emptyset$ and $X_{i_j}^1 \cup X_{i_j}^2 = X_i$, as required. It can be shown that this algorithm always reaches an optimal solution after all nodes are pruned. In which case, the upper and lower bounds coincide. In most cases, this takes too long time. Instead, a percentage tolerance, which is usually called *integrality* or MIP gap, is usually defined by any algorithm such that it will stop searching when the percentage difference between the two bounds is less than this gap.

The algorithm above represents the basic B&B algorithm for linear IP or MIP. Many details can be discussed to improve the algorithm, such as:

(a) The initial upper bound in the algorithm was set to ∞. Some heuristics can be used instead to find a better upper bound initially.

(b) LP relaxation was used in the algorithm above to find the lower bounds. It was explained above that LR may give better bounds in many cases, especially with duality theory.

(c) For LP relaxations which is solved at each iteration, simplex algorithm is usually used. However, alternative methods like interior-point showed better and faster behaviour in many cases.

Algorithm 2.1 Basic B&B Algorithm [112]

Set $\overline{V}$ to ∞

Store the set X in the list of active nodes   ▷ Active nodes not explored yet

procedure NODE SELECTION(List of active nodes)

     Choose an active node, and let it be X_i

end procedure

Solve over the LP relaxation of the set X_i ▷ Let the solution be V_i (which is a lower bound on V), and the minimizers be $(\mathbf{x^i}, \mathbf{y^i})$.

procedure PRUNING(X_i, V_i)

     if X_i is empty **then**

         prune and go to node selection

     end if

     if $V_i < \overline{V} \wedge \mathbf{y^i} \in \mathbb{Z}^!$ (see Remark 2.3) **then**

         store V_i in $\overline{V}$, and $\mathbf{y^i}$ in the incumbent

         prune and go to node selection

     end if

     if $V_i \geq \overline{V}$ **then**

         prune and go to node selection

     end if

end procedure

procedure BRANCHING$(X_i, \mathbf{y^i})$

     if $\mathbf{y^i} \notin \mathbb{Z}^p$, e.g $y_j^i \notin \mathbb{Z}$ **then**

         Partition X_i into $X_{i_j}^1$ and $X_{i_j}^2$

         $X_{i_j}^1 = X_i \cap \{y_j : y_j \leq \lfloor y_j \rfloor\}$ and

         $X_{i_j}^2 = X_i \cap \{y_j : y_j \geq \lceil y_j \rceil\}$ ▷ y_j^i is the jth element of the vector $\mathbf{y^i}$

         Update the node list and go to node selection

     end if

end procedure

if the node list is empty **then**

     terminate

end if

(d) Choosing an active node from the node list was left random in the algorithm above. Different strategies can be used for this choice. For example, one can choose the node that resulted from the current node, i.e. the down node. Another choice is to pick the node whose lower bound is less than the optimal solution.

(e) Branching was done in the algorithm by partitioning the set by using the floor and ceiling functions. Smarter strategies can be used instead, such as *generalized upper bound*.

(f) If the solution obtained for a variable, that is supposed to be integer, is non-integer, branching must be done. In large-size problems, a huge number of variables are involved. Thus, a huge number of nodes may be created, which in turn require a lot of computer memory. So, the nodes to be branched must be

chosen intelligently. One possible approach is to choose the variable with the largest fractional part.

In a nutshell, B&B represents the basic algorithm to solve IP or MIP. However, many techniques and strategies can be used to improve it.

2.3.3 Cutting Planes

As explained earlier, the feasibility set of IP or MIP is not a polyhedron, its convex-hull is, however. Finding the convex-hull of the feasibility set may be more complicated than solving the problem itself, as noted earlier, because it may need tremendous number of constraints. What one would think of is to find more valid inequalities and add them to the formulation, hoping that they may *cut* non-useful parts in the convex-hull of the feasibility set, and hence tighten the formulation. Such inequalities are referred to as CP constraints, or simply cuts. Of course, such cuts can be found depending on the problem and added to the constraints a priori, specifically before applying solving algorithms like B&B. Nevertheless, such cuts do not come at no expense. Adding many constraints to the problem may hinder solving the LP relaxation or the B&B algorithm, and thus, a trade-off must be made.

Alternatively, many CP algorithms have been proposed to generate such cuts automatically by the solvers. The basic idea of any CP algorithm can be described be Algorithm 2.2.

Algorithm 2.2 Basic CP Algorithm [112]

Solve the LP relaxation of the problem
if the solution is not integral **then**
    Find a CP from a family of valid inequalities that cut out the solution obtained
end if
Solve the LP relaxation again after adding the cut
if no cuts can be found **then**
    terminate
end if

The family of the valid inequalities can be found by one or more of the following methods: *Chvátal-Gomory*, or shortly, Gomory inequalities for IP and MIP, *Disjunctive* inequalities, *cover* inequalities for IP, and *flow cover* inequalities for MIP. If the CP algorithm is used throughout the B&B algorithm, explained earlier, the resulting algorithm is called *Branch-and-Cut* (B&C).

2.3.4 Solvers

Recently, most of the solver available such as: **CPLEX, XPRESS, GAMS**, and **Gurobi** depend mainly on B&B and CP. On the other hand, for MILNP, other techniques and algorithms have been proposed in the literature, such as: *Outer Approximation* (OA), and *Benders Decomposition* [9]. However, such techniques are not relevant to this thesis. For further details on the algorithms and software

used to solve MILNP, the reader is referred to the excellent survey in [12], whereas the surveys in [9] and [8] provide a good comprehension of optimization theory.

In summary of this section, the basic B&B and CP algorithms were presented. Further, the following conclusion can be pointed out. When it comes to solving a IP or MIP, one would have two approaches. The first is to try to modify the algorithms used to solve such problems, but this would be a tedious task, especially if the programming is to be done from scratch. The second is to depend on the solvers and try to reformulate the problem by adding more and stronger valid constraints, a priori. In this work, the latter approach was followed.

2.4 Propositional Calculus

Logic can be exploited to solve IP and MIP. In this section, the basics of logic and mixed-logical inequalities are introduced.

Propositional calculus or logic-based methods represent a useful mathematical tool in discrete optimization, nowadays [8]. Since those tools were used throughout this work, it looks a good idea to present a brief introduction on this field. Most of the material presented here is based on the introduction given in [8] and [7].

In *Boolean Algebra*, a statement is usually denoted by a capital, such as: A and B, and it is referred to as *literal*. Each literal can take one of two values only: True (T) or False (F). Literals can be connected by *connectives* that represent logical operation, like: **NOT** ($\neg$), **AND** ($\wedge$), **OR** ($\vee$), **IF** or implication ($\longrightarrow$), and **IFF** or "if and only if" ($\longleftrightarrow$). A *proposition* is a logical expression that comprises any set of literals with connectives. The action of any connective is described by a *truth table* that lists the outcomes of the proposition for all possible combinations of values of the literals. Table 2.1 shows the action of the main logical operations mentioned earlier.

A	B	$\neg A$	$A \wedge B$	$A \vee B$	$A \longrightarrow B$	$A \longleftrightarrow B$
F	F	T	F	F	T	T
F	T	T	F	T	T	F
T	F	F	F	T	F	F
T	T	F	T	T	T	T

Table 2.1: Truth table of basic logical operations.

It can be noted from the Table 2.1, that some operations can be expressed by others. The set of connectives that can replace all other connectives is called a *complete set* [7]. For example, the following relations can be proved from the table [7]:

$$A \longrightarrow B \equiv \neg A \vee B$$
$$A \longleftrightarrow B \equiv (A \longrightarrow B) \wedge (B \longrightarrow A)$$
$$\equiv (\neg A \vee B) \wedge (\neg B \vee A), \tag{2.11}$$

where the sign "$\equiv$" denotes "is equivalent to". Then, the following results can be verified by suitable truth tables:

1. $\neg\neg A = A$.

2. $\neg A \vee A = T$.

3. $\neg A \wedge A = F$.

4. De Morgan's laws:

$$\neg(A \vee B) \equiv (\neg A) \wedge (\neg B)$$
$$\neg(A \wedge B) \equiv (\neg A) \vee (\neg B) \tag{2.12}$$

5. Contraposition:

$$A \longrightarrow B \equiv \neg B \longrightarrow \neg A, \tag{2.13}$$

where the right-hand side proposition is called a *contrapositive*.

Now, if a binary variable $\beta_A \in \mathbb{B} = \{0, 1\}$ is assigned to the literal A, such that $\beta_A = 1$ when $A = T$ and $\beta_A = 0$ when $A = F$, then one can verify that each proposition can be expressed by a linear inequality or more, as shown in Table 2.2. The relations in Table 2.2 represent the core inequalities that can be used in

Proposition	Representation with binary variables
$\neg A$	$1 - \beta_A$
$A \vee B$	$\beta_A + \beta_B \geq 1$
$A \wedge B$	$\beta_A = 1, \ \beta_B = 1$
$A \longrightarrow B$	$1 - \beta_A + \beta_B \geq 1$
$A \longleftrightarrow B$	$\beta_A = \beta_B$

Table 2.2: Linear expressions or inequalities that represent propositions with basic logical operations.

disjunctive optimization mentioned earlier [8]. In addition, the authors in [7] used the relations in Table 2.2 to propose what they called *Mixed Logical Dynamic* (MLD) model that can be used to model control systems that involve logic rules. In such control systems, the above relations are not enough to model an output, a state, or in general a function that changes its behaviour according to logic rules. A set of inequalities, that involve continuous and binary variables, was proposed. Such inequalities are called *mixed-integer inequalities*.

Consider the statement $[f(x) \leq 0]$, where $f(x) : \mathbf{R} \mapsto \mathbf{R}$ is a linear function. Let the maximum and the minimum of $f(x)$ be M and m, respectively. Let also β be an arbitrary binary variable. Then, the following relations can be verified [7]:

1. $[f(x) \leq 0] \wedge [\beta = 1]$ is true iff, $f(x) - \beta \leq -1 + m(1 - \beta)$.

2. $[f(x) \leq 0] \vee [\beta = 1]$ is true iff, $f(x) \leq M\beta$.

3. $[f(x) \leq 0] \longrightarrow [\beta = 1]$ is true iff,

$$f(x) \geq \epsilon + (m - \epsilon)\beta, \tag{2.14}$$

where $\epsilon > 0$ is arbitrarily small.

4. $[f(x) \leq 0] \longleftrightarrow [\beta = 1]$ is true iff,

$$f(x) \leq M(1 - \beta)$$
$$f(x) \geq \epsilon + (m - \epsilon)\beta. \tag{2.15}$$

5. The product of two binary variables $\beta_1\beta_2$ can be replaced by an *auxiliary* binary variable β_3, with the following inequalities:

$$-\beta_1 + \beta_3 \leq 0$$
$$-\beta_2 + \beta_3 \leq 0$$
$$\beta_1 + \beta_2 - \beta_3 \leq 1. \tag{2.16}$$

6. Finally, the product $\beta f(x)$ can be replaced by an auxiliary variable z with the following constraints:

$$m\beta \leq z \leq M\beta$$
$$f(x) - M(1 - \beta)\beta \leq z \leq f(x) - m(1 - \beta) \tag{2.17}$$

The above mixed-integer inequalities can be considered as some kind of *bigM* inequalities [8]. The bigM inequalities are any form of inequalities that exploit any valid upper or lower bound. Generally speaking, bigM inequalities are not preferred in IP or MIP because they may loosen their formulation resulting from the LP relaxation [95].

2.5 Conclusion

To sum up, some basics on IP and MIP were presented. The basic concepts of polyhedra, convex-hull, facets, formulations, and relaxations were defined. A brief description of the algorithms used by solvers of IP and MIP, like B&B and CP, was introduced. Then, a brief summary of the logic-based methods and mixed-integer inequalities was presented. The material provided in this chapter will be used throughout this work.

Chapter 3

Optimization over Piecewise Linear Functions

> We have not succeeded in answering all our problems. The answers we have found only serve to raise a whole set of new questions. In some ways we feel we are as confused as ever, but we believe we are confused on a higher level and about more important things.
>
> From *Stochastic Differential Equations*
> by B. Øksendal

PWL approximation represents a powerful tool in optimization theory for non-linear functions. *Special Order Set* (SOS) seems to be the most common technique used to model such functions. Spurred by the thought that SOS may not be very accurate due to its dependence on convex combinations, we proposed a method to optimize over PWL functions in [102] and [107]. The proposed method is based on resolving non-linearities by using mixed-logical constraints. The method is best suited for special class of discontinuous PWL functions. Further, some strong inequalities were proposed to tighten the formulation and reduce the computational complexity. In this chapter, some literature review and main contributions on this topic is provided. Then, the proposed technique is introduced in the third section. Numerical experiments with discussion and evaluation are presented in the fourth section. Finally, some conclusions are drawn out in the last section.

3.1 Introduction

One of the most challenging problems in OR, especially those involving scheduling such as UC, is the following:

$$
\min_{\mathbf{x}, \boldsymbol{\alpha}} \quad J = F(\mathbf{x})
$$

$$
\text{s.t.}
$$

$$
\underline{x}_n \alpha_n \leq x_n \leq \overline{x}_n \alpha_n, \quad \forall n \in \{1, ..., N\}
$$

$$
A\mathbf{x} + B\boldsymbol{\alpha} \leq b, \tag{3.1}
$$

where $F(\mathbf{x}) : \mathbb{R}^N \longrightarrow \mathbb{R}$ is a linear or non-linear function, $x_n \in [\underline{x}_n, \overline{x}_n]$ and $\mathbf{x} = [x_1, ..., x_N]^T$, $\alpha_n \in \{0, 1\}$ is a binary indicator and $\boldsymbol{\alpha} = [\alpha_1, ..., \alpha_N]^T$. Further, A, B and b are matrices of appropriate dimensions. The binary indicator α_n is used to ensure that x_n lies in the specified domain, that is $x_n \in [\underline{x}_n, \overline{x}_n] \cup 0$. The problem in (3.1) belongs to the broad class of MIP. Of our interest is the the case when the function $F(\mathbf{x})$ is *separable*, i.e. it can be written as:

$$
F(\mathbf{x}) = \sum_{n=1}^{N} f_n(x_n), \tag{3.2}
$$

where $f_n(x_n) : \mathbb{R} \longrightarrow \mathbb{R}$ is continuous. Let further, the cost function comprise two parts: *variable-dependent* cost $f_n^v(x)$ and fixed or *no-load* cost c_n. Thus, $f_n(x_n) = f_n^v(x_n) + c_n$ such that $f_n^v(0) = 0$. Then, the problem in (3.1) can be rewritten as:

$$
\min_{\mathbf{x}, \boldsymbol{\alpha}} \quad J = \sum_{n=1}^{N} f_n^v(x_n) + c_n \alpha_n
$$

$$
\text{s.t.}
$$

$$
\underline{x}_n \alpha_n \leq x_n \leq \overline{x}_n \alpha_n, \forall n \in \{1, ..., N\}
$$

$$
A\mathbf{x} + B\boldsymbol{\alpha} \leq b. \tag{3.3}
$$

Note that the fixed cost c_n is multiplied by the binary indicator α_n in the objective function to guarantee that it is not taken into account unless the variable $x_n > 0$. Generally speaking, all functions can be approximated to good accuracy by a PWL function [50]. The accuracy of this approximation can be controlled by the number and location of the line segments used. For more information on this, the readers are referred to [27], and [54]. The reduction of computational time due to the exploitation of the PWL functions motivates replacing the cost function in (3.3) by:

$$
\min_{\mathbf{x}, \boldsymbol{\alpha}} \quad J = \sum_{n=1}^{N} \tilde{f}_n^v(x_n) + c_n \alpha_n
$$

$$
\text{s.t.}
$$

$$
\underline{x}_n \alpha_n \leq x_n \leq \overline{x}_n \alpha_n, \quad \forall n \in \{1, ..., N\}
$$

$$
A\mathbf{x} + B\boldsymbol{\alpha} \leq b, \tag{3.4}
$$

where $\tilde{f}_n^v(x_n)$ is the PWL approximation of $f_n^v(x_n)$ given by:

$$
\tilde{f}_n^v(x_n) = \left\{ \begin{array}{ll} m_1^n x_n, & \underline{x}_n \leq x_n \leq a_1^n \\ m_2^n x_n + d_2^n, & a_1^n \leq x_n \leq a_2^n \\ \vdots & \\ m_L^n x_n + d_L^n, & a_{L-1}^n \leq x_n \leq \overline{x}_n \end{array} \right\}, \tag{3.5}
$$

where the domain $[\underline{x}_n, \overline{x}_n]$ is partitioned into L intervals with $(L+1)$ break points $\underline{x}_n = a_0^n < a_1^n < ... < a_L^n = \overline{x}_n$, and $m_1^n, ..., m_L^n$, and $d_2^n, ..., d_L^n$ are the slopes and cost-intercepts of the line segments, respectively. Note that d_1^n was assigned to zero to satisfy the assumption that $\tilde{f}_n^v(0) = 0$. Without loss of generality, one can assume that the number of segments L is the same $\forall \ \tilde{f}_n^v(x_n), \ \forall n \in \{1, ..., N\}$, for, as hinted by the authors in [21], if they are not the same one can choose extra break points on segments of the same slope.

To this end, we will focus in this chapter on the optimization over general PWL functions denoted here by $\tilde{f}_n(x_n)$ and the techniques used to attack the optimization problems over such functions. Namely, we neglect the binary indicators α_n and the linear constraint $A\mathbf{x} + B\boldsymbol{\alpha} \leq b$ from the formulation in (3.4). Further, we don not necessitate that $\tilde{f}_n(0) = 0$.

3.2 Available Techniques

The optimization over PWL separable functions has been studied extensively in literature. The authors in [18] and [95], e.g. provided good surveys on the MIP methods used to model the PWL functions in optimization problem. While proposing a technique for optimization over non-separable PWL functions, the authors in [108] wrote an excellent literature review, as well. The models of PWL in optimization problems can be classified under three main methods: *incremental*, *convex combination*, and *multiple choice* explained briefly in the subsequent subsections. For convenience, we drop the index n from here on.

3.2.1 Incremental Method

According to the authors in [78] and [18], this method is attributed to the pioneering work in [68] and [19], and it is usually referred to in literature as the *Delta* method. In this method, each variable x is expressed as the sum[78]:

$$
x = a_0 + \sum_{l=1}^{L} y_l, \tag{3.6}
$$

where $y_l, \ \forall l \in \{1, .., L\}$ are continuous variables that satisfy the following two conditions [78]:

1. $0 \leq y_l \leq a_l - a_{l-1}$,

2. if $y_l < a_l - a_{l-1}$, then $y_{l+1} = 0$.

The second condition above is called a *dichotomy*, which indicates that it is a bidirectional decision, either $y_l = a_l - a_{l-1}$ or $y_{l+1} = 0$. In other words, $y_{l+1}(y_l -$

$a_l + a_{l-1}) = 0$ (not suitable for LP). The proposed technique to enforce the second condition is to use some binary variables $\beta_l \in \{0, 1\}$, $\forall l \in \{1, ..., L-1\}$, such that [78]:

$$
\begin{aligned}
y_1 &\leq a_1 - a_0 \\
y_L &\geq 0 \\
y_l &\geq (a_l - a_{l-1})\beta_l \forall l \in \{1, ..L-1\} \\
y_l &\leq (a_l - a_{l-1})\beta_{l-1} \forall l \in \{2, ..L\}.
\end{aligned}
\tag{3.7}
$$

Thus, if $\beta_{l-1} = 0$ then the only feasible solution would be when $\beta_l = 0$. Accordingly, if the PWL objective function is continuous, it can be expressed with the continuous variables y_l's as (see e.g. [78]):

$$
\tilde{f}(x) = f(a_0) + \sum_{l=1}^{L} m_l y_l.
\tag{3.8}
$$

3.2.2 Convex Combination Method

This method is based on the fact that if the variable x lies in the interval $[a_l, a_{l+1}]$, then x can be written as a convex combination of the two consecutive points a_l and a_{l+1}, since the domain of x is convex. Thus (see e.g. [95]):

$$
x = a_0 w_0 + a_1 w_1 + ... + a_L w_L,
\tag{3.9}
$$

where w_l, $\forall l \in \{0, .., L\}$ are continuous variables or weights that satisfy the following two conditions [95]:

1. $\sum_{l=0}^{L} w_l = 1$,

2. The set $\{w_l \in \mathbb{R}^+ : \forall l \in \{0, ..., L\}\}$ is *Special Order Set of type 2* (SOS2).

SOS2 sets are attributed to the work in [6], and they are defined as follows:

Definition 3.1. [6] A set is called SOS2 if at most two of its elements are non-zero, and in case two elements are non-zero they must be adjacent.

Several algorithms were proposed in the literature to satisfy the SOS2 condition. At the beginning, the authors in [6] proposed a modification in the B&B algorithm to capture this condition. The common practice is to introduce some binary variables to enforce the second condition above. For example, one can use the formulation given in [50] and [59] in which, the binary variables β_l, $\forall l \in \{1, ..., L\}$ are used with the following constraints:

$$
\begin{aligned}
\sum_{l=1}^{L} \beta_l &= 1 \\
w_0 &\leq \beta_1, \qquad w_L \leq \beta_L \\
w_l &\leq \beta_l + \beta_{l+1}, \quad \forall l \in \{1, ..., L-1\}.
\end{aligned}
\tag{3.10}
$$

Then, if the PWL objective functions is continuous, it can be expressed with the weights w_l's as (see e.g. [50]):

$$\tilde{f}(x) = \sum_{l=0}^{L} f(a_l) w_l. \tag{3.11}$$

The above method is sometimes called the *Lambda* method.

3.2.3 Multiple Choice Method

This technique is described in [18] and [108] among others. In this method, the variable x is expressed as:

$$x = \sum_{l=1}^{L} z_l, \tag{3.12}$$

where z_l, $\forall l \in \{1, .., L\}$ are continuous variables that carry the value of x if it lies in the lth segment. Hence, they must satisfy the following two conditions:

1. z_l must lie in the lth segment,

2. if $z_l = x$, all other $z_i = 0$, $\forall i \in \{1, ..., L\}, i \neq l$.

To satisfy the second condition binary variables β_l, $\forall l \in \{1, ..., L\}$ are again required, such that:

$$\beta_l a_{l-1} \leq z_l \leq \beta_l a_l, \quad \forall l \in \{1, ..., L\}$$

$$\sum_{l=1}^{L} \beta_l = 1. \tag{3.13}$$

Then, the PWL objective functions can be expressed as(see e.g. [18]):

$$\tilde{f}(x) = \sum_{l=0}^{L} m_l z_l + d_l \beta_l. \tag{3.14}$$

Remark 3.1. All of the methods above require the continuity of the PWL function $\tilde{f}(x)$.

3.2.4 Challenges and Improvements

The author in [78] compared between the incremental and convex combination formulations. He proved that the incremental formulation is *locally* ideal, whereas the convex combination formulation is not [78]. While, the concept of ideal formulation is explained in Definition 2.9, the term local is used to refer to the portion of the MIP that concerns the PWL approximations and not the whole problem. Hence, for the purposes of this chapter, the term locally will be redundant. Further, he proposed a modification to the convex combination method that makes it ideal. Nevertheless, the authors in [108] proved, some years later, that the convex combination formulation in (3.9) and (3.10) is sharp by describing the *epigraph* of the PWL function.

Inspired by the work in [78], the author in [92] modified the convex combination formulation not only to be ideal but also to make it suitable for discontinuous PWL functions, i.e, functions of the form:

$$\tilde{f}(x) = \left\{ \begin{array}{ll} m_1 x + d_1, & a_0 < x < a_1 \\ \vdots & \\ m_L x + d_L, & a_{L-1} < x < a_L \end{array} \right\}. \tag{3.15}$$

The discontinuity was handled in [92] by assigning two weights around each discontinuity from the left and right, and then the value of the function can be approximated by the convex combination of those weights. This lead to the development of the new method *disaggregated convex combination*. Later, the authors in [21] proposed a new type of ordered sets instead of SOS2 and they called it *Special Order Set of type 2 for Discontinuous functions* (SOSD). The formulation proposed in [21] can be used for any discontinuous PWL function and it is proved to be ideal, but the authors necessitated the weaker property of *lower semi-continuity*.

Definition 3.2. [21] A PWL univariate function $h(x) : [0, u] \longrightarrow \mathbb{R}$ where $u > 0$ is called lower semi-continuous if $h(x) \leq \lim_{x' \rightarrow x} \inf h(x')$, where $\{x'\}$ is a sequence in the domain of x

This property is weaker than continuity but it guarantees that the function has a minimum.

Remark 3.2. Lower semi-continuity is not equivalent to continuity from the left, as a function can be lower semi-continuous but not continuous from the left. Consider, e.g. the function:

$$f(x) = \left\{ \begin{array}{ll} 0, & x < 0 \\ -1, & x = 0 \\ 1, & x > 0 \end{array} \right\}.$$

Obviously, the function is not continuous from the left since $\lim_{x \rightarrow 0^-} f(x) = 0$, while $f(0) = -1$. However, it is lower semi-continuous since $f(0) = -1$ is less than $\lim_{x \rightarrow 0} \inf f(x) = 0$.

Remark 3.3. A lower semi-continuous function always has a minimum which can be seen directly from Definition 3.2. However, a function that is continuous from the left may not have a minimum. Moreover, for a PWL function, if it is continuous from the left it must have an infimum that is greater than $-\infty$.

Other authors, however, try to replace the need for binary integers by modifying the B&B technique. For example, the authors in [51] proposed a set of constraints (cuts) in a B&C technique to achieve that. Then, the authors in [109] proposed a formulation which describes the SOS2 condition that has a number of binary variables and extra constraints logarithmic in the number of continuous variables, and thus less number of binary variables and constraints. Finally, the authors in [95] modified the aforementioned methods to to be more efficient when binary indicators are added to the optimization problem over PWL functions as in (3.3). Briefly put, from the common consent in the OR society, it seems that SOS2 is the cutting-edge method to deal with PWL functions in optimization problems, and

the competition among researchers to improve PWL optimization methods, based on SOS2, is fierce.

3.3 Mixed-Logical Model

The problem with SOS is that it depends on the convex combination method which may make it less accurate than other methods. Incited by the previous idea, we proposed a MIP to optimize over PWL functions in [102], which we call here *mixed-logical model*. The proposed model in [102] looks intuitive but it may not be very efficient compared with the available techniques, as will be shown in the sequel. However, we show that this model can be used for discontinuous PWL functions that are not lower semi-continuous. Further, we propose here some strong inequalities to improve the model, and further we prove that they are facet-defining.

3.3.1 Mathematical Model

The proposed algorithm is inspired by the MLD model presented in [7]. In this model, the space of the feasibility set of variable x is lifted by introducing new variables, as will be shown in the sequel. Binary indicators for showing in which segment the variable x lies are introduced, similar to the multiple choice method. Then, the non-linearities are resolved by bigM linear inequalities, like those introduce in Section 2.4. So, we still consider optimization problems over PWL functions of the form shown in (3.5). As before, the index n is dropped to simplify the notation, and $x \in [\underline{x}, \overline{x}]$ is partitioned into L intervals with $(L+1)$ break points $\underline{x} = a_0 < a_1 < ... < a_L = \overline{x}$. Let us assume without loss of generality that $\underline{x} = a_0 = 0$. In case it is not, one can make the transformation $x' = x - a_0$ and minimize over the new variable x'. Now, let $\beta_1, ..., \beta_{L-1}$ be binary variables assigned for the variable x such that:

$$[x \leq a_1] \longleftrightarrow [\beta_1 = 1]$$
$$[x \leq a_2] \longleftrightarrow [\beta_2 = 1]$$
$$\vdots$$
$$[x \leq a_{L-1}] \longleftrightarrow [\beta_{L-1} = 1], \tag{3.16}$$

with the following conditions:

$$[\beta_1 = 1] \longrightarrow [\beta_2 = ... = \beta_{L-1} = 1]$$
$$[\beta_2 = 1] \longrightarrow [\beta_3 = ... = \beta_{L-1} = 1]$$
$$\vdots$$
$$[\beta_{L-2} = 1] \longrightarrow [\beta_{L-1} = 1]. \tag{3.17}$$

Based on (2.15), since the variable x is bounded because $x \in [0, a_L]$, each if-statement in (3.16) can be represented by a pair of linear bigM inequalities as:

$$x - a_l \leq (a_L - a_l)(1 - \beta_l)$$
$$x - a_l \geq \epsilon - (a_l + \epsilon)\beta_l, \quad \forall l \in \{1, ..., L-1\}, \tag{3.18}$$

where ϵ is an arbitrary small positive number. Based on the expression in Table 2.2, the condition in (3.17) can be satisfied with the following inequality:

$$0 \leq \beta_1 \leq \beta_2 \leq .... \leq \beta_{L-1} \leq 1. \tag{3.19}$$

Then, $\tilde{f}(x)$ can be described, by using the binary indicators $\beta_1, ..., \beta_{L_1}$, as:

$$
\begin{aligned}
\tilde{f}(x) &= \beta_1 m_1 x + (\beta_2 - \beta_1)m_2 x + ... + (\beta_{L-1} - \beta_{L-2})m_{L-1}x + m_L x \\
&\quad + d_1\beta_1 + (\beta_2 - \beta_1)d_2 + ... + (\beta_{L-1} - \beta_{L-2})d_{L-1} + d_L,
\end{aligned} \tag{3.20}
$$

which can be written in compact form as:

$$\tilde{f}(x) = \sum_{l=1}^{L-1} \left[(m_l - m_{l+1})x\beta_l + (d_l - d_{l+1})\beta_l \right] + m_L x + d_L. \tag{3.21}$$

The terms $x\beta_l$ in (3.21) are non-linear. Hence, one needs to linearise them. Based on (2.17), each of those terms is replaced with an auxiliary variable z_l, $\forall l \in 1, ..., L-1$, with the following bigM constraints:

$$0 \leq z_l \leq a_l\beta_l \tag{3.22a}$$
$$x - a_L(1 - \beta_l) \leq z_l \leq x, \tag{3.22b}$$
$$\forall l \in \{1, ..., L-1\}.$$

Then, the proposed MIP for optimization over PWL function $\tilde{f}(x)$ can be summarized as:

$$\tilde{f}(x) = m_L x + d_L + \sum_{l=1}^{L-1} \left[(m_l - m_{l+1})z_l + (d_l - d_{l+1})\beta_l \right], \tag{3.23}$$

subject to the constraints in (3.18), (3.19) and (3.22).
In order to evaluate the model above, we need to study the feasibility region described by the mentioned constraints. Let the feasibility set be:

$$P^A = \left\{ (x, \mathbf{z}, \boldsymbol{\beta}) \in \mathbb{R}^L \times \mathbb{B}^{L-1} : (3.18), (3.19), \text{and } (3.22) \text{ are satisfied} \right\}, \tag{3.24}$$

where $\mathbf{z} = [z_1, ..., z_{L-1}]$ and $\boldsymbol{\beta} = [\beta_1, ..., \beta_{L-1}]$. Let further, the LP relaxation of the set P^A be:

$$P_{LP}^A = \left\{ (x, \mathbf{z}, \boldsymbol{\beta}) \in \mathbb{R}^L \times [0, 1]^{L-1} : (3.18), (3.19), \text{and } (3.22) \text{ are satisfied} \right\}. \tag{3.25}$$

Note that P_{LP}^A is a polyhedron, and thus convex. Hence, we can prove the following results.

Proposition 3.1. *The formulation P_{LP}^A is not ideal.*

Proof. In order to prove that formulation P_{LP}^A is not ideal, it is enough to give an example in which an extreme point is not an integer in β_l. Consider the PWL function given by:

$$\tilde{f}(x) = \left\{ \begin{array}{ll} 1 - x & 0 \leq x \leq 1 \\ 2x - 2 & 1 \leq x \leq 2 \end{array} \right\}. \tag{3.26}$$

The minimization problem over $\tilde{f}(x)$ by using the proposed model can be written as:

$$\min_{x,\beta_1,z_1} 2x - 2 + 3\beta_1 - 3z_1$$

$$\text{s.t.}$$

$$x - 1 \leq (1 - \beta_1)$$
$$x - 1 \geq \epsilon - (1 + \epsilon)\beta_1$$
$$0 \leq z_1 \leq \beta_1$$
$$x - 2(1 - \beta_1) \leq z_1 \leq x. \tag{3.27}$$

It is not hard to see that the point $(x, \beta_1, z_1) = (1 + \epsilon, \frac{1-\epsilon}{2}, 0)$ is an extreme point for which $\beta_1 \notin \mathbb{B}$. □

Remark 3.4. Bearing in mind Remark 2.1, from the proof given for Proposition 3.1 it should be clear that $P_{LP}^A \neq conv(P^A)$.

Nevertheless, the subsequent result can be useful.

Proposition 3.2. *The formulation P_{LP}^A is sharp.*

Proof. In Appendix A. □

So, the projection of the formulation is equal to the convex-hull of the original space, that is the space of x in our case. Now, let us study the feasibility set itself.

Proposition 3.3. *$conv(P^A)$ is full-dimensional.*

Proof. In Appendix A. □

Note that the formulations with SOS are not necessarily full-dimensional as noted in [51]. Thus, we have a full-dimensional convex-hull of the feasibility region. As stated in Chapter 2 the convex-hull of a set could be very hard to obtain and it may take many inequalities to describe it, but as long as it is full-dimensional we can search for more facets to tighten the formulation. That is the topic of the next sub-sections. However, we first need to show why the proposed model can be more efficient than common methods for special class of PWL functions.

3.3.2 Discontinuity

So far, the PWL function $\tilde{f}(x)$ has been assumed continuous. As mentioned earlier, SOS can be used for discontinuous functions that are lower semi-continuous, but what if the function is not lower semi-continuous? Let us consider a discontinuous PWL function that is continuous from the left, i.e. of the form:

$$\tilde{f}(x) = \begin{cases} m_1 x + d_1, & 0 \leq x \leq a_1 \\ m_2 x + d_2, & a_1 < x \leq a_2 \\ \vdots \\ m_L x + d_L, & a_{L-1} < x \leq a_L. \end{cases} \tag{3.28}$$

Notice that, as mentioned in remark 3.3, such functions are not guaranteed to have a minimum. Nevertheless, with the proposed formulation we can find values very close to the infimum. To illustrate, because of the linearity of the PWL function, the infimum is guaranteed to be $> -\infty$. Note that in models with SOS, x must lie in the sub-interval $[a_{l-1}, a_l]$, so $\tilde{f}$ must be bounded on any of those sub-intervals. This explains why this model can not approximate the infimum in this case. In contrast, x in the set P^A can take any value in $[0, a_L]$ and the binary variables are used to indicate whether $x \leq a_l$ or not, regardless of the value of $\tilde{f}$. Assume that we have a discontinuity at $x = a_l$ at which the function is not lower semi-continuous, but still continuous from the left and has an infimum. Thus, $\tilde{f}$ is bounded by the value $\tilde{f}(a_l + \epsilon)$, where ϵ is the amount by which x exceeds a_l. Thus the model is still valid. Some would argue that this can be done using SOS2. In this case, the sub-interval in which x lies is $[a_l + \epsilon, a_{l+1}]$, which will make the domain of x non-convex. As far as we know, this is not appropriate for models with SOS2, unless the PWL function is modified, e.g. one can assume that the function is constant in the sub-interval $[a_l, a_l + \epsilon]$.

3.3.3 Strong Inequalities

As was explained in Chapter 2, if a polyhedron is full-dimensional a unique description for this polyhedron must exist with a finite number of facet-defining inequalities (see Definition 2.5). However, finding this minimal unique description is a tedious task by itself, and in some cases it is more complicated than solving the problem itself. For numerical methods to find facets, the reader is referred to [46]. Throughout our research some strong inequalities for the model P^A, that contribute to reducing the computational time, were found.

Proposition 3.4. *The inequalities*

$$z_l \leq x - a_l(1 - \beta_l) \qquad \forall l \in \{1, ..., L-1\} \qquad (3.29)$$

are valid for conv(P^A).

Proof. Indeed, if $x \leq a_l$, then $\beta_l = 1$ from (3.18). So, $z_l \leq x$ which is valid. On the other hand, if $x > a_l$, then $\beta_l = 0$ from (3.18). Accordingly, $z_l = 0$ from (3.22), and the inequality in (3.29) will yield $z_l \leq x - a_l$, which is also valid because $x - a_l > 0$. $\square$

The inequalities in (3.29) are not facet-defining. However, they cut out some part of the formulation P^A_{LP}. In order to see that, let us consider the projection of P^A_{LP} on the plane $x = \frac{a_1}{2}$. Let us take one inequality of this set given by:

$$z_1 \leq x - a_1(1 - \beta_1). \qquad (3.30)$$

It is not difficult to verify that the point $(z_1, \beta_1) = (\frac{a_1}{2}, \frac{1}{2} + \frac{\epsilon}{2(a_1 + \epsilon)})$ is an extreme point in that projection. Obviously, the point (z_1, β_1) does not satisfy (3.30). Thus, this inequality can be considered as a cut.

Furthermore, the following constraints proved improving the computational time of the proposed model.

Proposition 3.5. *The inequalities*

$$z_{l+1} - z_l \le (\beta_{l+1} - \beta_l)a_{l+1} \qquad \forall l \in \{1, ..., L-2\}$$
$$z_{l+1} - z_l \ge (\beta_{l+1} - \beta_l)(a_l + \epsilon) \qquad \forall l \in \{1, ..., L-2\}. \qquad (3.31)$$

are valid for conv(P^A).

Proof. First, we need to note that z_l is used to capture the value of $x\beta_l$. Then, from (3.19), (3.22) and the positivity of x, we see that:

$$0 \le z_1 \le ... \le z_{L-1} \le x. \qquad (3.32)$$

Now we have three cases for the binary variables β_{l+1} and β_l:

1. Both are zero. In this case, $z_{l+1} - z_l = 0$ which satisfies the two inequalities.

2. Both are one. In this case also, $z_{l+1} - z_l = 0$ which satisfies the two inequalities.

3. $\beta_l = 0$ and $\beta_{l+1} = 1$. In this case, we know from (3.18) that $x \in (a_l, a_{l+1}]$, or $a_l < x \le a_{l+1}$. From (3.22) one obtains $z_l = 0$ and $z_{l+1} = x$, hence the difference $a_l < z_{l+1} - z_l \le a_{l+1}$ as x.

Note also that the case $\beta_l = 1$, $\beta_{l+1} = 0$ is forbidden because of (3.19). $\qquad \square$

Proposition 3.6. *The inequalities in* (3.31) *define facets for conv(P^A).*

Proof. In Appendix A. $\qquad \square$

3.4 Numerical Results

To show the effectiveness of the proposed method, let us consider the functions $\tilde{f}_1(x_1)$ and $\tilde{f}_2(x_2)$ shown in Fig.3.1. The functions shown are not lower semi-continuous, and hence do not have global minima. However, the infima of $\tilde{f}_1(x_1)$ and $\tilde{f}_2(x_2)$ are clearly at $x_1 \longrightarrow 10^+$ and $x_2 \longrightarrow 4^+$, respectively. So, SOS2 method does not work in this case unless we modify the function as said earlier. We tried the proposed algorithm by minimizing over the sum of $\tilde{f}_1(x_1)$ and $\tilde{f}_2(x_2)$ subject to the constraints in (3.23), (3.18), (3.19), and (3.22). The minimization problem was solved by CPLEX solver in IBM ILOG CPLEX Studio V 12.5.1 on a PC with 3.0 GHz Intel(R) Core(TIM)2 Duo CPU and 4.0 GB RAM. The results obtained were ϵ close to the *infimizer*, namely, $x_1 = 10 + \epsilon$ and $x_2 = 4 + \epsilon$.

On the other hand, for continuous PWL functions the proposed algorithm proved to give same results as found by SOS2, but it takes longer time. To illustrate, consider the functions $\tilde{f}_3(x_3)$ and $\tilde{f}_4(x_4)$ in Fig.3.2. A minimization problem over the sum of the functions $\tilde{f}_3(x_3)$ and $\tilde{f}_4(x_4)$ was solved three times: one time by using the built-in function *piecewise* that uses the SOS2 method (first column in Table 3.1), another time by using the proposed formulation in P^A (second column in Table 3.1), and the last time by using the proposed formulation in P^A with the constraints in (3.31) (third column in Table 3.1). To increase the computational

(a) $\tilde{f}_1(x_1)$

(b) $\tilde{f}_2(x_2)$
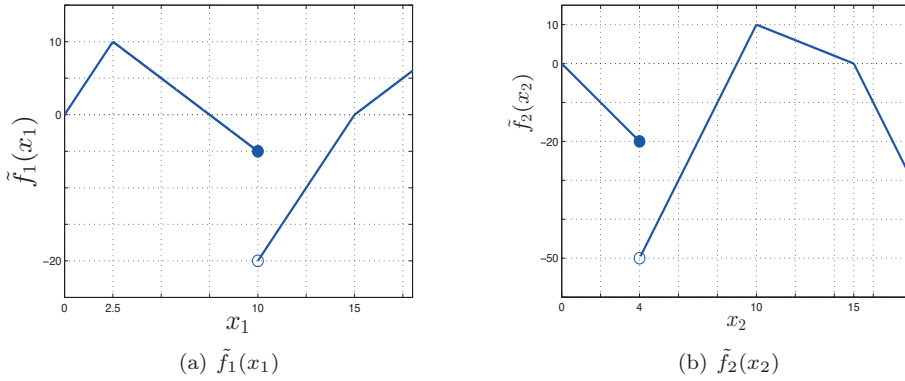
Figure 3.1: The functions used in the example.

complexity, the objective functions were replicated R times. The stopping criteria were set to defaults: below 0.01% relative MIP gap tolerance, or 1×10^{75}[s] time limit. The minima and optimizers found were the same by the two methods, that are obviously at $x_3 = 10$, and $x_4 = 18$. The computational times (CPU time not wall clock time) are recorded in Table 3.1.

(a) $\tilde{f}_3(x_3)$

(b) $\tilde{f}_4(x_4)$
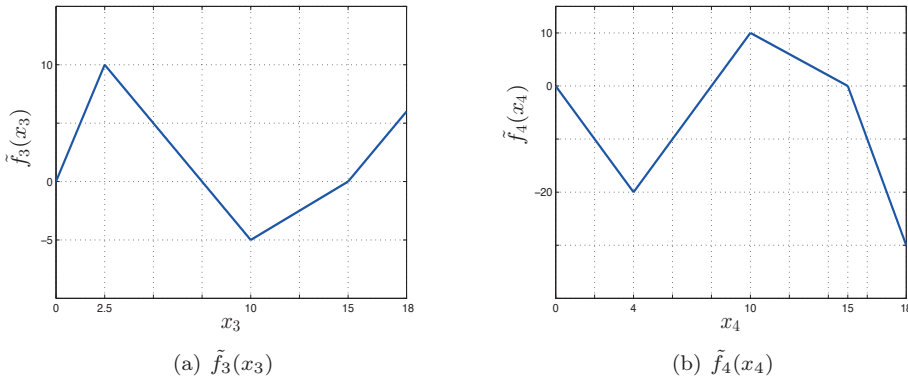
Figure 3.2: The functions used in the example

From Table 3.1 one can note that the proposed method is much more complex than the available techniques, especially SOS2. However, the strong inequalities in (3.31) have a major effect on reducing the computational time.

3.5   Conclusion

In this chapter, a technique was proposed for optimization over PWL functions. The formulation of the model was proved to be sharp, but not ideal. Further, the

R	CPU Time [s]		
	SOS2	by P^A	P^A with constrains (3.31)
10	0.02	0.08	0.02
100	0.08	0.85	0.41
1000	0.37	14.37	1.78
5,000	3.23	90.43	12.51
10,000	10.26	216.70	29.4

Table 3.1: Comparison between the computational times of the proposed algorithm and the standard built-in function.

convex-hull of the feasibility set was proved to be full-dimensional. While the commonly used models such as SOS2 necessitate lower semi-continuity, the proposed model can be used to approximate the infimum of PWL functions that are continuous from the left and not lower semi-continuous without the need to modify the PWL function. Some strong inequalities were found and proved to be strong, i.e. facet-defining or cutting planes. The numerical results show that the proposed inequalities have a strong impact on reducing the computational time. Hopefully, the research will continue in the future to find tighter formulation.

Chapter 4

Unit Commitment and Economic Dispatch

> If you don't have time to plan, do
> you have time to waste?
>
> ———————————
>
> P. Turla

The UC and ED problems are the most important problems in optimizing the operations of power systems. The increasing interest in those problems does not arise only from their essentiality for energy saving, maximizing profits, minimizing costs and exploiting renewable energy resources to the full extent, but also from the mathematical challenge they impose and the new horizons they can open in the field of operations research. In this chapter, we present the definition and purpose of the UC and ED problems. A comparison between the two problems is presented with the main contributions of the authors in this field, in the first section. The core MIP of the UC problem is introduced in the second section. Then, based on the previous work of the current authors in [106] and [105], a novel *dynamic* model to solve those problems is presented, analysed and evaluated in the remaining sections.

4.1 Introduction

In this section, UC and ED problems are introduced and compared. The main approaches used by the authors in this field are discussed, as well.

4.1.1 Unit Commitment and Economic Dispatch

UC problem is an optimization problem used to schedule the number of the committed generating units in a power system over some time horizon to meet the demand on the network at the least possible expense. On the other hand, ED problem is another optimization problem used to schedule the output power levels of the committed generating units over some time horizon to meet the demand at the minimum cost. The early works, on this topic, used to distinguish between the two problems. Thus, while UC aims to determine which units must be activated in

each time slot of the planning horizon [16], and [73], ED aims to determine the output power levels of each unit over the planning horizon assuming that the number of the committed units is determined in advance [113], and [29]. Actually, emphasizing the difference between the two terms depends very much on the technique used to solve the problem(s). For example, since the début of MIP the researchers using this technique treat the two problems as one. In opposition, the authors in [114] noted that the attention is drawn to separating the two problems. One possible reason for that is the fact that most of the power management and procuring companies still prefer to determine the committed units before their output levels. In general, UC problem is considered more complicated than ED problem because, usually, integer variables are used in the formulation [113]. Moreover, the complexity of those problems can arise drastically due to:

1. The existence of the energy storage device nowadays. A while ago, such techniques were not available, so the the researchers had to solve those problems assuming that the excess energy can not be stored.

2. The rapid change of prices of fuel. That was not taken in consideration, in this work.

3. Integrating intermittent energy sources such as wind and solar energy. This introduces stochastic variables to the problem.

4. Using different types of generators like thermal, hydro, and nuclear generating units in one network. This may lead to different objective functions and constraints.

5. The existence of private power companies that take care of distribution and procurement of energy. As investors, such companies are very keen on maximizing their profits and this leads to introducing economic variables that are highly stochastic. This gave rise to the emergence of the so-called *Self Unit Commitment* (self-UC) problem [28], and [93].

In fact, self-UC has been a hot topic in the field of smart grids. Assuming that power systems comprise power producing companies, power delivering companies, and *Independent System Operator* (ISO). Then, there will be a market for the electricity, in which the ISO receives bids from the production companies and offers from delivering companies, and then holds an auction to decide the hourly price [28]. The stochastic nature of the prices and the demand makes such problems much more complex than the usual UC.

In both problems, the demand on the power system is assumed known a priori. Power generation companies keep records of the demand on the network in different seasons and different times of the day. Hence, the demand can be predicted based on probabilistic calculations which are in most of the cases accurate.

4.1.2   Constraints

The constraints are used to guarantee appropriate working conditions for the the generating units and the distribution network. These constraints depend very much on the type of the generating unit, the purpose of that problem, and the network used.

1. Demand constraints, such as:
   a) *Power Balance*: to ensure that the sum of the output power levels at each time slot is enough to meet the demand and the losses in the network.
   b) *Power Generation*: to ensure that the output power level of each generating unit at each time slot does not exceed the limits recommended by the manufacturers.
   c) *Reserve*: to ensure some redundancy in the available power. In other words, power systems operators prefer to have some surplus power at hand in case of emergencies or in case the demand exceeds the predictions. In logistics jargon, such reserves may be called *inventory*. Reserves can be classified according to the status of the unit, into two classes [60]:
      i. *Spinning*: that can be provided by committed units within a time period much less than the length of the time slot.
      ii. *Non-spinning*: that can be provided by uncommitted units within a time period much less than the length of the time slot.

2. Unit constraints, such as:
   a) *Ramping-Up and Down* for thermal units: to ensure that the change of the output power level between successive time slots is within the limits recommended by the manufacturers. As explained in [110], the frequent turning on/off of the generators leads to the rotor fatigue and shortens the lifetime of the generating units, hence some constraints must be set to these changes to avoid rotor fatigue.
   b) *Minimum Up-Time and Down-Time* for thermal units: to ensure that if the unit is tuned on (off), it should stay on (off) for a minimum time called minimum up-time (minimum down-time). These constraints must be used because 'thermal units can only undergo gradual temperature changes [113].' Those constraints are important to reduce the maintenance costs, because too often switching of a unit would increase the thermal stress [75].
   c) Hydraulic constraints for hydroelectric units, such as: starting and ending reservoir volumes, total water discharge, and flow limits [113].

3. Distribution or transmission constraints. These are used to ensure the active and reactive power levels and voltage levels at each bus in the transmission network are optimal. Such constraints are usually called OPF constraints, see e.g. [113], and [22].

4. Economic constraints, which involve but not limited to:
   a) The constraints used to model the prices and revenues [28] for the self-UC.
   b) Fuel constraints. In some cases the amount of the fuel available is limited or the availability depends on the prices. In such cases extra constraints can be added [98].
   c) Crew constraints. In some cases the operators staff is limited, so turning the generating units on or off can not be done but for a limited number of units [113].

 d) Maintenance constraints. In some cases, especially for isolated power systems, periodic maintenance of the units should be respected while solving for the optimal scheduling of the units [48].

5. Other constraints, especially for the MIP formulations in which more constraints can be added to improve the description of the polytopes of the feasibility region, such as the *Logical Constraints* [73].

4.1.3 Objective Function

The objective function in UC and ED depends on the purpose and formulation of problems, as well. It depends also on whether the problems are considered separately or not. Further, it depends on the type of the generating units used in the system. Basically, the objective function involves:

1. Fuel consumption in case of thermal generating units. The fuel consumption function depends on the type of the generating unit, but for units driven by steam turbines the fuel cost function can be approximated by a convex quadratic function of the output power level, as [113]:

$$c^F(p) = ap^2 + bp + c, \tag{4.1}$$

 where a, b, and c are polynomial coefficients. The term c is usually referred to as the *no-load* cost, since this cost does not depend on the output power. Other fuel consumption functions have been used for other types of units.

2. Start-up cost for thermal units. The start-up cost depends on the type and size of the thermal unit used. For units driven by steam turbines, the start-up cost depends on the time the unit has been left off. Accordingly, it is modelled as an exponential function, as will be discussed in subsequent sections. On the other hand, for small-sized thermal units, the start-up cost can be considered fixed or even neglected.

3. Shut-down cost which is usually considered fixed or neglected regardless of the type of the unit.

4. Maintenance cost. In most of the cases, the maintenance cost is included in the fuel cost function in (4.1) [97]. In other cases, it can be modelled separately as a linear function of the output power level [79]. Generally speaking, the maintenance cost increases with the frequent switching of the units, so it can be included in the start-up cost.

Remark 4.1. If the ED problem is to be solved separately, start-up and shut-down costs are not considered, because in this case the committed units is assumed known a priori, as mentioned before.

Remark 4.2. For self-UC problem, the target is to maximize the profits. Hence, the objective function will be the difference between the revenue and the total production cost which may involve some or all of the aforementioned costs.

4.1.4 State-of-The-Art

A plethora of approaches has been suggested in literature to formulate and solve UC and ED. Early engineers in power generation companies used to depend on a PL prepared a priori to schedule the output power levels of the thermal units [29]. This method is still used extensively in the field. Then, DP technique was suggested based on the *Principle of Optimality*, see [113] and the references therein, and it is still in use [79], especially for solving ED separately. Later, LR was benefited to simplify the UC problem by resolving the coupling constraints [113], and hence DP solutions can be found more easily. In their pioneer work, the authors in [29], and [73] suggested IP or, more precisely, MIP to solve UC. Many numerical methods have been suggested to solve MIP, whether MILP, or MILNP including [8]: B&B, *Decomposition*, CP, and OA, as explained in Chapter 2. Spurred by the progress in the technology of processors and optimization solvers such as CPLEX, GAMS, XPRESS, and BARON many authors used the MIP to formulate the UC problem for on-land power systems, see e.g. [77], [2], [33], and [96].

AI algorithms contribute, as well, to the attractive tools used by researchers to solve UC. Many AI algorithms were suggested and used in literature to solve this problem such as: GA [65], and [20], *Tabu Search* [67], *Particle Swam Optimization* (PSO) [100], *Artificial Bee Colony* [15], *Fuzzy Logic* [89], and SA [116]. For detailed survey on the AI algorithms used to solve UC, the reader is advised to read the excellent surveys in [91], and [79].

4.2 Mixed-Integer Programming for UC and ED

In this section, the core MIP that has been used by the authors to solve UC and ED problems considering them as one problem is introduced. We consider thermal units only, so the hydraulic constraints are not required. Besides, the self-UC is beyond the scope of this thesis, thus the economic constraint shall not be discussed.

Let us start by defining appropriate index sets that we need throughout this work. Let $\mathcal{J} = \{1, ..., J\}$ be the generating units index set, assuming that there are J generating units in the power system of the same type. Let also $\mathcal{K} = \{1, ..., K\}$ be the set of the time slots that spans the planning horizon. Usually, each time slot is assumed to be one hour long, and the planning horizon may vary from one day to one week. Then, the following binary variables are defined: $\alpha_j(k)$, $\forall j \in \mathcal{J}, \forall k \in \mathcal{K}$, such that:

$$\alpha_j(k) = \left\{ \begin{array}{ll} 1 & \text{if unit } j \text{ is on during slot } k \\ 0 & \text{otherwise} \end{array} \right\}, \tag{4.2}$$

$\beta_j(k)$, $\forall j \in \mathcal{J}, \forall k \in \mathcal{K}$, such that:

$$\beta_j(k) = \left\{ \begin{array}{ll} 1 & \text{if unit } j \text{ is turned on at the beginning of slot } k \\ 0 & \text{otherwise} \end{array} \right\}, \tag{4.3}$$

and finally, $\gamma_j(k)$, $\forall j \in \mathcal{J}, \forall k \in \mathcal{K}$ such that:

$$\gamma_j(k) = \left\{ \begin{array}{ll} 1 & \text{if unit } j \text{ is turned off at the beginning of slot } k \\ 0 & \text{otherwise} \end{array} \right\}. \tag{4.4}$$

4.2.1 Constraints

Now we are ready to present the basic MIP of the constraints and cost functions explained in the previous section. Table 4.1 lists all possible combinations of the above binary variables, and the desired status of the unit in each case.

$\alpha_j(k-1)$	$\beta_j(k)$	$\gamma_j(k)$	$\alpha_j(k)$
0	0	0	0
0	0	1	Invalid
0	1	0	1
0	1	1	Invalid
1	0	0	1
1	0	1	0
1	1	0	Invalid
1	1	1	Invalid

Table 4.1: Desired status of the unit for all possible combinations of the binary variables

1. Logical constraint. From Table 4.1, the following logical constraint can be stated:
$$\alpha_j(k) - \alpha_j(k-1) = \beta_j(k) - \gamma_j(k), \ \forall j \in \mathcal{J}, \forall k \in \mathcal{K}. \tag{4.5}$$

   Note that for this constraint the initial status of the unit $\alpha_j(0)$ must be known. Actually, the constraint in (4.5) has been used since the beginning of the MIP for UC, see e.g. [29].

2. Power generation. Let $p_j(k)$ be the output power level of unit j in the kth time slot, $\overline{P}_j$ be the maximum allowed output power level, and $\underline{P}_j$ be the minimum allowed output power level, then the power generation constraint can be described as:

$$\underline{P}_j \alpha_j(k) \leq p_j(k) \leq \overline{P}_j \alpha_j(k), \ \forall j \in \mathcal{J}, \forall k \in \mathcal{K}. \tag{4.6}$$

3. Power balance. Let $D(k)$ denote the total demand on the network during the kth time slot and $D_{loss}(k)$ be the expected losses in the power system, then the power balance constraint is expressed as:

$$\sum_{j=1}^{J} p_j(k) = D(k) + D_{loss}(k), \ \forall k \in \mathcal{K} \tag{4.7}$$

4. Spinning reserve. The authors on the UC problem in on-land power systems, usually assume that the total spinning reserve for a certain system during the kth time slot is calculated by using heuristics, e.g. a percentage of the total demand [77]. Others prefer to calculate the spinning reserve by probabilistic methods [32]. Anyway, let the total spinning reserve on the network $SR(k)$ be given. Then, one way to model the spinning reserve constraint is to assign

a decision variable $r_j^s(k)$ that represents the spinning reserve share scheduled for unit j in the kth time slot, then the spinning reserve constraints can be forced by the following inequalities, see e.g. [38]:

$$\underline{P}_j \alpha_j(k) \leq p_j(k) + r_j^s(k) \leq \overline{P}_j \alpha_j(k),$$
$$0 \leq r_j^s(k) \leq RU_j \alpha_j(k), \ \forall j \in \mathcal{J}, \forall k \in \mathcal{K}$$
$$\sum_{j=1}^{J} r_j^s(k) \geq SR(k), \ \forall k \in \mathcal{K}. \tag{4.8}$$

The first constraint in (4.8) ensures that not all the units are working at the maximum allowed level, thus during any time slot some margins are left for the operators to procure more energy in case of emergencies. The second constraint in (4.8) while forcing $r_j^s(k)$ to be zero when the unit is off, i.e. $\alpha_j(k) = 0$, it allows $r_j^s(k)$ to vary up to the upper bound RU_j which is the upper bound of the ramping rate, as will be explained later. The last constraint in (4.8) guarantees that the sum of these left margins are enough to meet the the total spinning reserve required.

Another way to impose the spinning reserve constraint can be found in [77], for example. In this method, a decision variable is assigned for the maximum allowed output level of unit j, during the kth slot, and let it be $\overline{p}_j(k)$. Then, the following constraints can be used:

$$\underline{P}_j \alpha_j(k) \leq p_j(k) \leq \overline{p}_j(k) \leq \overline{P}_j \alpha_j(k), \ \forall j \in \mathcal{J}, \forall k \in \mathcal{K}$$
$$\sum_{j=1}^{J} \overline{p}_j(k) \geq D(k) + D_{loss}(k) + SR(k), \ \forall k \in \mathcal{K}. \tag{4.9}$$

5. Ramping constraints. As explained earlier, the change of the output power level, whether increase or decrease, between any two successive time slots must be bounded. Thus, the basic ramping constraint can be described as follows:

$$-\Delta_T RD_j \leq p_j(k) - p_j(k-1) \leq \Delta_T RU_j, \ \forall j \in \mathcal{J}, \forall k \in \mathcal{K}, \tag{4.10}$$

where RU_j and RD_j denote the maximum and minimum allowed ramping rate, respectively, and Δ_T is the length of one time slot. Note that for this constraint the initial power level of the unit $p_j(0)$ must be known, just like in the constraint (4.5). Recently, more restrictive constraints are considered for the ramping constraints before turning on from off state and vice versa, and this is one of the challenges in the UC problem, see e.g. [3], and [13].

6. Minimum up-time and down-time. As explained earlier, if the unit is turned on (off) it should remain on (off) for at least minimum up-time (down-time), denoted by UT_j (DT_j). Those constraints are 'so critical and they are one of the main reasons why UC problem is hard to solve as MILP [85].' In the surveys [91] and [79], no general construction of those constraints was documented. The authors in [2], proposed linear inequalities for the minimum

up and down times constraints based on the binary variables $\alpha_j(k)$, $\beta_j(k)$ and $\gamma_j(k)$. Then, the authors in [13] modified those constraints by using the binary variable $\alpha_j(k)$, only. Another formulation was suggested in [98]. Later, the authors in [58] described what they called *alternating up/down inequalities* to tighten the formulation. Later, the authors in [85] proposed alternative constraints based on what they called *turn on/off inequalities*, and they proved that those inequalities with some trivial logical constraints give tighter formulation than the alternating up/down inequalities. The problem with most of the proposed constraints is that they do not take in consideration the initial states required for the minimum up/down time constraints. That is to say, if a unit has been committed for t time slots before the beginning of the planning horizon, and this t is less than DT, then the unit should stay on for at least the first $DT - t$ time slots of the planning horizon before it is turned off. As far as we know, the only formulation that considers this initial state is the one presented in [77]:

$$\sum_{i=k}^{k+UT_j-1} \alpha_j(i) \geq UT_j\beta_j(k), \quad \forall k \in \{L_j + 1, ..., K - UT_j + 1\}, \ \forall j \in \mathcal{J}$$

$$\sum_{i=k}^{k+DT_j-1} (1 - \alpha_j(i)) \geq DT_j\gamma_j(k), \quad \forall k \in \{F_j + 1, ..., K - DT_j + 1\}, \ \forall j \in \mathcal{J}$$

$$\sum_{i=k}^{K} (\alpha_j(i) - \beta_j(k)) \geq 0, \quad \forall k \in \{K - UT_j + 1, ..., K\}, \ \forall j \in \mathcal{J}$$

$$\sum_{i=k}^{K} (1 - \alpha_j(i) - \gamma_j(k)) \geq 0, \quad \forall k \in \{K - DT_j + 1, ..., K\}, \ \forall j \in \mathcal{J}$$

$$\sum_{i=k}^{F_j} \alpha_j(i) = 0, \quad \forall j \in \mathcal{J}$$

$$\sum_{i=k}^{L_j} \alpha_j(i) = L_j, \quad \forall j \in \mathcal{J}, \quad (4.11)$$

where $F_j = \min\{K, D_j\}$ and $L_j = \min\{K, U_j\}$, with D_j (U_j) is the time slots for which unit j is required to be off (on) at the beginning of the planning horizon. These parameters represent the initial values or states needed for the minimum up/down time constraints. Actually, with these parameters, all minimum up/down constraints can be changed to take in consideration the initial states. However, these initial values should be updated every time we repeat the optimization, unless they are stored in state-space form, as will be shown later.

7. OPF constraints. In power systems, the buses or nodes can be classified into: slack node, $P - Q$ node, and $P - V$ node. Denoting the active and reactive power entering a node by P^{Net} and Q^{Net}, respectively, and denoting the voltage at the node and its angle by V and θ, respectively, the following definitions can be stated [22]:

a) Slack node is the node where V and θ are specified, while P^{Net} and Q^{Net} are not.

b) $P - Q$ node is the one with P^{Net} and Q^{Net} are specified, while V and θ are not.

c) $P - V$ node is the one with P^{Net} and V are specified, while θ and Q^{Net} are not.

Then, the OPF constraints are used to ensure that the injected net real and reactive power at each bus is equal to the real and reactive power consumed from that bus. This can be achieved by finding suitable functions for the unknown quantities at each bus from the specified quantities, depending on the type of the bus. OPF problem is a hot topic in power system stability and control by itself, and the literature on this topic is vast. For further details on the OPF constraints, the reader is referred to [79], and [22], but those constraints will be ignored throughout this work.

4.2.2 Objective Function

We restrict ourselves in this work to fuel consumption function and start-up cost only.

1. Fuel cost function. The function in (4.1) is usually divided into two parts. The first is the no-load cost c_j, that is the cost which does not depend on the output power level and it will be paid as long as the unit is committed. The other part is the generation cost $a_j p_j(k)^2 + b_j p_j(k)$ which depends on the output power level. Thus when the units is committed without loading, i.e. $p_j(k) = 0$, this cost will be zero. We can see that the no-load cost c_j will not be driven to zero when $\alpha_j(k) = 0$, whereas the power level $p_j(k)$ will be driven to zero by the constraint in (4.6). So, the fuel consumption function is usually modified to [26], [3]:

$$c_j^F(p_j(k)) = a_j p_j(k)^2 + b_j p_j(k) + c_j \alpha_j(k). \tag{4.12}$$

Actually, the fuel consumption function in (4.1) is quadratic, and usually convex, according to the authors in [113]. This encouraged the researchers to solve the combined UC and ED as a MIQP, presuming that the convexity of the objective function simplifies the problem. On the other hand, some prefer to formulate it as MILP by using the PWL approximation of the generation function $a_j p_j(k)^2 + b_j p_j(k)$. Denoting the PWL approximation by $\widetilde{c_j^G}$, the fuel consumption function can be rewritten as:

$$c_j^F(p_j(k)) = \widetilde{c_j^G} + c_j \alpha_j(k). \tag{4.13}$$

2. Start-up cost. As stated before, the start-up cost denoted here by $c_j^{SU}(k)$ is modelled as an exponential function of the form [110]:

$$c_j^{SU}(k) = Hc_j + (cc_j - Hc_j)(1 - e^{-x_j^{off}(k)/\tau_j}), \tag{4.14}$$

where τ_j is a time constant, Hc_j is the minimum start-up cost when the unit is still hot, cc_j is the maximum start-up cost when the unit gets completely cold,

see Fig.4.1, and $x_j^{off}(k)$ is the time period for which the unit has been left off measured in slots. The minimum start-up cost Hc_j can be considered zero, without loss of generality. This model of the start-up cost is very complicated to deal with due to the non-linearity and non-convexity of the exponential function, and the dependence on the variable $x_j^{off}(k)$. The common way to capture the start-up cost in MIP in the literature on UC (see e.g. [13] and [77]) is to discretize the exponential function in (4.14) into K_j^t steps $\forall t \in \{1, .., ND_j\}, \forall j \in \mathcal{J}$ as shown in Fig. 4.1, where ND_j is the period over which the start-up cost function is discretized in time slots. Then, the start-up cost is considered as a decision variable formulated as [75]:

$$c_j^{SU}(k) = \max_{t=1,..,ND} K_j^t \left(\alpha_j(k) - \sum_{i=1}^{t} \alpha_j(k-i) \right). \tag{4.15}$$

Note that the equation above does not include $x_j^{off}(k)$, so this model does not require counting the off time slots in advance. Instead, it looks through the past slots for the maximum start-up cost to be included in the cost function to be minimized. Based on (4.15), the start-up cost decision variable is bounded by the following constraints [13] and [77]:

$$c_j^{SU}(k) \quad \geq \quad K_j^t \left[\alpha_j(k) - \sum_{i=1}^{t} \alpha_j(k-i) \right],$$
$$\forall j \in \mathcal{J}, \quad \forall k \in \{\mathcal{K} : k > t\}, \quad \forall t \in \{1, ..., ND_j\}$$
$$c_j^{SU}(k) \quad \geq \quad 0, \quad \forall j \in \mathcal{J}, \forall k \in \mathcal{K}. \tag{4.16}$$

The current authors proposed another technique to capture the start-up cost in [101]. The start-up cost was described in [101] by a set of if statements. Then, the conditional statements were transformed into $bigM$ inequalities to define the feasibility region, see [101] for more details. However, we present in this work an alternative technique in later sections that captures the start-up cost even more tightly. The discrete start-up steps K_j^t are, usually, assumed to take two values only: hc_j for $t \leq DT_j + HT_j$, and cc_j for $DT_j + HT_j < t \leq ND_j$ [13] and [77], where HT_j is the time during which the unit is still considered hot.

4.2.3   Ramping Process

The ramping constraints in (4.10) restrict the ramping up and down rates of the generating unit from exceeding the limits RU and RD, respectively. Those limits in (4.10) are assumed fixed. This may not be realistic since the ramping rates limits may vary according to the generation output. Recently, more restrictive models of the ramping process have been proposed. The authors in [3] proposed a MILP to describe the output power trajectory during start-up (shut-down) process by assuming that the unit should increase (decrease) to (from) the minimum (maximum) allowed power level $\underline{P}$ ($\overline{P}$) gradually, when it is started-up (shut-down) from the off (on) state. Later, the authors in [61] differentiate among three types of ramping
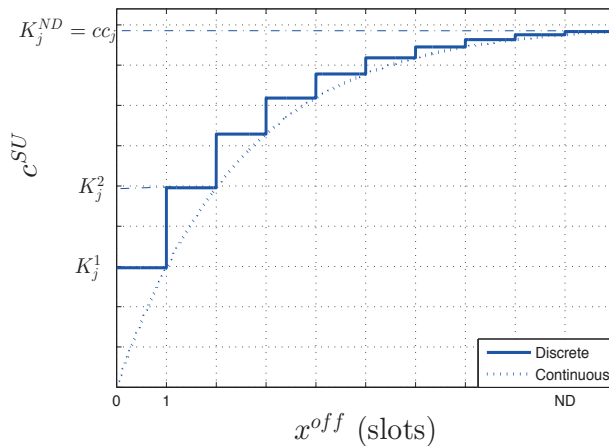
Figure 4.1: Exponential start-up cost (dashed) and discretized (solid).

rates limits: fixed, stepwise and PWL function of the generation output. Based on the latter option the authors proposed a dynamic ramping model [61]. Afterwards, another MILP was suggested in [93], in which the authors divided the power generation sequence of the thermal unit into four phases: synchronization, soak, dispatch and de-synchronization. Besides, they considered three start-up types depending on the prior reservation time: hot, warm and cold. Then, the formulation was proposed to enforce the ramping process to pass through the four phases of the sequence [93]. Finally, the authors in [77] proposed some strong inequalities to tighten the formulation of the ramping process assuming that the ramping rates limits are modelled as discrete steps.

Throughout this work, it was presumed sufficient to assume two discrete steps of the ramping rate limits during start-up or shut-down and during normal dispatch. So, when the unit is to be turned on it must not be loaded with more than a maximum allowed start-up ramping up rate, which we denote here by R_j^{SU}. Similarly, a unit must not be shut down when it is loaded by more than the maximum allowed shut-down ramping down rate R_j^{SD}. We use the model proposed in [77], described as:

$$p_j(k) - p_j(k-1) \leq \Delta_T \left[RU_j \alpha_j(k-1) + R_j^{SU} \beta_j(k) \right]$$
$$p_j(k) - p_j(k-1) \geq -\Delta_T \left[RD_j \alpha_j(k) + R_j^{SD} \gamma_j(k) \right]$$
$$\forall j \in \mathcal{J}, \forall k \in \mathcal{K}. \qquad (4.17)$$

In order to understand how the constraints above enforce different ramping rate limits, it is enough to recall from Table 4.1 that there 4 possible cases for the two successive binary indicators $\alpha_j(k)$ and $\alpha_j(k-1)$:

1. When $\alpha_j(k-1) = \alpha_j(k) = 1$. The unit stays on during the successive time slots and both variables $\beta_j(k)$ and $\gamma_j(k)$ are zero. Hence, the up and down ramping rate limits will be RU_j and $-RD_j$, respectively, as in (4.10).

2. When $\alpha_j(k-1) = \alpha_j(k) = 0$. The unit stays off during the successive time slots and and both variables $\beta_j(k)$ and $\gamma_j(k)$ are zero. Hence, the up and down ramping rate limits will be RU_j and $-RD_j$, respectively, as in (4.10).

3. When $\alpha_j(k-1) = 0$ and $\alpha_j(k) = 1$. The unit is off and is to be turned on in the kth slot implying that $\beta_j(k) = 1$ and $\gamma_j(k) = 0$. The ramping up limit will be R_j^{SU} from the right-hand side of the inequality in (4.17).

4. When $\alpha_j(k-1) = 1$ and $\alpha_j(k) = 0$. The unit is on and is to be turned off in the kth slot implying that $\gamma_j(k) = 1$ and $\beta_j(k) = 0$. The ramping down limit will be $-R_j^{SD}$ from the left-hand side of the inequality in (4.17).

However, the authors in [77] used $\bar{p}_j(k)$ in lieu of $p_j(k)$ in the constraints above because they used the constraints in (4.9) to model the spinning reserve. Having mentioned that, more restrictive constraints can be imposed on spinning reserve. De facto, the initial ramping rates when the unit is to be committed and final ramping rates when the unit is to be shut down should not be violated. So, the spinning reserve should not be scheduled for units when they are to be committed or de-committed. Hence, the constraints in (4.8) can be modified to:

$$\underline{P}_j\alpha_j(k) \leq p_j(k) + r_j^s(k) \leq \overline{P}_j\alpha_j(k)$$
$$0 \leq r_j^s(k) \leq RU_j\alpha_j(k)$$
$$0 \leq r_j^s(k) \leq RU_j\alpha_j(k-1),$$
$$\forall j \in \mathcal{J}, \ \forall k \in \mathcal{K}. \tag{4.18}$$

In order to see how the constraint above work, it suffices to note that the spinning reserve $r_j^s(k)$ will be zero, if one of the binary indicators $\alpha_j(k)$ and $\alpha_j(k-1)$ or both are zero. So, when $\beta_j(k) = 1$, the unit is to be committed in the kth slot, implying that $\alpha_j(k-1) = 0$ and $\alpha_j(k) = 1$, hence $r_j^s(k) = 0$ from the second inequality in (4.18). Moreover, when $\gamma_j(k) = 1$, the unit is to be shut down in the kth slot, implying that $\alpha_j(k-1) = 1$ and $\alpha_j(k) = 0$, hence, $r_j^s(k) = 0$ from the first inequality in (4.18). Thus, spinning reserves are not scheduled to units that during start-up or shut-down.

To conclude, the core MIP to solve both the UC and ED problems was introduced to be used in comparison with the proposed model in later sections.

4.3 Dynamic vs. Static Models

What we mean by the static model here is the one without differential (difference) equations in continuous (discrete) time, and we refer to models with differential or difference equations as dynamic models. In this section, some literature review on dynamic models for UC and\or ED is presented. Then, we discuss the potential reasons and motivations to use dynamic formulations.

Although power systems are dynamic in nature, the researchers in the field of scheduling and planning prefer to stick to the static model, especially when it comes to the MILP. In fact, the ramping rates constraints in (4.17) couple the output power levels, and thus make the ED a dynamic process. Inspired by this fact, the authors in [57] proposed a dynamic model of the generation process and

tried to combine the ED problem with the AGC problem by adding to cost function the deviation between the sum of the power levels and the total demand on the network, as follows:

$$\min_{u_j(t)} \int_0^T q_1 (\sum p_j^c - D)^2 + q_2 \sum h_j(p_j^c) + q_3 \sum m_j^h |u_j|^{\sigma_j}, \qquad (4.19)$$

subject to

$$\frac{dp_j^c}{dt} = u_j$$
$$-U_j^c \le u_j(k) \le U_j^c, \qquad (4.20)$$

where D is the total demand on the network, $h_j(p_j^c)$ is the steady-state heat rate characteristics which is equivalent to the fuel consumption function, U_j^c is the upper limit on the control input u_j, m_j^h and σ_j are parameters to describe the cost of the control input which is equivalent to the ramping cost, T is the control horizon, and $q_1 - q_3$ are weighting factors. So, the output power level is modelled as a continuous state variable with the dynamics in the first equation in (4.20), and the problem is solved in the *Optimal Control* (OC) philosophy. The model above may not be suitable for scheduling purposes because of the control horizon T. That is to say, this model aims to find the optimal control input $u_j^*(t)$ that drives the power level trajectory p_j^c from an initial value $p_j(0)$ till the final value $p_j(T)$ at the end of the control horizon to meet the fixed demand D. So, this formulation may be suitable after solving the scheduling problem, in order to achieve the AGC target, which is to keep the power level close to the demand in case the latter varies by small amount.

A few years later, the authors in [88] proposed a dynamic model to solve ED only, as:

$$p_j(k+1) = p_j(k) + \Delta_T r_j(k), \qquad (4.21)$$

subject to the constraints:

$$-RD_j \le r_j(k) \le RU_j, \ \forall j \in \mathcal{J}, \ \forall k \in \{0, ..., K-1\} \qquad (4.22)$$

and

$$\underline{P}_j \le p_j(k) \le \overline{P}_j, \ \forall j \in \mathcal{J}, \ \forall k \in \mathcal{K}, \qquad (4.23)$$

and the power balance constraint in (4.7), where $r_j(k)$ is the ramping rate, Δ_T, RU_j and RD_j are as before. This gave rise to the so-called *Dynamic Economic Dispatch* (DED) problem. In the above model, the ramping process is translated into a linear difference equation considering the output power level $p_j(k)$ as a discrete state variable and the ramping amount as a control input. Thus, the problem can be solved in OC philosophy by minimizing the fuel consumption given in (4.12), assuming that the initial states $p_j(0)$, $\forall j$ are given, just like the MILP explained in the previous section. Actually, the above model is not different from the constraints in (4.17) if we make the transformation $r_j(k) = p_j(k) - p_j(k-1)$. The only difference between the two models is that the difference equation in (4.21) and constraint in (4.23) do not include binary indicators and hence presume that the number

of committed units is known in advance by solving the UC, separately. So, the target of this problem is to schedule the available or committed generating units to meet the demand at the minimum cost respecting the network constraints such as the OPF constraints and ramping constraints. Since then, many authors tried to contributed to the DED problem, see e.g. [38], [37] and [63]. Other authors such as in [99], proposed pricing techniques based on the DED model. Recently, some authors try to formulate ED models suitable for MPC philosophy, such as [114]. In fact, all of the aforementioned authors considered the ED problem only without relating it to the UC problem. In contrast, some authors on this topic proposed solving the combined UC and ED problems in a dynamic model, e.g. [81] and [24], as will be discussed later.

The question which poses itself here is: Why do we need a dynamic model?

As mentioned before, power systems are dynamic in nature, so the best answer to the above question depends very much on the purpose of the model, and what states are considered. In the following subsections we discuss the potential reasons to use dynamic models.

4.3.1   States and Dynamics

For scheduling purposes, the decision variables of interest are the output power level $p_j(k)$ and the binary status indicator $\alpha_j(k)$. The length of the time slot considered is much longer than what is required for the dynamics of the generating units to take actions. To elaborate, the time slot is usually taken to be one hour, some authors assumed that it can be decreased to 15min, but even the 15min may be longer than the time needed to ramp up or down the unit, depending on its type and size. That is why the small variations in demand, for scheduling engineers, are neglected because they can be taken care of by the AGC. Thus, static MILP without dynamic states are fair enough for scheduling purposes, as long as the small variations can be handled by the AGC. However, there are some other situations in which more states in power systems can be of interest, such as but not limited to:

1. More accurate dynamic models of the ramping process of generating units. If the time slot in the planning horizon is to be shortened to less than 10 or 5 min, then more accurate dynamics for the ramping are needed.

2. More accurate dynamic models of the demand, spinning reserve and losses in the network. In some cases, especially for isolated power systems, the demand can be predicted more precisely if dynamic models are involved.

3. Prices and spot markets. The prices of electricity in a spot market keeps changing with time, see e.g. [99] and the references therein.

4. Different types of generating units, especially if renewable energy resources are exploited. Renewable resources like wind and solar energy are intermittent highly stochastic. Thus, dynamic models may be needed to describe such units, see e.g. [81] and [34].

5. Storage units and combined cycles. Such units need require advanced models for the stored or delivered power, which may be hard to handle with the normal MILP, see e.g. [62] and [14]. Some authors, such as [63] and [111],

proposed a dynamic model of power grids that involve controllable and uncontrollable devices, whether they store or deliver power. However, their model did not involve solving the UC problem.

4.3.2 Model Predictive Control

Another reason that can be thought of to propose dynamic models is to simplify the MPC or *receding horizon* strategy. In MPC an open-loop OC problem is solved at each time slot using the current states as initial states, then the first element of the obtained control sequence is applied to the plant and the process is repeated [69]. MPC philosophy is attractive for industry in general due to its ability to handle hard constraints and control problems in which the control law is hard to find [86]. For scheduling problems involving integer variables such as UC, MPC looks even more interesting for the following reasons:

1. Optimization problems involving integer variables formulated for infinite horizon are inappropriate theoretically and practically [7]. Since in MPC the optimization problem is solved on-line in each time step repeatedly for the next K time slots (the planning horizon), the scheduling recedes over the original planning horizon. Thus, the infinite horizon scheduling can be achieved implicitly.

2. The expected demand and spinning reserve are usually assumed known a priori for the whole planning horizon, as explained earlier. Although this can be a good simplification for large power systems due to the periodicity of the demand and its dependence on seasons and times, this assumption may not be valid for isolated power systems where the demand may vary significantly and unexpectedly. Besides, this is not granted when small variations of the demand is to be considered. Thus, by using MPC, this problem can be solved because the expectation can be updated each time slot.

3. The dynamic models of the ED and UC have no equilibrium points in the physical sense, assuming that we have a rigorous dynamic model. So, stability is not a key issue in the MPC strategy when used to solve the ED and UC problems. Instead, non-emptiness of the feasibility region of the optimization problem has to be guaranteed to ensure solvability.

The authors in [114] stated that the MPC philosophy has been used by some power companies to solve ED problem in a "primitive way". The example explained in [114] about using the MPC is about a company which solves the ED problem for 7-day horizon, at the end of each day the problem is solved again for the next 7-day, and so on. The authors in [114] highlighted a very important drawback in solving ED problem, namely the violation of the ramping rates. The ramping rates constraints can be violated at the beginning of the planning horizon because of the lack of information about the initial states of the generating units, or at the end of the planning horizon because the last time slot in the planning horizon is not correlated to the first time slot of the next planning horizon. In addition, the authors in [114] showed that while the dynamic model of the ED problem may suffer from the violation of the ramping constraints at the end of the horizon, the static model of the ED problem may result in violation of ramping constraints at

the beginning of the horizon. Thus, they proposed a MPC strategy to solve the ED problem to guarantee that the ramping constraints are always satisfied. The algorithm proposed in [114] is elegant, but it neglects the UC problem and assumes that the number of the committed generating units is known in advance, and of course since the demand is assumed to have some form of periodicity, no extra units will be required to be committed.

Actually, the reason behind this problem of violating the ramping rates pointed out by the authors in [114] and explained above are questionable. Many authors in the field of static models overcame this problem by storing the initial states of the generating units in special variables to be used in the ramping constraints as in (4.17), this was done, e.g. in [72]. So, dynamic models are not better than static ones for exploiting MPC. However, we believe that updating the states in dynamic models would be easier due to the differential or difference equations which can be used to update the states more easily.

4.3.3   Better Models

Increasing the number of the variables and constraints in any model does not necessarily mean that it gets worse. When it comes to IP and MIP, it is essential to use combinatorial optimization tools to compare between formulations. Thus, exploiting more variables as states and proposing dynamics in difference or differential equations do not increase the burden of the computational complexity in all cases. On the contrary, dynamic MIP may be tighter and more accurate than static MIP, as will be shown in the proposed model in the next section.

Recently, many authors have proposed novel dynamic models for scheduling and planning in power systems. The authors in [24] proposed a dynamic MIP model of the UC taking in consideration the ED. In their model in [24], they proposed four states for the system, the output power level $p_j(k)$, binary status indicator of each unit $\alpha_j(k)$, and two other states to store the time slots for which the unit has been left on and off. Then, they used a transformation based on the transformation proposed in [30] to convert the problem into a usual OC problem. Then, the authors in [34] proposed a dynamic MIP for the UC problem for a wind farm. In spite of the novelty of the proposed model, it is designed for wind turbines only, hence the ramping of the thermal units is not considered. Besides, the authors in [34] did not provide any analysis regarding the feasibility region using combinatorics tools. One can find another dynamic model in [111] that models the injection and absorption of power at each bus in a network which contains renewable energy resources. Further, the authors used the proposed model [111] not only to schedule the generating and storage units, but also to find what they called *affine policy*, which they defined as a 'series of planned linear modifications that depend on the prediction errors that will become known at future times [111].' Although they did not include the UC in their model in [111], they claimed that it can be included through a MLD model, without any further details. Finally, the authors in [81] proposed a dynamic model for micro grids that comprise generating units, storage devices and controllable loads. Then, they used the proposed model to formulate a MPC controller that aims to provide the optimal schedule of the storage devices and controllable loads, solve the UC and ED problems, and further decide how

much energy should be purchased or sold from or to the utility (main) grid [81]. The proposed model in [81] is quite impressive and collateral, nevertheless they used the core MIP presented in the previous section, without any improvements. At last, those were the most important contributions to the dynamic models to solve UC and\or ED problems. Besides, the possible motivations to propose a dynamic model to solve the combined UC and ED problems. In the next section, the proposed model is presented and discussed.

4.4 The Proposed Model

In this section, based on the previous work of the present authors in [106], we present the proposed dynamic MIP to solve the combined UC and ED problem, step by step. Each subsection describes a part of the proposed model with some numerical experiments for evaluation purposes.

4.4.1 Power Generation

To put a dynamic model comprising difference equations, we have to think of the required states. Obviously, to describe the power generation the status of the unit $\alpha_j(k)$ and the power level $p_j(k)$ represent the basic states of the sought model. We commence by re-defining the binary variables $\beta_j(k)$ and $\gamma_j(k)$ to be input controls that change the status of the unit $\alpha_j(k)$ in the dynamic model. So, let $\beta_j(k)$, $\forall j \in \mathcal{J}, \forall k \in \{0, ..., K-1\}$ be defined as:

$$\beta_j(k) = \left\{ \begin{array}{cc} 1 & \text{if unit } j \text{ is to be turned on in slot } k+1 \\ 0 & \text{otherwise,} \end{array} \right\} \qquad (4.24)$$

and, let also $\gamma_j(k)$, $\forall j \in \mathcal{J}, \forall k \in \{0, ..., K-1\}$ be such that:

$$\gamma_j(k) = \left\{ \begin{array}{cc} 1 & \text{if unit } j \text{ is to be turned off in slot } k+1 \\ 0 & \text{otherwise,} \end{array} \right\} \qquad (4.25)$$

then, considering the variables $\beta_j(k)$ and $\gamma_j(k)$ as control inputs, the logical constrain in (4.5) describes the dynamics of $\alpha_j(k)$, as:

$$\alpha_j(k+1) = \alpha_j(k) + \beta_j(k) - \gamma_j(k), \ \forall k \in \{0, ..., K-1\} \qquad (4.26)$$

assuming that feasible $\alpha_j(0)$ is given.
For the power generation trajectory, the model in (4.21) with the constraint in (4.22) can be used. However, the binary indicators used in UC problem and the ramping dynamics restrictions introduced in Subsection 4.2.3 have to be considered. Firstly, the constraint in (4.23) is replaced with (4.6) to involve the binary indicators. Then, the output power dynamics in (4.21) is modified to capture the ramping dynamics, as introduced in (4.17), to:

$$p_j(k+1) = p_j(k) + \Delta_T \left[r_j(k) + R_j^{SU}\beta_j(k) - R_j^{SD}\gamma_j(k) \right]. \qquad (4.27)$$

Since the constraint in (4.6) ensures that the power level will be driven to zero when the unit is turned off, the ramping input $r_j(k)$ is required to vary to accomplish

that. However, R_j^{SU} (R_j^{SD}) adds to (subtracts from) the ramping input when the unit is to be on (off). So, we need to make sure that the ramping rate input $r_j(k)$ is zero whenever the unit is to be turned on or off, more specifically when either $\gamma_j(k)$ or $\beta_j(k)$ is one. Thus, we suggest modifying the ramping constraints as shown in the following proposition.

Proposition 4.1. *The inequalities*

$$-RD_j\alpha_j(k) \leq r_j(k) \leq RU_j\alpha_j(k)$$
$$-RD_j\alpha_j(k+1) \leq r_j(k) \leq RU_j\alpha_j(k+1)$$
$$\forall j \in \mathcal{J}, \ \forall k \in \{0, ..., K-1\}, \tag{4.28}$$

with the dynamics in (4.27) *are correct.*

Proof. Recall from Table 4.1 that there 4 possible cases for the two successive binary indicators $\alpha_j(k)$ and $\alpha_j(k+1)$:

1. When $\alpha_j(k) = \alpha_j(k+1) = 1$. The unit stays on during the successive time slots, implying that $\beta_j(k)$ and $\gamma_j(k)$ are zero. Hence, the model in (4.27) will boil down to (4.21) and the ramping rate can take any value between $-RD_j$ and RU_j as in (4.22).

2. When $\alpha_j(k) = \alpha_j(k+1) = 0$. The unit stays off during the successive time slots, implying that $\beta_j(k)$ and $\gamma_j(k)$ are zero.

3. When $\alpha_j(k) = 0$ and $\alpha_j(k+1) = 1$. The unit is off and is to be turned on in the $(k+1)$th slot implying that $\beta_j(k) = 1$. From the first inequality in (4.28), $r_j(k) = 0$. Hence, the power level $p_j(k)$ changes by R_j^{SU} by the action of (4.27).

4. When $\alpha_j(k) = 1$ and $\alpha_j(k+1) = 0$. The unit is on and is to be turned off in the $k+1$th slot implying that $\gamma_j(k) = 1$. From the second inequality in (4.28), $r_j(k) = 0$. Hence, the power level $p_j(k)$ changes by $-R_j^{SD}$ by the action of (4.27).

$\square$

Remark 4.3. The model in (4.27) subject to the constraints in (4.28) is more demanding than that in (4.17) in the sense that the power level in the proposed model is restricted to change exactly by R_j^{SU} during start-up and R_j^{SD} during shut-down. In contrast, the model in (4.17) allows the ramping to change up to R_j^{SU} during start-up and R_j^{SD} during shut-down. However, this discrepancy between the two models should not affect the ramping process because the power generation is limited by $\underline{P}_j$ and $\overline{P}_j$ from (4.6) and the ramping rate limits during start-up and shut-down R_j^{SU} and R_j^{SD} are usually chosen to be equal to to the minimum allowed output power $\underline{P}_j$.

In the light of the above remark, the following result can be proved.

Proposition 4.2. *The ramping process model in* (4.27) *subject to the constraints in* (4.28) *is identical to that in* (4.17) *if* $\Delta_T R_j^{SU}$ *and* $\Delta_T R_j^{SD}$ *are set to* $\underline{P}_j$. *Further, if* $\Delta_T R_j^{SU}$ *or* $\Delta_T R_j^{SD}$ *are set to values less than* $\underline{P}_j$ *no feasible solution exists by both models.*

Proof. In Appendix B ☐

Finally, we use the same constraint in (4.18) to describe the spinning reserve.

4.4.2 Minimum Up and Down Times Constraints

One of the challenges in the MILP for the UC problem is the minimum up and down times constraints, as mentioned earlier. As explained in the first section, in MIP commonly used in literature, there are no states that store (accumulate) the on/off statuses over the time horizon. However, some authors on UC suggested a state to store the number of the on/off time slots. For example, the authors in [115], proposed a state $x_j(k)$ that stores the on/off statuses over the time horizon, such that $x_j(k) > 0$ when unit j has been left on up to time slot k, and $x_j(k) < 0$ when it has been left off. The following dynamics was proposed in [115] to capture this:

$$x_j(k+1) = \left\{ \begin{array}{ll} x_j(k) + \alpha_j(k), & \text{if } x_j(k)\alpha_j(k) > 0 \\ \alpha_j(k), & \text{if } x_j(k)\alpha_j(k) < 0 \end{array} \right\}. \qquad (4.29)$$

The above dynamics are not suitable for MILP because of the non-linear constraints, that is why the authors in [115] used LR to dissolve the coupling constraints in (4.7) and (4.18) and thus dividing the problem into sub-problems for each unit [115]. Similarly, the authors in [16] used almost the same model in (4.29) but they used DP to solve the UC problem. In addition, the authors in [90] proposed two states x^{on} and x^{off} to store the on/off statuses, but they used the PL technique to solve the problem. So, to our best knowledge, no one has used dynamic states to store the on/off statuses in an MIP for UC before.

To this end, we suggest two dynamic states $(x_j^{on}(k))$ and $(x_j^{off}(k))$ to count the time slots during which the units has been left on and off, respectively, $\forall k \in \{0, ..., K\}$. The following difference equation can be used for $x_j^{on}(k)$, assuming that feasible $x_j^{on}(0)$ is given:

$$x_j^{on}(k+1) = x_j^{on}(k) + \alpha_j(k+1) - r_j^{on}(k), \qquad (4.30)$$

which can be rewritten by using (4.26) as:

$$x_j^{on}(k+1) = x_j^{on}(k) + \alpha_j(k) + \beta_j(k) - \gamma_j(k) - r_j^{on}(k), \qquad (4.31)$$

where $r_j^{on}(k), \ \forall k \in \{0, ..., K-1\}$ is an input control used to drive the state $x_j^{on}(k+1)$ to zero when the unit is to be shut down, as will be shown in the sequel. Now, the state $x_j^{on}(k)$ is required to be zero when unit j is off. So, it must be bounded by the binary indicators $\alpha_j(k)$. Thus, an upper bound is required for $x_j^{on}(k)$ because unit j cannot be committed for ever, and let it be cK where c is an arbitrary positive integer. The following constraints ensure the precedent requirement:

$$0 \leq x_j^{on}(k) \leq cK\alpha_j(k), \quad \forall j \in \mathcal{J}, \ \forall k \in \mathcal{K}. \qquad (4.32)$$

In order to guarantee the action of the control input $r_j^{on}(k)$, the following constraint is suggested:

$$0 \le r_j^{on}(k) \le cK\gamma_j(k)$$
$$x_j^{on}(k) - 2K(1 - \gamma_j(k)) \le r_j^{on}(k) \le x_j^{on}(k)$$
$$\forall j \in \mathcal{J}, \ \forall k \in \{0, ..., K - 1\}. \tag{4.33}$$

Thus, when $\alpha_j(k + 1)$ is zero, i.e. $\gamma_j(k) = 1$ from (4.26), $x_j^{on}(k + 1)$ will be zero by (4.32). To allow this to happen, $r_j^{on}(k)$ must carry the value of $x_j^{on}(k)$, which is guaranteed from the second inequality in (4.33). In contrast, when $\gamma_j(k) = 0$, $r_j(k) = 0$ by the first inequality in (4.33). Now, the minimum up time constraint can be enforced by the following constraint:

$$UT_j\gamma_j(k) \le x_j^{on}(k), \quad \forall j \in \mathcal{J}, \ \forall k \in \{0, ..., K - 1\}, \tag{4.34}$$

in lieu of the constraints in (4.11). The last constraint in (4.34) ensures that the unit will not be shut down in the $(k + 1)$th slot (i.e. $\gamma_j(k) = 1$), unless $x_j^{on}(k)$ is at least equal to UT_j. To elaborate, $\gamma_j(k) = 0$ as long as $x_j^{on}(k) < UT_j$, which can be expressed as the following if statement:

$$[x_j^{on}(k) < UT_j] \longrightarrow [\gamma_j(k) = 0].$$

The contrapositive of the above statement can be stated as:

$$[\gamma_j(k) = 1] \longrightarrow [x_j^{on}(k) \ge UT_j],$$

which is exactly the inequality in (4.34) if $\gamma_j(k) = 1$.

Similarly, the dynamics of $x_j^{off}(k)$, assuming that feasible $x_j^{off}(0)$ is given, can be stated as:

$$x_j^{off}(k + 1) = x_j^{off}(k) + 1 - \alpha_j(k + 1) - r_j^{off}(k), \tag{4.35}$$

which can be rewritten by using (4.26) as:

$$x_j^{off}(k + 1) = x_j^{off}(k) + 1 - \alpha_j(k) - \beta_j(k) + \gamma_j(k) - r_j^{off}(k), \tag{4.36}$$

where $r_j^{off}(k)$ is an input control used to drive the state $x_j^{off}(k + 1)$ to zero when the unit is to be started up. With the same argument used to justify the constraints (4.32), (4.33), and (4.34), the following constraints are added:

$$0 \le x_j^{off}(k) \le cK(1 - \alpha_j(k)), \quad \forall j \in \mathcal{J}, \ \forall k \in \mathcal{K}, \tag{4.37}$$

$$0 \le r_j^{off}(k) \le \beta_j(k)$$
$$x_j^{off}(k) - cK(1 - \beta_j(k)) \le r_j^{off}(k) \le x_j^{off}(k)$$
$$\forall j \in \mathcal{J}, \ \forall k \in \{0, ..., K - 1\}, \tag{4.38}$$

and

$$DT_j\beta_j(k) \le x_j^{off}(k), \quad \forall j \in \mathcal{J}, \ \forall k \in \{0, ..., K - 1\}. \tag{4.39}$$

Remark 4.4. Just like all states in dynamic models, feasible initial values of the states $x_j^{on}(0)$ and $x_j^{off}(0)$ must be given. Feasibility here means that those initial values must be integers and satisfy (4.32) and (4.37).

Remark 4.5. Although the states $x_j^{on}(k)$ and $x_j^{off}(k)$ are required to hold integer values representing the number of the time slots for which unit j has been left on and off respectively, the model above does not require them to be $\in \mathbb{Z}$. Actually, as long as the initial values $x_j^{on}(0)$ and $x_j^{on}(0)$ are integers, all subsequent $x_j^{on}(k)$ and $x_j^{off}(k)$, $\forall k \in \mathcal{K}$ will be integers because the variables $\alpha_j(k)$, $\beta_j(k)$, and $\gamma_j(k)$ are binary. This also applies for $r_j^{on}(k)$ and $r_j^{off}(k)$.

Before presenting the numerical results, let us define some symbols for notational convenience. Let:

$$P_1^{\text{stat}} = \left\{ \begin{array}{c} (\mathbf{p}, \mathbf{r^s}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}) \in \mathbb{R}^{2JK} \times \mathbb{B}^{3JK} : (4.5) - (4.7),\ (4.11),\ (4.17), \\ \text{and } (4.18) \text{ are satisfied, and feasible } \alpha_j(0),\ p_j(0)\ \forall j \in \mathcal{J} \text{ are given} \end{array} \right\},$$
$$(4.40)$$

where

$$\mathbf{p} = [p_1(1), ..., p_1(K), ..., p_J(1), ..., p_J(K)]^T$$
$$\mathbf{r^s} = [r_1^s(1), ..., r_1^s(K), ..., r_J^s(1), ..., r_J^s(K)]^T$$
$$\boldsymbol{\alpha} = [\alpha_1(1), ..., \alpha_1(K), ..., \alpha_J(1), ..., \alpha_J(K)]^T$$
$$\boldsymbol{\beta} = [\beta_1(1), ..., \beta_1(K), ..., \beta_J(1), ..., \beta_J(K)]^T$$
$$\boldsymbol{\gamma} = [\gamma_1(1), ..., \gamma_1(K), ..., \gamma_J(1), ..., \gamma_J(K)]^T.$$

be the feasibility set of the commonly used static MIP described in the previous section. Let also,

$$P_1^{\text{dyn}} = \left\{ \begin{array}{c} (\mathbf{p}, \mathbf{r}, \mathbf{r^s}, \mathbf{x^{on}}, \mathbf{x^{off}}, \mathbf{r^{on}}, \mathbf{r^{off}}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}) \in \mathbb{R}^{7JK} \times \mathbb{B}^{3JK} : (4.6),\ (4.7),\ (4.18), \\ (4.26) - (4.28),\ (4.31) - (4.34),\ \text{and } (4.36) - (4.39) \text{ are satisfied,} \\ \text{and feasible } \alpha_j(0),\ p_j(0),\ x_j^{on}(0),\ x_j^{off}(0)\ \forall j \in \mathcal{J} \text{ are given} \end{array} \right\},$$
$$(4.41)$$

be the feasibility set of the MIP of the proposed dynamic model such that:

$$\mathbf{r} = [r_1(0), ..., r_1(K-1), ..., r_J(0), ..., r_J(K-1)]^T$$
$$\mathbf{x^{on}} = [x_1^{on}(1), ..., x_1^{on}(K), ..., x_J^{on}(1), ..., x_J^{on}(K)]^T$$
$$\mathbf{x^{off}} = [x_1^{off}(1), ..., x_1^{off}(K), ..., x_J^{off}(1), ..., x_J^{off}(K)]^T$$
$$\mathbf{r^{on}} = [r_1^{on}(0), ..., r_1^{on}(K-1), ..., r_J^{on}(0), ..., r_J^{on}(K-1)]^T$$
$$\mathbf{r^{off}} = [r_1^{off}(0), ..., r_1^{off}(K-1), ..., r_J^{off}(0), ..., r_J^{off}(K-1)]^T$$
$$\boldsymbol{\beta} = [\beta_1(0), ..., \beta_1(K-1), ..., \beta_J(0), ..., \beta_J(K-1)]^T$$
$$\boldsymbol{\gamma} = [\gamma_1(0), ..., \gamma_1(K-1), ..., \gamma_J(0), ..., \gamma_J(K-1)]^T.$$

Remark 4.6. Because of the difference equations in (4.26), (4.27), (4.31) and (4.36) the states $\alpha_j(k)$, $p_j(k)$, $x_j^{on}(k)$ and $x_j^{off}(k)$ can be substituted by:

$$\alpha_j(k) = \alpha_j(0) + \sum_{i=0}^{k-1} (\beta_j(i) - \gamma_j(i))$$

$$p_j(k) = p_j(0) + \sum_{i=0}^{k-1} \Delta_T \left(r_j(i) + R_j^{SU} \beta_j(i) - R_j^{SD} \gamma_j(i) \right)$$

$$x_j^{on}(k) = x_j^{on}(0) + \sum_{i=0}^{k-1} \alpha_j(0) + (k-i)(\beta_j(i) - \gamma_j(i)) - r_j^{on}(i)$$

$$x_j^{off}(k) = x_j^{off}(0) + \sum_{i=0}^{k-1} 1 - \alpha_j(0) - (k-i)(\beta_j(i) - \gamma_j(i)) - r_j^{off}(i), \qquad (4.42)$$

respectively. This stacking of variables may add to the complexity of the problem, especially when quadratic objective functions are used. However, for linear objective functions no difference of the computational time was noted.

Numerical Results

The example solved in [13], which is based on the case study in [49], is the core of our numerical experiments in this chapter. The example is about a ten-unit system, the specifications of the units used are listed in Table 4.2. The planning horizon was assumed one day ahead divided into 24 time slots each of one hour length. The demand on the network for a 24-hour horizon is listed in Table 4.3.

Unit Type	$\overline{P}$	$\underline{P}$	RD (RU)	DT (UT)	a $\times 10^{-4}$	b	c
	MW	MW	MWh	h	\$/MW^{2}h	\$/MWh	\$/h
I	455	150	225	8	4.8	16.9	1000
II	455	150	225	8	3.1	17.26	970
III	130	20	50	5	20	16.6	700
IV	130	20	50	5	21.1	16.5	680
V	162	25	60	6	39.8	19.7	450
VI	80	20	60	3	71.2	22.26	370
VII	85	25	60	3	7.9	27.74	480
VIII	55	10	10	1	41.3	25.92	660
IX	55	10	10	1	22.2	27.27	665
X	55	10	10	1	17.3	27.79	670

Table 4.2: The specifications of the generation units used in numerical solutions [13], [49]

In order to increase the computational complexity of the problem, the example above was repeated several times. Each time the units were replicated R times to

Time slot	1	2	3	4	5	6	7	8
Demand MW	700	750	850	950	1000	1100	1150	1200
Time slot	9	10	11	12	13	14	15	16
Demand MW	1300	1400	1450	1500	1400	1300	1200	1050
Time slot	17	18	19	20	21	22	23	24
Demand MW	1000	1100	1200	1400	1300	1100	900	800

Table 4.3: Total demand assumed [13], [49]

construct a harder example with $10 \times R$ units, and so was the demand. The initial values of the power generation were assumed to be:

$$
\begin{aligned}
[\alpha_1(0), ..., \alpha_{10}(0)] &= [1, 1, 0, ..., 0] \\
[p_1(0), ..., p_{10}(0)] &= [450, 220, 0, ..., 0],
\end{aligned} \tag{4.43}
$$

replicated R times for each example. The initial values of the on/off states for the proposed formulation P_1^{dyn} were assumed:

$$
\begin{aligned}
[x_1^{on}(0), ..., x_{10}^{on}(0)] &= [4, 5, 0, ..., 0] \\
[x_1^{off}(0), ..., x_{10}^{off}(0)] &= [0, 0, 1, 1, 1, 2, 2, 1, 1, 1],
\end{aligned} \tag{4.44}
$$

replicated R times for each example, as well. The arbitrary coefficient c in (4.32) and (4.37) was assumed to be 2. In order to ensure fair comparison, the periods for which unit j is required to be on or off at the beginning of the planning horizon denoted in (4.11) by U_j or D_j respectively, were assumed to be the complement to UT_j and DT_j given in Table 4.2, respectively. To elaborate, if the initial on-status of unit j is $x_j^{on}(0)$, then it is required to be on at the beginning of the planning horizon for at least $UT_j - x_j^{on}(0)$, whereas if $x_j^{on}(0) = 0$ there is no requirement for unit j to be on. Thus, the initial states U_j and D_j for formulation P_2^{stat} were assumed to be:

$$
\begin{aligned}
[U_1, ..., U_{10}] &= [4, 3, 0, 0, 0, 0, 0, 0, 0, 0] \\
[D_1, ..., D_{10}] &= [0, 0, 4, 4, 5, 1, 1, 0, 0, 0],
\end{aligned} \tag{4.45}
$$

replicated R times for each example. The minimization problem in each example was solved twice, one over the decision variables $(\mathbf{p}, \mathbf{r^s}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}) \in P_1^{\text{stat}}$ in (4.40), and another over the decision variables $(\mathbf{p}, \mathbf{r}, \mathbf{r^s}, \mathbf{x^{on}}, \mathbf{x^{off}}, \mathbf{r^{on}}, \mathbf{r^{off}}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}) \in P_1^{\text{dyn}}$ in (4.41). All problems were solved on a PC with 3.0 GHz Intel(R) Core(TIM)2 Duo CPU and 4.0 GB RAM, by the solver GUROBI 5.6 in MATLAB R2012b environment with YALMIP interface. The program was set to stop whenever the **relative MIP gap tolerance** falls below 0.01%, or when the total **CPU time** exceeds 500s when $R \leq 5$, and 750s when $R > 5$. We wanted to investigate the results of the proposed model with the PWL objective function and the quadratic objective function to study its behaviour as MILP and MIQP.

1. As MILP:

   The PWL approximation of the fuel consumption function in (4.13) was minimized for each example over P_1^{stat} and P_1^{dyn}. The PWL approximation $\widetilde{c_j^G}$

was made by two segments of the generation cost function $a_j p_j(k)^2 + b_j p_j(k)$ for all units, as:

$$\widetilde{c_j^G} = \left\{ \begin{array}{ll} m_j^1 p_j & 0 \le p_j \le P_j^1 \\ m_j^2 p_j + d_2 & P_j^1 \le p_j \le \overline{P}_j \end{array} \right\}, \tag{4.46}$$

where the parameters P_j^1, m_j^1 and m_j^2 are given in Table 4.4. The PWL approximation was implemented by the SOS2 technique explained in Subsection 3.2.2. The value of the best objective function, CPU time, and MIP gap at stopping are recorded in Table 4.5.

Unit Type	P^1	m^1	m^2
	MW	\$/MWh	\$/MWh
I	227.5	16.2992	16.5176
II	227.5	17.3305	17.4716
III	65.0	16.7300	16.9900
IV	65.0	16.6371	16.9115
V	81.0	20.0224	20.6671
VI	40.0	22.5448	23.1144
VII	42.5	27.7736	27.8407
VIII	27.5	26.0336	26.2607
IX	27.5	27.3310	27.4532
X	27.5	27.8376	27.9327

Table 4.4: Parameters of the PWL function $\widetilde{c_j^G}$, $\forall j \in \mathcal{J}$

It can be noted from Table 4.5 that the solutions obtained for the best objective functions are close to each other. For the cases when $R \le 6$ the proposed model in P_1^{dyn} could obtain solutions to within less MIP gap, while for the remaining cases the model P_1^{stat} found the solutions to less gaps. On the other hand, the optimal solutions found by the two models were the same for the cases when $R \le 3$, whereas the model in P_1^{stat} found less objective functions, especially when $R \ge 8$. Recall that the difference equations in (4.27) for the output power levels $p_j(k)$ introduce hyperplanes for the set P_1^{dyn}. Besides, each variable $p_j(k)$ is modelled as a convex combination of the breaking points used in the PWL approximation and the weights are found as a SOS2 set as explained in Subsection 3.2.2, which introduces another hyperplane for each variable. Hence, the proposed model in P_1^{dyn} may not be suitable for large-sized problems because searching for intersection points of excessive number of hyperplanes may add to the computational complexity.

2. As MIQP:
The quadratic fuel consumption function in (4.12) was minimized for each example twice, one over P_1^{stat} and another over P_1^{dyn}. The value of the best objective function, CPU time, and MIP gap at stopping are recorded in Table 4.6.

	Over P_1^{stat}			Over P_1^{dyn}		
R	Best Obj	Time[s]	Gap %	Best Obj	Time[s]	Gap %
	Time limit 500s					
1	581,104.65	42.75	0.01	581,104.65	108.30	0.01
2	1,159,436.69	500.00	0.20	1,159,373.04	500.00	0.17
3	1,736,226.08	500.00	0.24	1,736,226.08	500.00	0.16
4	2,315,907.93	500.00	0.32	2,315,862.01	500.00	0.27
5	2,892,456.80	500.00	0.24	2,892,581.82	500.00	0.22
	Time limit 750s					
6	3,470,157.85	750.00	0.22	3,470,418.27	750.00	0.21
7	4,049,702.37	750.00	0.26	4,050,851.49	750.00	0.27
8	4,627,343.57	750.00	0.26	4,632,221.50	750.00	0.32
9	5,206,168.87	750.00	0.28	5,214,680.74	750.00	0.39
10	5,782,638.73	750.00	0.24	5,787,884.57	750.00	0.29

Table 4.5: The results obtained by solving the UC problem over P_1^{stat} and P_1^{dyn} as MILP.

	Over P_1^{stat}			Over P_1^{dyn}		
R	Best Obj	Time[s]	Gap %	Best Obj	Time[s]	Gap%
	Time limit 500s					
1	580,922.23	164.60	0.01	580,922.23.04	181.00	0.01
2	1,159,009.22	500.00	0.27	1,159,062.06	500.00	0.28
3	1,735,708.18	500.00	0.20	1,735,708.18	500.00	0.16
4	2,314,776.11	500.00	0.27	2,314,863.11	500.00	0.18
5	2,892,324.84	500.00	0.25	2,894,245.95	500.00	0.44
	Time limit 750s					
6	3,469,790.82	750.00	0.22	3,471,260.54	750.00	0.19
7	4,049,061.29	750.00	0.25	4,051,798.70	750.00	0.24
8	4,628,950.82	750.00	0.28	4,627,926.02	750.00	0.17
9	5,208,061.50	750.00	0.31	5,206,690.30	750.00	0.27
10	5,790,929.91	750.00	0.35	5,790,790.78	750.00	0.74

Table 4.6: The results obtained by solving the UC problem over P_1^{stat} and P_1^{dyn} as MIQP.

It can be noted from Table 4.6 that the proposed model in P_1^{dyn} could find solution to within smaller MIP gaps than those found by the model in P_1^{stat} except for the cases when $R = 2$, 5 and 10. However, one can note that the solutions obtained by the proposed model in P_1^{dyn} are less than those obtained by the model in P_1^{stat} when $R = 8$, 9 and 10.

To wrap up, for UC problems of power systems comprising up to 100 units and covering 24 time slots, the proposed dynamic model in P_1^{dyn} gave almost as same results as those obtained by the commonly used static model in P_1^{stat}, it can be even better when the number of units is not so large, whether with the quadratic objective function or its PWL approximation. The difference between the results obtained by the two models, however, is not large.

4.4.3 Start-Up Cost

As mentioned earlier, the start-up cost is an exponential function of the form given in (4.14) and depicted in Fig. 4.1. We stated also that the commonly used way to capture the start-up cost in MIP is to discretize it as shown in Fig. 4.1, and model it as a decision variable with the constraints in (4.16). It was mentioned also that the discrete start-up steps K_j^t are, usually, assumed to take two values only: hc_j for $t \le DT_j + HT_j$, and cc_j for $DT_j + HT_j < t \le ND_j$ according to the authors in [13]. The reason for that may be to reduce the number of constraints which may affect the computational time with inconsiderable increase of accuracy.

Our proposal is to model the start-up cost as a linear function with saturation as:

$$
c_j^{SU}(k) = \left\{
\begin{array}{ll}
hc_j, & x_j^{off}(k-1) \le HT_j \\
m_j^{SU}(x_j^{off}(k-1) - HT_j) + hc_j, & HT_j \le x_j^{off}(k-1) \le CT_j \\
cc_j, & x_j^{off}(k-1) \ge HT_j
\end{array}
\right\},
$$
(4.47)

$\forall k \in \mathcal{K}$, as depicted in Fig. 4.2, with m_j^{SU} is the slope of the linear start-up cost function, HT_j is as before, and CT_j is the time period in slots after which unit j can be considered completely cold. Without loss of generality, CT_j is considered equal to ND_j, $\forall j \in \mathcal{J}$, from here on. Since the start-up cost is bounded from below by hc_j and from above by cc_j, and it is supposed to be zero unless $\beta_j(k) = 1$, the start-up cost is constrained by:

$$
hc_j \beta_j(k) \le c_j^{SU}(k) \le cc_j \beta_j(k) \ \forall k \in \{0, ..., K-1\}, \ \forall j \in \mathcal{J}.
$$
(4.48)

In the proposed model, the state $x_j^{off}(k)$ stores the off-statuses of unit j, but this state is not restricted by $\beta_j(k)$ as required for the start-up cost. However, the input control $r_j^{off}(k)$ is restricted by $\beta_j(k)$ and it carries the value of $x_j^{off}(k-1)$ as shown in (4.38). Hence, it can be used to tighten the start-up cost constraint as:

$$
c_j^{SU}(k) \ge \left[r_j^{off}(k) - HT_j \beta_j(k) \right] m_j^{SU} + hc_j \beta_j(k)
$$
$$
\forall k \in \{0, ..., K-1\}, \ \forall j \in \mathcal{J}.
$$
(4.49)

Note that, from the constraints in (4.48), the start-up cost in the kth slot depends on $r_j^{off}(k)$, which in turn, depends on the off-statuses counter x^{off} of the $(k-1)$th

slot, because in the kth slot $x^{off}(k) = 0$. Besides, the variable $r_j^{off}(k)$ is decreased by HT_j to model the lower bound shown in Fig. 4.2 by a horizontal shift. Further, when $\beta_j(k) = 0$, the lower bound of the start-up cost in (4.49) will be zero, as required. Nevertheless, the constraints (4.49) can not deal with saturation when $r_j^{off}(k)$ exceeds CT_j. So, the model may lead to start-up costs higher than cc_j. In order to solve this problem, a binary indicator is suggested: $\xi_j^{CT}(k)$, $\forall j \in \mathcal{J}$, $\forall k \in \{0, ..., K-1\}$ to indicate whether $r_j^{off}(k)$ exceeds CT_j or not. The logical action of such indicators can be described as follows:

$$[r_j^{off}(k) \leq CT_j] \longleftrightarrow [\xi_j^{CT}(k) = 0], \tag{4.50}$$

which can be translated into linear inequalities of the form:

$$
\begin{aligned}
r_j^{off}(k) - CT_j &\geq -CT_j + (CT_j + \epsilon)\xi_j^{CT}(k) \\
r_j^{off}(k) - CT_j &\leq (cK - CT_j)\xi_j^{CT}(k) \\
&\qquad \forall j \in \mathcal{J}, \quad \forall k \in \{0, ..., K-1\}.
\end{aligned}
\tag{4.51}
$$

Now, the start-up cost constraints in (4.48) and (4.49) can be combined and
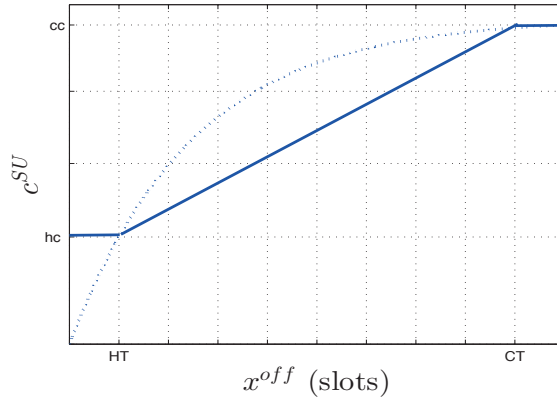
Figure 4.2: Exponential start-up cost (dashed) and linear (solid).

modified as shown in the following proposition.

Proposition 4.3. *The constraints*

$$
\begin{aligned}
c_j^{SU}(k) &\geq hc_j\beta_j(k) \\
c_j^{SU}(k) &\geq (\xi_j^{CT}(k) + \beta_j(k) - 1)cc_j \\
c_j^{SU}(k) &\geq \left[r_j^{off}(k) - HT_j\beta_j(k) - cK\xi_j^{CT}(k)\right]m_j^{SU} + hc_j\beta_j(k) \\
c_j^{SU}(k) &\leq cc_j\beta_j(k) \\
&\qquad \forall j \in \mathcal{J}, \quad \forall k \in \{0, ..., K-1\},
\end{aligned}
\tag{4.52}
$$

describe the start-up cost function in (4.47) correctly.

Proof. In Appendix B ∎

Remark 4.7. Actually, one can use a PWL approximation with more segments to model the start-up cost more accurately. However, this needs more binary variables and may add to the complexity of the problem for insignificant increase in the accuracy.

Before going to the numerical results, let us define some symbols for notational convenience, again. Let:

$$P_2^{\text{stat}} = \left\{ \begin{array}{c} (\mathbf{p}, \mathbf{r^s}, \mathbf{c^{SU}}, \; \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}) \in \mathbb{R}^{3JK} \times \mathbb{B}^{3JK} : (4.5) - (4.7), \; (4.11), \\ \text{and } (4.16) - (4.18) \text{ are satisfied}, \\ \text{and feasible } \alpha_j(0), \; p_j(0) \; \forall j \in \mathcal{J} \text{ are given} \end{array} \right\},$$

(4.53)

be the feasibility set of the commonly used static MIP described in the second section with start-up cost, where

$$\mathbf{c^{SU}} = [c_1^{SU}(1), ..., c_1^{SU}(K), ..., c_J^{SU}(1), ..., c_J^{SU}(K)]^T. \qquad (4.54)$$

Let also,

$$P_2^{\text{dyn}} = \left\{ \begin{array}{c} (\mathbf{p}, \mathbf{r}, \mathbf{r^s}, \mathbf{x^{on}}, \mathbf{x^{off}}, \mathbf{r^{on}}, \mathbf{r^{off}}, \mathbf{c^{SU}}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\xi^{CT}}) \in \mathbb{R}^{8JK} \times \mathbb{B}^{5JK} : \\ (4.6), \; (4.7), \; (4.18), \; (4.26) - (4.28), \; (4.31) - (4.34), \\ (4.36) - (4.39), \; (4.51), \; \text{and } (4.52) \text{ are satisfied, and feasible} \\ \alpha_j(0), \; p_j(0), \; x_j^{on}(0), \; x_j^{off}(0) \; \forall j \in \mathcal{J} \text{ are given} \end{array} \right\},$$

(4.55)

be the feasibility set of the MIP of the proposed dynamic model with the start-up cost such that:

$$\boldsymbol{\xi^{CT}} = [\xi_1^{CT}(0), ..., \xi_1^{CT}(K), ..., \xi_J^{CT}(1), ..., \xi_J^{CT}(K-1)]^T$$

Numerical results

The same 10-unit system from the study case in Subsection 4.4.2 was used here. The start-up cost parameters used for the model in P_2^{stat} of each generating unit are listed in Table 4.7. For the proposed model in P_2^{dyn}, CT_j was assumed equal to ND_j, $\forall j$ in Table 4.7. The slope of the linear start-up cost m_j^{SU} in (4.47) was calculated by $\frac{cc_j - hc_j}{CT_j - HT_j}$, $\forall j \in \mathcal{J}$.

As was done in Subsection 4.4.2, several examples were solved by replicating the system and the demands R times. For each example, the minimization problem was solved twice, one over the decision variables $(\mathbf{p}, \mathbf{r^s}, \mathbf{c^{SU}}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}) \in P_2^{\text{stat}}$ in (4.53), and another over the decision variables
$(\mathbf{p}, \mathbf{r}, \mathbf{r^s}, \mathbf{x^{on}}, \mathbf{x^{off}}, \mathbf{r^{on}}, \mathbf{r^{off}}, \mathbf{c^{SU}}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\xi^{CT}}) \in P_2^{\text{dyn}}$ in (4.55). The same initial values in (4.43), (4.45) and (4.44) were used here also for both models, as explained earlier. Moreover, each experiment was solved twice for both models, one with the PWL objective function as MILP, and another with the quadratic objective function as MIQP. As before, all minimization problems were solved by GUROBI 5.6 in MATLAB R2012b environment with YALMIP interface. The program was set to stop whenever the relative MIP gap tolerance falls below 0.01%, or when the

Unit Type	ND	HT[h]	$cc = 2hc$
	h	h	\$/h
I	17	5	9000
II	17	5	10000
III	13	4	1100
IV	13	4	1120
V	14	4	1800
VI	8	2	340
VII	8	2	520
VIII	3	0	60
IX	3	0	60
X	3	0	60

Table 4.7: The specifications of the start-up cost of the generation units used in numerical solutions [13], [49]

total CPU time exceeds 500s when $R \leq 5$, and 750s when $R > 5$. The results are shown in Table 4.9 and Table 4.10.

Before trying to comment on and compare the results shown in Table 4.9 and Table 4.10, some notes about the differences of the best objectives obtained by the two models must be mentioned. The differences of the best objective functions obtained are mainly due to the discrepancy of the start-up cost models. To elucidate, let us have a look at the output power levels over the planning horizon obtained from solving the UC problem by the two models for the case when $R = 1$, whether as MILP or as MIQP. The power trajectories over the planning horizon obtained by the two models are depicted in Fig. 4.3.

The start-up cost of each unit at each time slot obtained by the two models are listed in Table 4.8, together with the parameters ND_j and $HT_j + DT_j$ to simplify referring to them since they are needed for the discussion.

From Fig. 4.3 and Table 4.8, one can note that the first two units are not to be started up during the planning horizon because they had already been on as shown in the initial values in (4.43). On the other hand, Fig. 4.3 shows that the remaining units are to be started up during the planning horizon. The only difference in the start-up schedule is for the 7th unit, where it should be started-up in the 19th time slot by the model in P_2^{dyn} and in the 18th time slot by the model in P_2^{stat}. Let us take, e.g. the 3rd unit that has to be started up in the 5th time slot by the two models. From Table 4.8 we see that up to the end of the 4th time slot the unit has been left off for 5 time slots, i.e. $x_3^{off}(4) = 5$. This is correct because, from the initial values $x_3^{off}(0)$ in (4.44) for the model in P_2^{dyn} or the initial value D_3 in (4.45) for the model in P_2^{stat}, we know that the unit has been left off for one time slot before the beginning of the planning horizon, i.e. it has to remain off for at least 4 more time slots to satisfy the minimum down time which is equal to 5 slots for that unit. Hence, in the 4th slot of the planning horizon, unit 3 will have been left off for 5 time slots. According to the start-up model in (4.16) over P_2^{stat} and
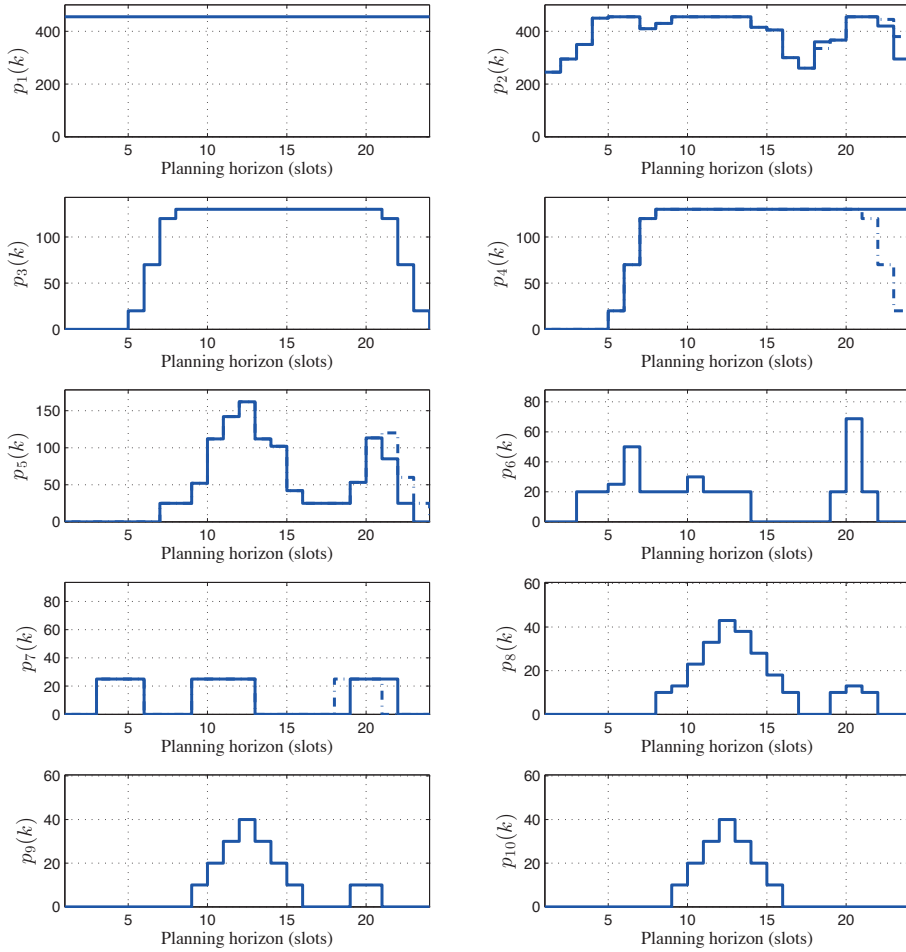
Figure 4.3: Output power levels over the planning horizon for all units for the case when $R = 1$: solid over P_2^{dyn}, and dashed over P_2^{stat}.

Unit	ND	HT	DT	P_2^{dyn}			P_2^{stat}	
				slot	x^{off}	c^{SU}	slot	c^{SU}
	h	h	h	(k)	(k-1)	(k)	(k)	(k)
3	13	4	5	5	5	611.1	5	550
4	13	4	5	5	5	622.2	5	560
5	14	4	6	7	7	1170	7	900
6	8	2	3	3	4	226.7	3	170
				19	5	255	19	170
7	8	2	3	3	4	346.7	3	260
				9	3	303.3	9	260
				19	6	433.3	18	260
8	3	0	1	8	8	60	8	60
				19	2	50	19	60
9	3	0	1	9	9	60	9	60
				19	3	60	19	60
10	3	0	1	9	9	60	9	60

Table 4.8: Comparison of start-up costs obtained by P_2^{stat} and P_2^{dyn} when solved as MIQP.

because two discrete steps only of that cost are considered, the minimum feasible start-up cost will be $hc_3 = 550$ because $k = 5 < DT_3 + HT_3 = 5 + 4 = 9$ from the parameters in Table 4.7. For the proposed model over P_2^{dyn}, the start-up cost is obtained from the constraints in (4.52) based on the linear function in (4.47), where $m_3^{SU} = \frac{1100-550}{13-4} = 61.1$, from the parameters in Table 4.7. Since $4 < k = 5 < 13$, the minimum start-up cost will be $hc_3 + m_3^{SU}(5-4) = 611.1$, which is identical to the value obtained by the solver and listed in Table 4.8.

Thus, we see that the start-up cost model in P_2^{dyn} is more realistic than that of the model in P_2^{stat}, unless more discrete steps are to be included in the model in P_2^{stat} which may have negative influence on the computational time. Now, we present the results of the numerical experiments by the two models as MILP and MIQP.

1. As MILP:

    The sum of the start-up cost and the function in (4.13) with the PWL approximation in (4.46) was minimized for each example over both formulations. The optimal value of the objective function, CPU time, and MIP gap at stopping are recorded in Table 4.9.

    Due to the discrepancy in the models used to capture the start-up cost between the two formulations, the values of the best objective obtained can not be used for the comparison. Instead, the MIP gap at stopping is the only base for this comparison. One can see from the results shown in Table 4.9 that the proposed model in P_2^{dyn} could reach optimal solutions to within MIP gaps that are comparable to those obtained by the model in P_2^{stat} for the cases when $R \leq 7$. It even gave solutions to within better gaps for the cases when $R = 3$, 6 and 7. However, for the cases when $R > 7$, the commonly

	Over P_2^{stat}			Over P_2^{dyn}		
R	Best Obj	Time[s]	Gap %	Best Obj	Time[s]	Gap %
	Time limit 500s					
1	584,817.96	104.10	0.01	585,409.35	97.60	0.01
2	1,166,985.35	500.00	0.20	1,167,990.35	500.00	0.29
3	1,747,698.80	500.00	0.23	1,749,367.21	500.00	0.19
4	2,330,234.21	500.00	0.25	2,333,020.97	500.00	0.26
5	2,911,322.17	500.00	0.20	2,914,909.49	500.00	0.23
	Time limit 750s					
6	3,492,972.68	750.00	0.21	3,496,626.61	750.00	0.19
7	4,075,951.40	750.00	0.25	4,080,673.58	750.00	0.21
8	4,658,349.21	750.00	0.26	4,670,933.30	750.00	0.39
9	5,240,207.77	750.00	0.26	5,261,723.44	750.00	0.52
10	5,822,205.16	750.00	0.27	5,844,657.83	750.00	0.53

Table 4.9: The results obtained by solving the UC problem over P_2^{stat} and P_2^{dyn} with the start-up cost, as MILP.

used model in P_2^{stat} found solutions to within much better gaps. This can be explained by the same argument used in comparing the results listed in Table 4.5. That is to say, the SOS2 model of the PWL approximation may not be suitable with the model in P_2^{dyn} for large-sized problems due to the increased number of hyperplanes and variables.

2. As MIQP:
   The sum of the start-up cost and the quadratic fuel consumption function in (4.12) was minimized for each example over both sets. The value of the best objective function, CPU time, and MIP gap at stopping are recorded in Table 4.10.
   As mentioned before, the optimal solutions obtained by the two models can not be compared due to the difference in the start-up cost models. Obviously, the proposed model in P_2^{dyn} could find optimal solutions to within smaller gaps than those obtained over P_2^{stat}, especially for large-sized problem when $R = 10$. Further, the MIP gaps at stopping recorded for the proposed model in P_2^{dyn} with the quadratic objective function listed in Table 4.10 is comparable to those recorded for those obtained by the model in P_2^{stat} in Table 4.9, which indicates that the PWL approximation does not necessarily reduce the complexity of the problem.

To sum up, in this section a state-space model of the power generation scheduling is presented with difference equations in discrete time steps, with a new technique to capture the start-up cost that could be more realistic than the commonly used techniques in literature. The numerical results showed that the proposed model is comparable to the commonly used techniques, whether with quadratic objective function as MIQP problem, or with its PWL approximation as MILP problem. In addition, the results showed that the proposed technique may not be appropriate

R	Over P_2^{stat}			Over P_2^{dyn}		
	Best Obj	Time[s]	Gap %	Best Obj	Time[s]	Gap %
	Time limit 500s					
1	584,620.03	180.90	0.01	585,227.48	137.40	0.01
2	1,166,472.77	500.00	0.25	1,167,623.74	500.00	0.21
3	1,747,071.88	500.00	0.21	1,748,850.21	500.00	0.19
4	2,329,559.31	500.00	0.26	2,333,030.78	500.00	0.24
5	2,910,482.03	500.00	0.21	2,915,167.50	500.00	0.21
	Time limit 750s					
6	3,492,224.70	750.00	0.21	3,497,611.06	750.00	0.20
7	4,077,003.79	750.00	0.29	4,082,858.53	750.00	0.26
8	4,657,134.38	750.00	0.22	4,665,605.12	750.00	0.25
9	5,247,380.30	750.00	0.37	5,252,206.61	750.00	0.37
10	5,846,814.12	750.00	0.65	5,832,997.39	750.00	0.30

Table 4.10: The results obtained by solving the UC problem over P_2^{stat} and P_2^{dyn} with the start-up cost, as MIQP.

for large-sized problems when SOS2 technique is used to represent the PWL approximation, but it can be considered better than the commonly used static models when quadratic objective function is used.

4.5 Valid Inequalities

In this section, we introduce some valid inequalities and investigate their influence on the proposed model and the commonly used one.

Proposition 4.4. *The inequalities*

$$\beta_j(k) + \alpha_j(k-1) \leq 1$$
$$\gamma_j(k) - \alpha_j(k-1) \leq 0$$
$$\forall k \in \mathcal{K}, \ \forall j \in \mathcal{J}, \qquad (4.56)$$

are valid for $conv(P_1^{stat})$ *and* $conv(P_2^{stat})$.

Proof. It is enough to note that if a unit is on during the $(k-1)$th slot, i.e. $\alpha_j(k) = 1$, it cannot be started up in the next slot because it is already on. This is guaranteed by the action of the constraint in (4.5) and by the fact that $\alpha_j(k)$ is a binary variable that cannot exceed unity.

For the second inequality, the same argument applies, but here $\gamma_j(k)$ cannot be one when $\alpha_j(k-1) = 0$ because it is already off. $\qquad\square$

Remark 4.8. The inequalities proposed in (4.56) are also valid for $conv(P_1^{\mathrm{dyn}})$ and $conv(P_2^{\mathrm{dyn}})$ with a minor change in the time index, specifically:

$$\beta_j(k) + \alpha_j(k) \leq 1$$
$$\gamma_j(k) - \alpha_j(k) \leq 0$$
$$\forall k \in \{0, ..., K-1\}, \ \forall j \in \mathcal{J}, \qquad (4.57)$$

due to the change in the definition of the control inputs $\beta_j(k)$ and $\gamma_j(k)$ in the proposed model. The validity can be proved with the same argument in the proof of the proposition above.

As a matter of fact, the constraints in (4.56) may not be facet-defining according to Definition 2.5. However, they represent a cutting plane for $conv(P_1^{\mathrm{stat}})$, and hence for $conv(P_2^{\mathrm{stat}})$. In order to see that, let us take the projection of $conv(P_1^{\mathrm{stat}})$ on the space of the variables $(\alpha_j(k-1), \beta_j(k), \gamma_j(k), \alpha_j(k))$, for some $j \in \mathcal{J}$. We can find some extreme points in this projection that can be cut out by the inequalities in (4.56). Consider the points:

$$(\alpha_j(k-1), \beta_j(k), \gamma_j(k), \alpha_j(k)) = \left\{ \begin{array}{c} (0,1,1,0) \\ (1,1,1,1) \end{array} \right\}. \qquad (4.58)$$

Obviously, the given points are in $conv(P_1^{\mathrm{stat}})$ because they satisfy the constraint in (4.5), and extreme, as well. However, they do not satisfy the inequalities in (4.56), hence the proposed inequalities provide cutting planes.

4.5.1 Numerical Results

The same 10-unit system from the study case in Subsection 4.4.2 and 4.4.3 was used here. As was done there, several examples were solved by replicating the system and the demands R times. The experiments presented previously are repeated here as well. The same initial values in (4.43), (4.45) and (4.44) were used here also, as explained earlier. As before, all minimization problems were solved by GUROBI 5.6 in MATLAB R2012b environment with YALMIP interface. The program was set to stop whenever the relative MIP gap tolerance falls below 0.01%, or when the total CPU time exceeds 500s when $R \leq 5$, and 750s when $R > 5$.

Without Start-Up Cost

In this subsection, we present the results we had from repeating the experiments by solving the minimization problem over P_1^{stat} and P_1^{dyn} without the start-up cost. As before, two objective functions were tried, the quadratic fuel consumption function and its PWL approximation.

1. As MILP.
   The proposed inequalities in (4.56) were added to the model in P_1^{stat}, whereas the inequalities in (4.57) were added to P_1^{dyn}. The PWL approximation of the fuel consumption function in (4.46) with the parameters in Table 4.4 was used here. The results are listed in Table 4.11, in which the results in Table 4.5 are listed as well, to simplify referring to them for comparing.

	R	Over P_1^{stat}			Over P_1^{dyn}		
		Best Obj	Time[s]	Gap %	Best Obj	Time[s]	Gap %
With proposed inequalities		Time limit 500s					
	1	581,104.65	43.30	0.01	581,104.65	77.30	0.01
	2	1,159,373.04	500.00	0.10	1,159,415.35	500.00	0.16
	3	1,736,226.08	500.00	0.11	1,736,226.08	500.00	0.16
	4	2,315,656.48	500.00	0.22	2,315,417.00	500.00	0.21
	5	2,892,590.76	500.00	0.17	2,893,643.50	500.00	0.26
		Time limit 750s					
	6	3,470,545.32	750.00	0.14	3,470,982.31	750.00	0.19
	7	4,050,124.94	750.00	0.20	4,050,526.87	750.00	0.26
	8	4,627,898.66	750.00	0.17	4,628,707.04	750.00	0.23
	9	5,206,197.92	750.00	0.20	5,207,938.33	750.00	0.30
	10	5,783,531.52	750.00	0.18	5,784,881.79	750.00	0.26
Without proposed inequalities		Time limit 500s					
	1	581,104.65	42.75	0.01	581,104.65	108.30	0.01
	2	1,159,436.69	500.00	0.20	1,159,373.04	500.00	0.17
	3	1,736,226.08	500.00	0.24	1,736,226.08	500.00	0.16
	4	2,315,907.93	500.00	0.32	2,315,862.01	500.00	0.27
	5	2,892,456.80	500.00	0.24	2,892,581.82	500.00	0.22
		Time limit 750s					
	6	3,470,157.85	750.00	0.22	3,470,418.27	750.00	0.21
	7	4,049,702.37	750.00	0.26	4,050,851.49	750.00	0.27
	8	4,627,343.57	750.00	0.26	4,632,221.50	750.00	0.32
	9	5,206,168.87	750.00	0.28	5,214,680.74	750.00	0.39
	10	5,782,638.73	750.00	0.24	5,787,884.57	750.00	0.29

Table 4.11: The results obtained by solving the UC problem over P_1^{stat} and P_1^{dyn} with the proposed inequalities, as MILP.

One can note from Table 4.11 that the effect of the proposed inequalities on the proposed model in P_1^{dyn} is not major, in general, because it could find solutions to within MIP gaps smaller than those without the proposed inequalities, but the differences of the gaps are not considerable. However, the proposed inequalities have a strong impact on the results obtained by the commonly used formulation P_1^{stat}, especially on the gaps at stopping that are definitely less than those obtained with the proposed inequalities.

2. As MIQP
The proposed inequalities in (4.56) were added to the model in P_1^{stat}, whereas the inequalities in (4.57) were added to P_1^{dyn}. The quadratic fuel consumption function in (4.12) with the parameters in Table 4.2 was used here. The results are listed in Table 4.12, in which the results in Table 4.6 are listed as well, to simplify referring to them for comparing.

		Over P_1^{stat}			Over P_1^{dyn}		
	R	Best Obj	Time[s]	Gap %	Best Obj	Time[s]	Gap%
With proposed inequalities		Time limit 500s					
	1	580,922.23	112.40	0.01	580,922.23	154.85	0.01
	2	1,159,009.22	500.00	0.16	1,159,009.22	500.00	0.26
	3	1,735,708.18	500.00	0.12	1,735,708.18	500.00	0.17
	4	2,314,744.64	500.00	0.21	2,314,746.30	500.00	0.18
	5	2,891,728.76	500.00	0.13	2,891,607.10	500.00	0.14
		Time limit 750s					
	6	3,469,425.00	750.00	0.13	3,469,576.94	750.00	0.13
	7	4,049,473.88	750.00	0.17	4,048,755.12	750.00	0.14
	8	4,627,155.50	750.00	0.17	4,630,847.18	750.00	0.44
	9	5,205,604.80	750.00	0.17	5,206,594.93	750.00	0.18
	10	5,785,161.30	750.00	0.21	5,785,713.65	750.00	0.19
Without proposed inequalities		Time limit 500s					
	1	580,922.23	164.60	0.01	580,922.23.04	181.00	0.01
	2	1,159,009.22	500.00	0.27	1,159,062.06	500.00	0.28
	3	1,735,708.18	500.00	0.20	1,735,708.18	500.00	0.16
	4	2,314,776.11	500.00	0.27	2,314,863.11	500.00	0.18
	5	2,892,324.84	500.00	0.25	2,894,245.95	500.00	0.44
		Time limit 750s					
	6	3,469,790.82	750.00	0.22	3,471,260.54	750.00	0.19
	7	4,049,061.29	750.00	0.25	4,051,798.70	750.00	0.24
	8	4,628,950.82	750.00	0.28	4,627,926.02	750.00	0.17
	9	5,208,061.50	750.00	0.31	5,206,690.30	750.00	0.27
	10	5,790,929.91	750.00	0.35	5,790,790.78	750.00	0.74

Table 4.12: The results obtained by solving the UC problem over P_1^{stat} and P_1^{dyn} with the proposed inequalities, as MIQP.

It can be noted from Table 4.12 that the effect of the proposed inequalities on the proposed model over P_1^{dyn} with the quadratic function is better than that with PWL approximation, in general. We can see the the MIP gaps at stopping are better, especially for the cases when $R > 4$, except for one case when $R = 8$. The solutions found by the model in P_1^{dyn} with the proposed inequalities are also better. Expectedly, the proposed inequalities have a strong impact on the results obtained by the commonly used model over P_1^{stat} as can be seen from the results on both the solutions found and the gaps at stopping. Furthermore, the results obtained by the commonly used model over P_1^{stat} with a quadratic objective function are better than those by the PWL approximation.

With Start-Up Cost

In this subsection, we present the results we had from repeating the experiments by solving the minimization problem over P_2^{stat} and P_2^{dyn} with the start-up cost. As

before, two objective functions were tried, the quadratic fuel consumption function and its PWL approximation.

1. As MILP

   The proposed inequalities in (4.56) were added to the model in P_2^{stat}, whereas the inequalities in (4.57) were added to P_2^{dyn}. The PWL approximation of the fuel consumption function with the start-up cost was minimized over both formulations. The results are listed in Table 4.13, in which the results in Table 4.9 are listed as well, to simplify referring to them for comparing.

	R	Over P_2^{stat}			Over P_2^{dyn}		
		Best Obj	Time[s]	Gap %	Best Obj	Time[s]	Gap %
		Time limit 500s					
With proposed inequalities	1	584,817.96	45.70	0.01	585,409.35	89.2060	0.01
	2	1,166,830.15	500.00	0.11	1,167,990.35	500.00	0.26
	3	1,747,588.87	500.00	0.12	1,749,367.21	500.00	0.19
	4	2,330,287.69	500.00	0.21	2,332,931.76	500.00	0.24
	5	2,912,023.85	500.00	0.18	2,915,201.62	500.00	0.23
		Time limit 750s					
	6	3,493,144.73	750.00	0.15	3,496,707.74	750.00	0.19
	7	4,076,625.99	750.00	0.20	4,083,322.50	750.00	0.30
	8	4,657,490.46	750.00	0.15	4,670,600.57	750.00	0.37
	9	5,239,948.02	750.00	0.17	5,261,206.35	750.00	0.51
	10	5,823,230.88	750.00	0.20	5,838,333.81	750.00	0.40
		Time limit 500s					
Without proposed inequalities	1	584,817.96	104.10	0.01	585,409.35	97.60	0.01
	2	1,166,985.35	500.00	0.20	1,167,990.35	500.00	0.29
	3	1,747,698.80	500.00	0.23	1,749,367.21	500.00	0.19
	4	2,330,234.21	500.00	0.25	2,333,020.97	500.00	0.26
	5	2,911,322.17	500.00	0.20	2,914,909.49	500.00	0.23
		Time limit 750s					
	6	3,492,972.68	750.00	0.21	3,496,626.61	750.00	0.19
	7	4,075,951.40	750.00	0.25	4,080,673.58	750.00	0.21
	8	4,658,349.21	750.00	0.26	4,670,933.30	750.00	0.39
	9	5,240,207.77	750.00	0.26	5,261,723.44	750.00	0.52
	10	5,822,205.16	750.00	0.27	5,844,657.83	750.00	0.53

Table 4.13: The results obtained by solving the UC problem over P_2^{stat} and P_2^{dyn} with the start-up cost and the proposed valid inequalities, as MILP.

No conclusion can be drawn about the influence of the proposed inequalities on P_2^{dyn} with the PWL approximation of the objective function. Actually, it is fuzzy because in some cases they could improve but not all of them. On the other hand, the proposed inequalities could indeed decrease the gaps at stop-

ping over P_2^{stat}, although the solutions obtained are in general greater than those obtained by the same formulation without the proposed inequalities.

2. As MIQP

The proposed inequalities in (4.56) were added to the model in P_2^{stat}, whereas the inequalities in (4.57) were added to P_2^{dyn}. The quadratic fuel consumption function was minimized with the start-up cost. The results are listed in Table 4.14, in which the results in Table 4.10 are listed as well, to simplify referring to them for comparing.

	R	Over P_2^{stat}			Over P_2^{dyn}		
		Best Obj	Time[s]	Gap %	Best Obj	Time[s]	Gap %
With proposed inequalities		Time limit 500s					
	1	584,620.03	78.03	0.01	585,227.48	171.35	0.01
	2	1,166,472.77	500.00	0.21	1,167,623.74	500.00	0.20
	3	1,747,083.27	500.00	0.17	1,748,850.21	500.00	0.18
	4	2,329,549.68	500.00	0.20	2,332,259.94	500.00	0.21
	5	2,910,484.90	500.00	0.16	2,914,428.19	500.00	0.19
		Time limit 750s					
	6	3,491,822.57	750.00	0.14	3,496,458.90	750.00	0.16
	7	4,075,086.82	750.00	0.18	4,080,452.99	750.00	0.21
	8	4,656,720.16	750.00	0.17	4,663,246.33	750.00	0.22
	9	5,251,625.22	750.00	0.41	5,245,865.68	750.00	0.19
	10	5,836,690.47	750.00	0.72	5,830,929.96	750.00	0.46
Without proposed inequalities		Time limit 500s					
	1	584,620.03	180.90	0.01	585,227.48	137.40	0.01
	2	1,166,472.77	500.00	0.25	1,167,623.74	500.00	0.21
	3	1,747,071.88	500.00	0.21	1,748,850.21	500.00	0.19
	4	2,329,559.31	500.00	0.26	2,333,030.78	500.00	0.24
	5	2,910,482.03	500.00	0.21	2,915,167.50	500.00	0.21
		Time limit 750s					
	6	3,492,224.70	750.00	0.21	3,497,611.06	750.00	0.20
	7	4,077,003.79	750.00	0.29	4,082,858.53	750.00	0.26
	8	4,657,134.38	750.00	0.22	4,665,605.12	750.00	0.25
	9	5,247,380.30	750.00	0.37	5,252,206.61	750.00	0.37
	10	5,846,814.12	750.00	0.65	5,832,997.39	750.00	0.30

Table 4.14: The results obtained by solving the UC problem over P_2^{stat} and P_2^{dyn} with the start-up cost and the proposed valid inequalities, as MIQP.

This time we can see from Table 4.14 that the proposed inequalities improved the solutions obtained by the proposed model P_2^{dyn} even if the MIP gaps are worse for some cases. For example, note that for the case $R = 10$, the proposed inequalities made P_2^{dyn} reach a better solution within the same time but with worse gap. For the commonly used model in P_2^{stat}, the proposed inequalities

have improved the solutions for all the cases except for the cases $R = 9$, and the gaps ate stopping are general better with the proposed inequalities.

To conclude, the proposed inequalities in (4.56) improved the commonly used models described by the sets P_1^{stat} and P_2^{stat} whether with a quadratic objective function or its PWL approximation. On the other hand, the proposed inequalities in (4.57) improved the proposed models described by the sets P_1^{dyn} and P_2^{dyn} when used with quadratic objective functions, while with the PWL approximation the effect is inconsiderable.

4.6 UC as Optimal Control Problem

The proposed dynamic model in state-space form makes the UC some kind of a OC problem. In general, OC problem is a control problem in which the dynamic system to be controlled is described by differential (difference) equations in continuous (discrete) time. Then, a cost function (performance index) that represents the cost on the control inputs and the states is minimized over some constraints. The most important of those constraints are the differential or difference equations themselves because they describe the only possible path the states can take. Thus, the target of the OC problem is to find the cheapest control inputs that drive the states trajectories to stability, i.e. an equilibrium or reference point within the control horizon from given feasible initial states, respecting the constraints on the states and the control inputs. For our case, the dynamic model proposed in the previous section comprises difference equations in state-space form, repeated here for convenience:

$$
\begin{aligned}
\alpha_j(k+1) =& \alpha_j(k) + \beta_j(k) - \gamma_j(k) \\
p_j(k+1) =& p_j(k) + \Delta_T r_j(k) \\
x_j^{on}(k+1) =& x_j^{on}(k) + \alpha_j(k) + \beta_j(k) - \gamma_j(k) - r_j^{on}(k) \\
x_j^{off}(k+1) =& x_j^{off}(k) + 1 - \alpha_j(k) - \beta_j(k) + \gamma_j(k) - r_j^{off}(k) \\
& \forall j \in \mathcal{J}, \forall k \in \mathcal{K},
\end{aligned}
\tag{4.59}
$$

where $\alpha_j(k)$, $p_j(k)$, $x_j^{on}(k)$ and $x_j^{off}(k)$ represent the states, and $\beta_j(k)$, $\gamma_j(k)$, $r_j(k)$, $r_j^{on}(k)$ and $r_j^{off}(k)$ are the control inputs, as explained before. The equilibrium point of the above model is trivially zero. So, this formulation changes the combined UC and ED problems into a mixed-integer tracking problem, in which the reference points that the control input should track are the minimizers of the cost function. Since the dynamics proposed is linear in nature, and the problem is a tracking problem, one does not have to worry about the stability of the system. The most important aspect of this problem is the solvability which is guaranteed if feasibility region is non-empty, as will be defined in details later.

Power systems are dynamic in nature. In practice, hierarchical control structure is used to control different parts of the system. The question that arises now is whether or not we can merge some control systems? For example, can one merge the AGC with the scheduling program? Of course, one would try to propose a model for the process before designing a controller. Unfortunately, we don not have

an answer for that. However, we believe that the proposed dynamic model could be exploited in the sought model. This opens more horizons for future research. For example, a more accurate ramping model of the generating unit can be used with the proposed model. Additionally, models of different types of loads can be suggested with the proposed model. Hence, more reliable and efficient scheduling or planning can be achieved. Moreover, models of energy prices and electricity spot markets can be incorporated with the proposed model. In the following subsection, we discuss how the proposed model can be more efficient if MPC philosophy is to be integrated.

4.6.1   UC and MPC

In Subsection 4.3.2 MPC exploitation in UC and ED problems was discussed. The reasons that make MPC philosophy so tempting when it comes to scheduling and planning were presented. We discussed the attempts of the authors in this field and the manufacturers in the industry to use the MPC approach. Then, we stipulated a dynamic model of the power generation systems to simplify the MPC utilization. Presumably, formulating the UC problem in dynamic sense, i.e. in state-space form, infers easier implementation of the MPC because the initial values will be updated more easily.

Since we have the dynamics that describes the power scheduling in (4.59) at hand, we can formulate the MPC problem as follows:

Problem 1. Given J generating units with their parameters, the states $\alpha_j(t)$, $p_j(t)$, $x_j^{on}(t)$, $x_j^{off}(t)$, and $\xi_j^{CT}(t)$ at any time t, the demand and spinning reserve expectation $D(k'|t)$ and $SR(k'|t)$ for the next K time slots, find a sequence of feasible control inputs $\beta_j(k')$, $\gamma_j(k')$, $r_j(k')$, $r_j^{on}(k')$, and $r_j^{off}(k')$ and decision variables $c_j^{SU}(k')$, $r_j^s(k')$, and $\xi_j^{CT}(k')$ $\forall k' \in \{t, ..., K + t - 1\}$, that minimize the cost function:

$$\sum_{k'=t+1}^{K+t} \sum_{j=1}^{J} c_j^F(p_j(k')) + c_j^{SU}(k') \tag{4.60}$$

subject to the constraints:

$$\alpha_j(k' + 1) = \alpha_j(k') + \beta_j(k') - \gamma_j(k')$$
$$p_j(k' + 1) = p_j(k') + \Delta_T r_j(k')$$
$$x_j^{on}(k' + 1) = x_j^{on}(k') + \alpha_j(k') + \beta_j(k') - \gamma_j(k') - r_j^{on}(k')$$
$$x_j^{off}(k' + 1) = x_j^{off}(k') + 1 - \alpha_j(k') - \beta_j(k') + \gamma_j(k') - r_j^{off}(k')$$

$$\sum_{j=1}^{J} p_j(k') = D(k'|t)$$

$$\sum_{j=1}^{J} r_j^s(k') \geq SR(k'|t)$$

$$\forall j \in \mathcal{J}, \ \forall k' \in \{t, ..., K + t - 1\}, \tag{4.61}$$

and the remaining constraints describing P_2^{dyn} in (4.55).

    With the model in Problem 1, the MPC algorithm works as follows. At each
time slot t the optimization problem is solved for the control inputs, and of course
we can consider the states as decision variables to resolve the stacking problem
explained earlier. Then, the control inputs are used to update the states for the
next time slot $t + 1$, and the optimization problem is solved again in this slot,
and so on. As stated before, one does not need to worry about the stability of
the dynamic model above, instead it is enough to ensure the non-emptiness of the
feasibility region and hence the solvability of Problem 1. In order to guarantee that
the feasibility region is not empty the following assumptions must be stated.

Assumption 1. At any time slot, the demand and spinning reserve are less than
the maximum capacity of the power system, videlicet:

$$D(k'|t) + SR(k'|t) \leq \sum_{j \in \mathcal{J}^*} \overline{P}_j.$$

    In the assumption above, $\mathcal{J}^* \subseteq \mathcal{J}$ denotes the set of indexes of all units ready to
be used, namely those units that are turned on or have been turned off for at least
DT_j. This assumption concerns the proper design of the power system. Reword,
power systems are usually designed to be reliable, this includes choosing generating
units that can provide some redundancy at each bus when required. The second
assumption, we need to make, concerns the ability of the power generation system
to adapt to the changes in the demand on the network.

Assumption 2. The increase in the demand over any two successive time slots
should be affordable by the system, i.e.

$$D(k' + 1|t) - D(k'|t) \leq \sum_{j \in \mathcal{J}} \Delta_T RU_j \alpha_j(k' + 1).$$

    To elucidate, under Assumption 2 the change of the total demand on the net-
work is guaranteed to be less than maximum allowed ramping rate of all committed
units in the next time slot.
It is noteworthy that the length of the time slot plays a vital role here. As ex-
plained earlier, the authors in scheduling literature stick to the static formulations
due to, among other reasons, long time slots which implies that the faster dynamics
of the generating units are insignificant. However, the computational time of the
optimization problem, especially for large-scale problems like those presented in
this work, is considerable, 500s (around 8min) and 750s (12.5min). So, the choice
of the best length of the time slot should be a trade-off between the accuracy of
the dynamics that describes the ramping of the generating units, and the compu-
tational time required to solve the problem. To this end, it suffices to show that the
proposed dynamic formulation could be more suitable for exploiting MPC philos-
ophy without showing any numerical results, because illustrative examples require
practical cases which certainly represent an interesting topic for future work.

4.7   Conclusion

The UC and ED problems were introduced and compared in this chapter. The
basic or commonly used MIP used to solve the UC and ED was presented for

comparing purposes. The main contributions of this chapter can be summarized in the following points:

1. A novel MIP was suggested to solve the combined UC and ED problems by using state-space form.

2. Based on that model a more realistic technique was proposed to capture the start-up cost as a PWL approximation.

3. Numerical experiments were carried out for case studies that involve up to 100 units over 24 time slots. The results show that the proposed model without start-up cost gives better results than the commonly used model for small-sized problems, especially with quadratic objective function. Besides, the proposed model with the proposed technique to capture the start-up cost enabled the solver to reach optimal solutions to within smaller gaps when quadratic objective function was used, while with PWL the proposed model behaved slightly worse than the commonly used models.

4. Strong inequalities were also suggested and proved to be good cuts. The numerical results presented showed that the effect of the proposed constraints on the commonly used models are stronger than that on the proposed model.

Finally, the state-space form used in the proposed model is believed to simplify the exploitation of the MPC philosophy in the planning and scheduling of power systems. In addition, it can be more adequate for integrating with dynamic models of power generation and different types of loads, which may be an interesting topic for future work.

Part II

Control of Gensets

Chapter 5

Diesel Generator Set

A Genset comprises a prime mover such as a Diesel Engine, and a synchronous generator. The most important controllers of such systems are the *speed governor* to regulate the engine or shaft speed and the *automatic voltage regulator* (AVR) to regulate the terminal voltage. The speed governor is a PID controller that uses the difference between the speed and its desired value as a feedback signal to change the fuel mass input by changing the fuel rack position. AVR is also a PID that uses the difference between the terminal voltage of the generator and its desired value, and changes it by manipulating the voltage of the field excitation circuit. Thus, the two controllers act separately. That is to say, if the speed varies from the desired value, the speed governor will react, while the AVR will not react as long as the voltage is stable, and vice versa. In this chapter, a control-oriented model is suggested for a Genset, and then a controller, that regulates the shaft speed and the terminal voltage, is designed by feedback linearisation. The proposed controller has two inputs: the fuel mass and the field circuit voltage. Simulations show that the proposed controller makes the two inputs act, simultaneously. Thus, any change of the speed e.g., forces the two input controls to react, in contrast to the ordinary PID controllers. Further, we discuss the robustness of the proposed controller to uncertainties and time delay. This chapter and the following one are based on the results published by the current author in [104].

5.1   Introduction

The most important control objectives in power systems stability studies are the voltage control, which leads to reactive power control, and frequency control, which leads to active power control. For these purposes, the AVR and speed governor are still the corner stone in the control hierarchy. The AVR is a PID controller that uses the error between the terminal voltage and its desired value as a feedback

signal to control the terminal voltage by controlling the magnetic field of the *rotor*, which can be produced by an excitation circuit or a permanent magnet, as will be explained later. The speed governor, on the other hand, regulates the torque provided by the prime mover, and thus the rotational speed, on which the frequency of the produced currents and voltages depends. The manipulated control inputs of both controllers depend on the type of the machine and the prime mover used.

For large power systems that contain many generating units this looks sufficient, because the complicated hierarchical control structure of power systems may not be so vulnerable to small variations in voltage or frequency. However, for isolated power systems this may not be the case. The authors in [70] showed that the speed governor and the AVR act separately, which may create problems. Then, they gave the following example to show their point [70]. If the load increases suddenly, the speed of the engine and the terminal voltage will drop. Hence, both controllers will react by increasing the fuel input to the engine, and the field circuit excitation. However, this may cause the terminal voltage to increase above the steady-state value, and hence increasing the load on the engine, which in turn drives the governor to increase the fuel, and stability may be lost or retarded. That is why, the researchers in this field have been trying to design a controller that drives the two control inputs; the field excitation and the fuel input, simultaneously, since it is believed that if the two input controls are coordinated, the performance of the controller under peculiar situations such as the one explained in [70] will be improved.

The authors in [84] proposed a non-linear model of the isolated permanent magnet synchronous generator driven by a Diesel engine. Then, the non-linear model was linearised around steady-state values. Further, a performance index was suggested to find an optimal controller to regulate the terminal voltage and frequency, where they used two control inputs; the fuel mass to control the torque of the engine, and the firing angle of a thyristor to control the terminal voltage [84]. Although the authors in [84] used a first order model of the Diesel engine, the simulations presented showed the effectiveness of the proposed controller. Then, the authors in [31] proposed a sliding-mode speed controller of the Genset. Later, in [71], the authors proposed a fuzzy logic speed controller. Recently, the phenomenon of power oscillations of marine power systems, that have several Gensets working in parallel, was studied and analysed in [43]. Moreover, the author proposed a non-linear model of the Genset and designed a robust synthetic controller to control this phenomenon [43]. In spite of the novelty of his proposed controller, the author in [43] did not propose a model for the engine torque, instead he used the actuator dynamics. Further, the model proposed for the dynamics of the terminal voltage may not be adequate for such purposes.

The current author proposed a model for the Genset that has a propeller in addition to the engine and the machine in [103]. However, in [103] we used the flux linkage as a state variable which is usually difficult to be measured. Besides, we used the first-order model of the Diesel engine in [25] that uses the fuel rack position input to manipulate the torque, without taking in consideration the air dynamics. In this chapter, the terminal voltage is used instead of the flux linkage because it is easier to be measured. Further, the model of the torque developed by the Diesel engine proposed in [45] is used here because it describes the torque more accurately than

the first-order model in [25], and it takes the air dynamics in consideration, as will be shown later. Then, a controller is designed by using feedback linearisation. The proposed controller has two manipulated control inputs; the fuel mass input to the engine, and the field excitation circuit voltage. The air/fuel ratio in the Diesel engine is modelled as uncertain parameter. The simulations presented, show that the two control inputs act simultaneously, i.e. they communicate with each other, as will be explained in the discussion of the simulation results. In addition, the simulations show that, the controller performs satisfactorily for small values of time delay imposed by the Diesel engine, when it is considered. Moreover, our simulations show that the proposed controller can be considered robust to the uncertainty of the air/fuel ratio.

5.2 Mathematical Model

In this section, we present the proposed model of the Genset and its controller.

5.2.1 Synchronous Machine

Synchronous machines are essential components of power systems. They can be used as generators or as motors. The basic principle of operation of the synchronous machine as a generator can be explained as follows. A synchronous machine comprises a rotating part called *rotor*, and a stationary part called *stator*. The stator carries the armature windings which are 3-phase ac windings separated by 120°. On the other hand, the rotor provides a direct magnetic field by either a permanent magnet or a dc field winding around the rotor powered by an exciter which may take several forms. When the rotor is rotated by a prime mover such as a Diesel engine in our case, an *electromotive force* (emf) will be induced in the armature windings. Because of the windings structure, an ac current will be induced in the stator, and hence an ac magnetic field will be created from each phase winding in the armature. Thus, the resultant magnetic field of the armature windings will be rotating, i.e. sinusoidal function of time. The rotating magnetic field from the stator tries to catch the rotating magnetic field produced from the rotor, and hence a torque is created and transformed into electrical energy in the form of the ac produced in the stator. The steady state speed with which the magnetic field produced in the armature rotates is called the *synchronous speed*. It can be shown that this synchronous speed is related to the mechanical speed of the rotor through:

$$\Omega_M = \frac{2}{p}\Omega_S, \tag{5.1}$$

where p is the number of the magnetic poles in the rotor, Ω_M is the mechanical rotational speed measured in mechanical `rad/s`, and Ω_S is the electrical rotational speed measured in electrical `rad/s`.

In power systems analysis, it is more convenient to use *per unit* (`p.u.`) notation to express the quantities and variables. Basically, a quantity in `p.u.` is a normalized quantity with respect to an appropriate base value, that is to say [56]:

$$\text{Quantity in p.u.} = \frac{\text{Quantity}}{\text{Base value of the quantity}}. \tag{5.2}$$

In order to appreciate the benefit of the p.u. system, it is enough to note that ω_M and ω_S in (5.1) will be equal in p.u. To elaborate, let the base mechanical rotational speed and the base electrical rotational speed be their rated values, denoted from here on by the superscript (^{r}), i.e. Ω_M^r, and Ω_S^r. Then, by using (5.1), one obtains:

$$\frac{\Omega_M}{\Omega_M^r} = \frac{\frac{2}{p}\Omega_S}{\frac{2}{p}\Omega_S^r} = \frac{\Omega_S}{\Omega_S^r} = \omega, \tag{5.3}$$

where ω is the synchronous speed in p.u.

Another important common simplification in power systems analysis is the use of the dq−frame. Since the magnetic field produced by the armature winding rotates with the same rotational speed of the rotor in steady state, this magnetic field appears stationary from the rotor side. Hence, it can be resolved on two perpendicular axes, that are called the *direct* (d) axis along the rotor, and *quadrature* (q) axis perpendicular to the rotor, as shown in Fig. 5.1. The main advantage of using the dq-transformation is to cancel the variations of the mutual and self inductances of the coils due to their dependence on the angle of the rotation. For more details on the dq-transformation, the reader is referred to any reference on basic power systems analysis or machines, such as [56] and [64].
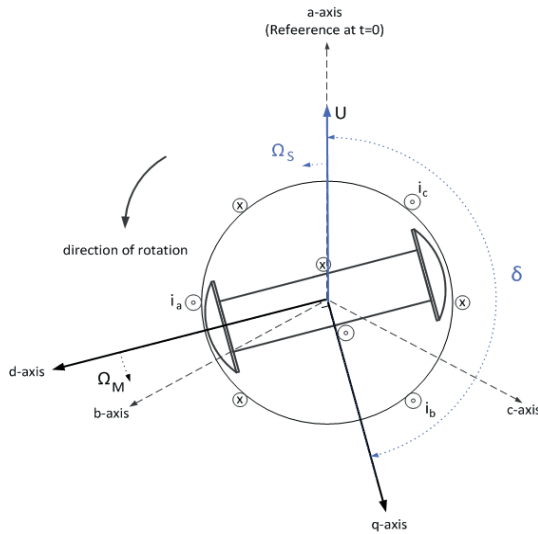
Figure 5.1: *dq*-frame in the synchronous machine and the angles involved.

Now, the synchronous generator model, in p.u., and in *dq*-frame, can be described by relations of flux linkages as follows [56], [64]:

$$\psi_d = -X_d i_d + X_{Fd} i_F \tag{5.4a}$$
$$\psi_q = -X_q i_q \tag{5.4b}$$
$$\psi_F = X_F i_F - X_{Fd} i_d, \tag{5.4c}$$

and the voltages as follows, [56], [64]:

$$u_d = \dot{\psi}_d - \omega\psi_q - R_a i_d \tag{5.5a}$$

$$u_q = \dot{\psi}_q + \omega\psi_d - R_a i_q \tag{5.5b}$$

$$u_F = \dot{\psi}_F + R_F i_F, \tag{5.5c}$$

where ψ_d, ψ_q, u_d, u_q, i_d, and i_q are the $d-$, and $q-$axis components of the stator flux linkages, terminal voltage, and stator current, respectively. Further, ψ_F, u_F and i_F are the field circuit flux linkage, voltage, and current respectively. X_d, X_q, an R_a are the $d-$, and $q-$axis components of the stator self inductance, and the armature resistance, respectively. It is worth mentioning here that R_a is usually much less than the inductances X_d and X_q. Moreover, X_F, and R_F are the field circuit self inductance, and resistance, respectively. X_{Fd} is the mutual inductance between the field circuit and stator windings. Recall that all quantities are in p.u. In the study of power system stability, it is usually acceptable to assume the following:

1. The dynamics of the stator flux linkages $\dot{\psi}_d$ and $\dot{\psi}_q$ are negleible.

2. ω in (5.5a) and (5.5b) is equal to unity.

For the motivation of the above assumptions the reader is referred to [56], [64] and the references therein. Hence, the equations in (5.5) can be rewritten as [56], [64]:

$$u_d = -\psi_q - R_a i_d \tag{5.6a}$$

$$u_q = \psi_d - R_a i_q \tag{5.6b}$$

$$u_F = \dot{\psi}_F + R_F i_F. \tag{5.6c}$$

Because the flux linkages are hard to measure in practice, it is also common in the literature on power system stability to use the following transformations, [56] and [64]:

$$E_I = X_{Fd} i_F$$

$$E_q' = \frac{X_{Fd}}{X_F} \psi_F$$

$$E_F = \frac{X_{Fd}}{R_F} u_F, \tag{5.7}$$

with which the equations in (5.4) can be rewritten as [56], [64]:

$$\psi_d = -X_d i_d + E_I \tag{5.8a}$$

$$\psi_q = -X_q i_q \tag{5.8b}$$

$$E_q' = E_I - (X_d - X_d') i_d, \tag{5.8c}$$

where $X_d' = X_d - \frac{X_{FD}^2}{X_F}$, and (5.6c) can be rewritten as:

$$\dot{E}_q' = \frac{1}{T_{d0}'} (E_F - E_I), \tag{5.9}$$

89

where T'_{d0} is a time constant in seconds. Notice that the only differential equation in this model is the one given in (5.9), which will be used later in the proposed state-space model. Eq. (5.9) describes the dynamics of the voltage E'_q, which could be easier to measure than the flux linkage ψ_F in (5.6c). Nevertheless, we would like to replace this with a differential equation that describes the dynamics of the terminal voltage. Thus, inserting (5.6b) and (5.8a) in (5.8c), one can easily obtain the following relation:

$$E'_q = u_q + X'_d i_d + R_a i_q. \tag{5.10}$$

In order to describe the dynamics of the terminal voltage, let us differentiate the above equation, to get:

$$\dot{E}'_q = \dot{u}_q + X'_d \frac{di_d}{dt} + R_a \frac{di_q}{dt}. \tag{5.11}$$

Differentiating (5.6a) and (5.6b) with respect to time and after neglecting the derivatives of the flux linkages as was done before, the derivatives of the stator currents in terms of the voltages can be described by:

$$\frac{di_d}{dt} = -\frac{\dot{u}_d}{R_a}$$
$$\frac{di_q}{dt} = -\frac{\dot{u}_q}{R_a}. \tag{5.12}$$

Remark 5.1. The rate of change of the terminal voltage is much less than the rate of change of the stator currents. This can be seen from (5.12) because R_a is usually small.

However, the voltages in $dq-$frame are expressed, by definition as shown in Fig. 5.1, as:

$$u_d = U \sin(\delta)$$
$$u_q = U \cos(\delta), \tag{5.13}$$

where U is the terminal voltage in p.u., and δ is the angular position of the rotor with respect to the rotating reference, as shown in Fig. 5.1, in electrical rad. Hence, the derivatives of the voltages in $dq-$frame are given by:

$$\dot{u}_d = \dot{U} \sin(\delta) + \dot{\delta} U \cos(\delta)$$
$$\dot{u}_q = \dot{U} \cos(\delta) - \dot{\delta} U \sin(\delta). \tag{5.14}$$

Finally, inserting (5.9), (5.12), and (5.14) in (5.11), and by using (5.8c) and (5.10), after some algebraic simplifications and by neglecting the term with R_a^2 we obtain:

$$\dot{U} = (\frac{R_a}{T'_{d0} X'_d} - \dot{\delta}) U \cot(\delta) + \frac{R_a X_d}{T'_{d0} X'_d \sin(\delta)} i_d - \frac{R_a}{T'_{d0} X'_d \sin(\delta)} E_F. \tag{5.15}$$

The rotor is supposed to rotate with the synchronous speed, and thus the angle δ is supposed to be constant in steady state. Nevertheless, in the general form, the angle δ will change as [56], [64]:

$$\delta = \Omega_S t - \Omega_S^r t + \delta_0, \tag{5.16}$$

where δ_0 is the initial value. Then, the dynamics of the rotor angle δ can be obtained, by differentiating the above equation, and by using (5.3), to be:

$$\dot{\delta} = \Omega_S^r(\omega - 1). \tag{5.17}$$

Finally, we need to calculate the electromagnetic torque of the synchronous machine. As mentioned earlier, the rotating magnetic field from the armature windings tries to catch the magnetic field of the rotor, and thus a torque is created. If the machine is working as a motor, the current will be fed into the stator, the torque developed will be mechanical and thus the rotor keeps moving. If the machine is working as a generator, the rotor is moved by an external prime mover, and an electromagnetic torque is produced to induce currents in the armature windings. The electromagnetic torque q_S, in p.u., can be expressed by:

$$q_S = \psi_d i_q - \psi_q i_d, \tag{5.18}$$

which can be simplified by using (5.4) and (5.6), and by neglecting R_a, to:

$$\begin{aligned} q_S =& u_q \frac{u_d}{X_q} + u_d i_d = \frac{1}{X_q} U^2 \sin(\delta)\cos(\delta) + U i_d \sin(\delta) \\ =& \frac{1}{2X_q} U^2 \sin(2\delta) + U i_d \sin(\delta). \end{aligned} \tag{5.19}$$

5.2.2 Diesel Engine

Diesel engines belong to the class of the *Compression Ignition* (CI) engines, that do not need a spark to start the ignition, of *Internal Combustion* (IC) engines. Diesel engines are the most efficient IC engines, they can achieve over 50% efficiency [35]. Diesel engines suffer from two main problems: low power density and nitric oxygen (NO_x) emission [36]. The problem with power density is solved by using a *turbocharger*, while the problem with the NO_x emission is overcome by using *Exhaust Gas Recirculation* (EGR). Basically, a turbocharger is a compressor that pushes more air into the cylinder and thus more fuel is burnt in the same volume. The EGR, on the other hand, is a technique that depends on valves that allow the exhaust gas to be fed back into the cylinder, and thus reducing the NO_x emissions. To get more details on how the EGR does that, the reader is referred to [41], [35], and [36] and the references therein.

Actually, discussing the control of the Diesel engine in details is beyond the scope of this work. However, we need to emphasize that the modern Diesel engine is a complicated system by itself that contains several interconnected control loops, mainly the air path, fuel path, and exhaust gas path, that need to be coordinated. For example, increasing the EGR valve opening reduces the NO_x, but at the same time it increases the *Brake Specific Fuel Consumption* (BSFC) and the particulates emission. Hence, the best performance of the Diesel engine is a trade-off among many factors. That is why, it is very important when designing a controller to choose the target taking in consideration the other control loops.

In order to design a controller that regulates the shaft speed and the terminal voltage in a Genset, one is interested in, mainly, the torque provided by the Diesel

engine. Many models have been suggested in the literature for the torque produced by the Diesel engine depending on the objective of the proposed model. For the control of a Genset, it is enough to develop a simplified model based on the speed governor. The author in [25] stated three models described by transfer functions between the output, that is the torque produced by the Diesel engine Q_E in N m, and the input that is y the fuel pump index. The most common of them is the one which we re-write here in time domain [25]:

$$\dot{Q}_E = \frac{-1}{T_y} Q_E + \frac{K_y}{T_y} y(t - \tau), \qquad (5.20)$$

where T_y is a time constant, K_y is a gain constant, and τ is a time delay or *dead time*, in seconds, of the Diesel engine given by [25]:

$$\tau = \frac{1}{2 N_c n_E}, \qquad (5.21)$$

where N_c is the number of the cylinders in the Diesel engine, and n_E is its speed in revolutions per second (rps). By using this model, we can consider the engine torque as another state in the proposed model. This stipulates that the torque is available for measuring. In fact, torque is not easy to measure in practice, hence observers are required to overcome this problem. Alternatively, one can simplify the model above by neglecting the engine torque dynamics $\dot{Q}_E$. This simplification is very common in literature and practice, see e.g. [39], and it implies that the torque is considered equal to the input control multiplied by its gain $K_y y(t - \tau)$. Anyway, by using the speed governor, the amount of fuel is adjusted by controlling the rack position based on the error signal of the speed, which can be measured easily. Now, if we would like to combine the action of the speed governor and the AVR, a state-space model is needed because of the non-linearity of the system. Although, the model in (5.20) was used in [40] to model and control the Genset, we believe that this model may not be sufficient for the following two reasons. First, the air dynamics are not taken in consideration. Then, the engine torque is difficult to measure.

Many researchers have been trying to propose a model to determine the torque in order to avoid measuring it. The model we adopt in this work is the one suggested in [45]. The torque produced by the Diesel engine Q_E is the difference between the indicated torque Q_{ind} and the frictional torque Q_{Fr} [41]. The indicated torque can be described by the following well-known relation [47], [35]:

$$Q_{\mathrm{ind}} = m_\phi H_l \eta_{\mathrm{ind}}, \qquad (5.22)$$

where H_l is the fuel lower heating value in J/kg which is constant for Diesel fuel and it will be considered in all simulations equal to 42 MJ/kg. Furthermore, m_ϕ is the fuel mass injected in kg, and η_{ind} denotes the indicated efficiency. The authors in [45] proposed a *mean-value* model of the different parts and loops of a small turbocharged Diesel engine used in vehicular systems. They analysed experimental data to show that the indicated efficiency is a function of the engine speed and the fuel/air equivalence ratio Φ, as [45]:

$$\eta_{\mathrm{ind}} = (a_1 + a_2 \Omega_E + a_3 \Omega_E^2)(1 - a_4 \Phi^{a_5}), \qquad (5.23)$$

where Ω_E is the rotational speed of the engine in mechanical `rad/s`, and $a_1, \ldots, a_5$ are parameters of appropriate dimensions. According to the definitions in [41], the fuel/air equivalence ratio is the reciprocal of the *relative* air/fuel ratio λ defined by [41]:

$$\Phi^{-1} = \lambda = \frac{A/F}{(A/F)_s}, \tag{5.24}$$

where the subscript $(_s)$ denotes the value corresponding to the *stoichiometric* combustion (see Subsection 5.4.2), and A/F is the air/fuel ratio given by [41]:

$$A/F = \frac{\dot{m}_a}{\dot{m}_f}, \tag{5.25}$$

with $\dot{m}_a$ and $\dot{m}_f$ represent the flow rate of the air and the fuel, respectively. On the other hand, the frictional torque Q_{Fr} is described by [47], and [41]:

$$Q_{\mathrm{Fr}} = \frac{p_{\mathrm{fme}} V_d}{2\pi\nu}, \tag{5.26}$$

where V_d denotes the displacement volume in $\mathtt{m^3}$, ν is the number of revolutions for each power stroke per cycle, and p_{fme} is the friction mean effective pressure in `Pa`. The author in [41] deduced from motoring tests of different types of Diesel engines that friction mean effective pressure p_{fme} can be described by:

$$p_{\mathrm{fme}} = 1000(C_1 + 0.048\frac{60}{2\pi}\Omega_E + 0.4\overline{S}_p^2), \tag{5.27}$$

where C_1 is a constant in `kPa` that depends on the engine type, $\overline{S}_p$ is the mean piston speed in `m/s`. The last thing we need to take in consideration is the time delay given in (5.21), because the indicated torque Q_{ind} requires some time to have effect. Thus, the final model of the torque produced by the engine is given by:

$$\begin{aligned} Q_E(t) =& Q_{\mathrm{ind}}(t-\tau) - Q_{\mathrm{Fr}}(t) \\ =& H_l(a_1 + a_2\Omega_{E_\tau} + a_3\Omega_{E_\tau}^2)(1 - a_4\Phi_\tau^{a_5})m_{\phi_\tau} \\ & - \frac{1000 V_d}{2\pi\nu}(C_1 + 0.048\frac{60}{2\pi}\Omega_E + 0.4\overline{S}_p^2), \end{aligned} \tag{5.28}$$

where the subscript $(_\tau)$ denotes the delayed signal, e.g. $\Omega_{E_\tau} = \Omega_E(t-\tau)$.
Actually, the authors in [45] claimed that the model above can be used for large Diesel engines, as well. However, analysing experimental data to verify this claim is beyond the scope of this work, so we depend on the previous claim to design the proposed controller.

5.2.3 Shaft Dynamics

One of the most fundamental relations in the study of power system stability is the so-called *swing equation*. This equation connects the mechanical and electrical aspects of the generators. The synchronous machine and the prime mover are

connected mechanically through a shaft and sometimes a gear box. Thus, the rotational speed of the engine is as same as that of the shaft, while the speed of the synchronous machine is related to that of the engine by:

$$\Omega_M = R_M \Omega_E, \tag{5.29}$$

where R_M is the gear ratio, and Ω_M is the mechanical rotational speed of the synchronous machine in mechanical $\mathtt{rad/s}$, as defined in (5.1). Then, applying Newton's second law of rotating objects to the mechanical system, one gets:

$$(I_E + R_M^2 I_M)\dot{\Omega}_E = Q_E - R_M Q_S - R_M Q_D, \tag{5.30}$$

where Q_E is as before, Q_S denote the electromagnetic torque consumed by the synchronous machine in $\mathtt{N\,m}$, and Q_D is a damping torque. Further, I_E and I_M are the moment of inertia, in $\mathtt{kg\,m^2}$, of the engine and synchronous machine, respectively.

Remark 5.2. Note that, in the model in (5.30), the moment of inertia of the shaft is neglected, this should not be a problem since this quantity is constant and it can be added to the moment of inertia of the engine or the machine. Further, the shaft introduces some kind of friction to the motion. One can find many models to include the opposing torque resulting from this friction, see e.g. [82] and the references therein. The simplest one may be to assume the frictional torque proportional to the rotational speed. In this work, the frictional torque is neglected because including it should not affect the proposed strategy to design the controller, as long as the frictional torque function is "nice", and hence can be added to the load torque.

Following the procedure in [56], define the $\mathtt{p.u.}$ inertia constant H_T as the ratio of the stored kinetic energy at rated speed to the base value of the apparent power of the synchronous machine S_base in $\mathtt{VA}$, i.e.:

$$H_T = \frac{1}{2}\frac{(I_E + R_M^2 I_M)(\Omega_E^r)^2}{S_\text{base}}. \tag{5.31}$$

Then, substitute H_T in (5.30) to get:

$$2H_T \frac{S_\text{base}}{(\Omega_E^r)^2}\dot{\Omega}_E = Q_E - R_M Q_S - R_M Q_D, \tag{5.32}$$

which can be simplified to:

$$2H_T \frac{d}{dt}\left(\frac{\Omega_E}{\Omega_E^r}\right) = \frac{Q_E}{S_\text{base}/\Omega_E^r} - \frac{R_M Q_S}{S_\text{base}/\Omega_E^r} - \frac{R_M Q_D}{S_\text{base}/\Omega_E^r}. \tag{5.33}$$

Now, choose the base quantities as follows:

$$\begin{aligned}
\Omega_{E_\text{base}} &= \Omega_E^r \\
Q_{E_\text{base}} &= S_\text{base}/\Omega_E^r \\
Q_{S_\text{base}} &= S_\text{base}/\Omega_M^r = S_\text{base}/R_M \Omega_E^r \\
Q_{D_\text{base}} &= S_\text{base}/\Omega_M^r = S_\text{base}/R_M \Omega_E^r.
\end{aligned} \tag{5.34}$$

Note that the p.u. speed of the engine is equal to the p.u. speed of the machine because:

$$\omega = \frac{\Omega_M}{\Omega_M^r} = \frac{\Omega_M/R_M}{\Omega_M^r/R_M} = \frac{\Omega_E}{\Omega_E^r}, \tag{5.35}$$

from (5.3) and (5.29). Now, substitute the base quantities in (5.33) to obtain:

$$\dot{\omega} = \frac{1}{2H_T}(q_E - q_S - q_D), \tag{5.36}$$

where q_S is the electromagnetic torque in p.u. given by (5.19), q_D is the damping torque proportional to speed deviation as [56]:

$$q_D = k_D(\omega - 1), \tag{5.37}$$

with k_D is the damping coefficient, and q_E is the torque produced by the engine in p.u. notation, given by $\frac{Q_E}{S_{\text{base}}/\Omega_E^r}$.

5.3 Simplified Model and Control Design

To begin with, let us define some parameters to simplify the notation.

$$
\begin{aligned}
k_E &= \frac{H_l \Omega_E^r}{S_{\text{base}}} \\
\Theta_E &= 1 - a_4 \Phi^{a_5} \\
\tilde{a}_2 &= a_2 \Omega_E^r \\
\tilde{a}_3 &= a_3 (\Omega_E^r)^2 \\
k_{f_1} &= k_D + 48 \frac{60}{2\pi} \frac{V_d(\Omega_E^r)^2}{2\pi\nu S_{\text{base}}} \\
k_{f_2} &= \frac{1000 V_d \Omega_E^r}{2\pi\nu S_{\text{base}}}(C_1 + 0.4\overline{S}_p^2) - k_D
\end{aligned}
\tag{5.38}
$$

Because the model we have so far is highly non-linear, we need to make the following assumptions to design the controller:

1. The time delay τ is negligible.
2. The quantity Θ_E is constant.

The assumptions above, especially the second one, may seem restrictive. However, we show in the next section of this chapter that without these assumptions the proposed controller still performs satisfactorily.

Now, the proposed model that can be used for control design of the shaft speed and the terminal voltage of the Genset, is described by the following three-state model:

$$
\begin{aligned}
\dot{\delta} &= \Omega_S^r(\omega - 1) \\
\dot{\omega} &= \frac{1}{2H_T}\left(k_E \Theta_E(a_1 + \tilde{a}_2\omega + \tilde{a}_3\omega^2)m_\phi - \frac{1}{2X_q}U^2 \sin(2\delta) - Ui_d \sin(\delta) - k_{f_1}\omega - k_{f_2}\right) \\
\dot{U} &= \left(\frac{R_a}{T'_{d0}X'_d} - \Omega_S^r(\omega - 1)\right)U \cot(\delta) + \frac{R_a X_d}{T'_{d0}X'_d \sin(\delta)}i_d - \frac{R_a}{T'_{d0}X'_d \sin(\delta)}E_F.
\end{aligned}
\tag{5.39}
$$

Note that the states of this model are the rotor angle, the shaft speed and the terminal voltage, which are easily measurable. The manipulated input controls are the fuel mass injected in the engine m_ϕ, and the voltage proportional to the field circuit voltage E_F. Finally, we model the stator current in $d-$axis i_d as a disturbance that is available for measurement.

Let us denote the states in the model above by $\mathbf{x}$ given by:

$$\mathbf{x} = [x_1, x_2, x_3]^T = [\delta, \omega, U]^T.$$

Let also the input controls of the above systems be $\mathbf{u} = [u_1, u_2]^T = [m_\phi, E_F]^T$. Let further the current i_d be denoted by d. Then, the final model writes:

$$\dot{\mathbf{x}} = f(\mathbf{x}) + g_1(\mathbf{x})u_1 + g_2(\mathbf{x})u_2 + p(\mathbf{x})d, \tag{5.40}$$

where,

$$f(\mathbf{x}) = \begin{bmatrix} \Omega_S^r(x_2 - 1) \\ \frac{-1}{2H_T}\left(\frac{1}{2X_q}x_3^2\sin(2x_1) + k_{f_1}x_2 + k_{f_2}\right) \\ \left(\frac{R_a}{T'_{d0}X'_d} - \Omega_S^r(x_2 - 1)\right)x_3\cot(x_1) \end{bmatrix},$$

$$g_1(\mathbf{x}) = \begin{bmatrix} 0 \\ \frac{k_E\Theta_E}{2H_T}\left(a_1 + \tilde{a}_2x_2 + \tilde{a}_3x_2^2\right) \\ 0 \end{bmatrix},$$

$$g_2(\mathbf{x}) = \begin{bmatrix} 0 \\ 0 \\ -\frac{R_a}{T'_{d0}X'_d\sin(x_1)} \end{bmatrix}, \text{ and } p(\mathbf{x}) = \begin{bmatrix} 0 \\ \frac{-1}{2H_T}x_3\sin(x_1) \\ \frac{R_aX_d}{T'_{d0}X'_d\sin(x_1)} \end{bmatrix}.$$

To this end, we use feedback linearisation to design a controller for this system. It is important in feedback linearisation to choose the appropriate outputs. As mentioned before, the sought controller is supposed to regulate the terminal voltage and the shaft speed, simultaneously. Thus, the terminal voltage is definitely one of the outputs. Instead of the shaft speed as a second output, we choose the angle for two reasons. Firstly, the model will be **exactly** feedback linearisable with these outputs. Secondly, controlling the rotor angle ensures the stability of the speed and thus the frequency of the generated currents. Hence, let the outputs be:

$$y_1 = h_1(\mathbf{x}) = x_1 - x_1^d$$
$$y_2 = h_2(\mathbf{x}) = x_3 - x_3^d, \tag{5.41}$$

where x_1^d and x_3^d are the desired angle and output voltage to be tracked. Following the well-known procedure for feedback linearisation as in [44] e.g., define the external state $z_1 = h_1(\mathbf{x}) = x_1 - x_1^d$, the derivative is obtained to be:

$$\dot{z}_1 = L_f h_1(\mathbf{x}) = \Omega_S^r(x_2 - 1), \tag{5.42}$$

where $L_f h_1(\mathbf{x})$ denotes the Lie derivative of $h_1(\mathbf{x})$ along the vector field $f(\mathbf{x})$. Then, define the state $z_2 = \Omega_S^r(x_2 - 1)$ whose derivative is given by:

$$\dot{z}_2 = L_f^2 h_1(\mathbf{x}) + u_1 L_{g_1}L_f h_1(\mathbf{x}) + u_2 L_{g_2}L_f h_1(\mathbf{x}) + dL_p L_f h_1(\mathbf{x}), \tag{5.43}$$

in which $L_{g_2} L_f h_1(\mathbf{x}) = 0$. Thus, the model in (5.40) has a relative degree two with respect to output y_1, i.e., $r_1 = 2$. Moreover, define the state $z_3 = h_2(\mathbf{x}) = x_3 - x_3^d$, and differentiate to get:

$$\dot{z}_3 = L_f h_2(\mathbf{x}) + u_1 L_{g_1} h_2(\mathbf{x}) + u_2 L_{g_2} h_2(\mathbf{x}) + d L_p h_2(\mathbf{x}), \qquad (5.44)$$

where $L_{g_1} h_2(\mathbf{x}) = 0$. Hence, the relative degree with respect to the output y_2 is one. Since, $r_1 + r_2 = 3$, the model can be feedback linearised exactly, and hence no *zero dynamics* will be encountered. Choosing the input control laws as follows:

$$u_1 = \frac{1}{L_{g_1} L_f h_1(\mathbf{x})} (-L_f^2 h_1(\mathbf{x}) - d L_p L_f h_1(\mathbf{x}) + v_1)$$

$$u_2 = \frac{1}{L_{g_2} h_2(\mathbf{x})} (-L_f h_2(\mathbf{x}) - d L_p h_2(\mathbf{x}) + v_2), \qquad (5.45)$$

where the explicit Lie derivatives are given by:

$$L_f^2 h_1(\mathbf{x}) = \frac{-\Omega_S^r}{2H_T} \left(\frac{1}{2X_q} x_3^2 \sin(2x_1) + k_{f_1} x_2 + k_{f_2} \right)$$

$$L_{g_1} L_f h_1(\mathbf{x}) = \frac{\Omega_S^r k_E \Theta_E}{2H_T} (a_1 + \tilde{a}_2 x_2 + \tilde{a}_3 x_2^2)$$

$$L_p L_f h_1(\mathbf{x}) = \frac{-\Omega_S^r}{2H_T} x_3 \sin(x_1), \qquad (5.46)$$

and

$$L_f h_2(\mathbf{x}) = (\frac{R_a}{T'_{d0} X'_d} - \Omega_S^r (x_2 - 1)) U \cot(x_1)$$

$$L_{g_2} h_2(\mathbf{x}) = -\frac{R_a}{T'_{d0} X'_d \sin(x_1)}$$

$$L_p h_2(\mathbf{x}) = \frac{R_a X_d}{T'_{d0} X'_d \sin(x_1)}, \qquad (5.47)$$

the external dynamics will be:

$$\dot{\mathbf{z}} = \begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \\ \dot{z}_3 \end{bmatrix} = A_z \mathbf{z} + B_z \mathbf{v} \qquad (5.48)$$

where $\mathbf{v} = [v_1, v_2]^T$ is an auxiliary stabilizing input control vector, and

$$A_z = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \ B_z = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}. \qquad (5.49)$$

The model above is linear and controllable, hence one can find a control law:

$$\mathbf{v} = -\mathbf{K_p} \mathbf{z}, \qquad (5.50)$$

that stabilizes the system in (5.48) by pole placement, for example. Note that the disturbance is used as a feed forward signal in this control law and hence completely decoupled from the output.

5.3.1 Simulation Results

The model in (5.40) with the control input laws in (5.45) was simulated in MAT-LAB R2012b. The synchronous machine we used for our simulations has the parameters listed in Table 5.1. Finding the coefficients $a_1, \ldots, a_5$ of the indicated efficiency model given in (5.23) is challenging. The specification sheets of Diesel engines do not include such functions. Hence, in order to find those parameters one needs to get experimental data and perform multi-variable non-linear regression or curve-fitting techniques on them. However, that is outside the scope of the work. Since, we are interested in assessing the proposed controller, and we assumed the validity of the claim proposed in [45] that the indicated efficiency can be approximated by a function of the form in (5.23), reasonable values are chosen in this work. Table 5.2 lists representative parameters of a Diesel engine. Further, Θ_E in (5.38) was assumed constant of 0.4, as stated before. Finally, the desired outputs to be tracked in (5.41) were set to 1.

It may not be suitable to simulate the model from zero initial states and see how to it will reach the desired values because Diesel engines are usually started for some time and then it will be loaded gradually. Thus, we chose to run the simulations from steady-state values, namely, $\delta(0) = 1$, $\omega(0) = 1$, and $U(0) = 1$, and then a perturbation was applied to see how the proposed controller would perform. The stator current in d-axis i_d was assumed of sinusoidal nature over 20 s, the simulation horizon, as:

$$i_d = 1 + 0.04\sin(5\pi t), \tag{5.51}$$

and it was fed forward to the controller. The initial values of the states were assumed equal to the steady state value. Then, a sudden step of 0.1 p.u. in the load torque, or in the terminal voltage, was introduced from $t = 5$ s till $t = 7$ s, as a perturbation that is not seen by the controller. The gain matrix $\mathbf{K_p}$ in (5.50) was obtained by pole placement.

In order to justify the choice of the poles, we need to notice from the structure of the matrices of the linearised model in (5.49), that the gain $\mathbf{K_p}$ for any poles vector $[p_1 \; p_2 \; p_3]$ has the following form:

$$\mathbf{K_p} = \begin{bmatrix} p_2 p_3 & -(p_2 + p_3) & 0 \\ 0 & 0 & -p_1 \end{bmatrix}. \tag{5.52}$$

Thus, the first pole controls the output voltage state $z_3 = x_3 - x_3^d = U - 1$, while the last 2 poles controls the angle and the speed. In fact, choosing the gains of the angle and speed states is a little tricky. To elucidate, high gains make the controller aggressive, which leads to an increase in the first control input signal, that is the fuel mass input $u_1 = m_\phi$. On the other hand, low gains make the angle drop to lower values, and hence the second control input u_2, that is the field circuit input voltage E_F, may reach undesired values or even explode due to the sine function in the denominator of the control law in (5.45), as can be seen in the Lie derivatives. Thus, choosing the poles and hence the gains is a trade-off between reducing the fuel mass input m_ϕ and avoiding unreasonable levels of the field circuit voltage input E_F. In order to illustrate the aforementioned analysis, two gains were tried in the simulations. Gain $\mathbf{K_1}$ obtained by placing the poles at [-5 -20 -5], and gain

K$_2$ obtained by placing the poles at [-5 -20 -25]. As mentioned earlier, a sudden increase step of 0.1 p.u. in the load torque was introduced between $t = 5$ s and $t = 7$ s. The results are shown in Fig. 5.2.

At $t = 5$ s when a sudden load increase is introduced, as can be seen from Fig. 5.2,

Quantity	I_M	X_d	X'_d	X_q	R_a	Frequency
	kg m^2	p.u.	p.u.	p.u.	p.u.	Hz
Value	250	2.0	0.25	1.0	0.004	60
Quantity	k_D	T'_{d0}	S_{base}	Ω^r_S	p	U_{base}
	p.u.	s	MVA	rad/s		V
Value	0.05	3.0	3.0	377	6	360

Table 5.1: The Parameters of the synchronous machine

Quantity	I_E	V_d	C_1	N_c	Ω^r_E	Rated Power
	kg m^2	dm^3	kPa		rad/s	MW
Value	1800	193	110	6	78.54	3
Quantity	$(A/F)_s$	R_M	$\overline{S}_p$	a_1	a_2	a_3
			m/s		s/m	s^2/m^2
Value	14.9	1.6	10	0.674	2.41×10^{-4}	-3.28×10^{-8}

Table 5.2: The Parameters of the Diesel engine

the speed decreases abruptly, and thus the angle will drop to another steady state-value. Note that the new steady-state value of the angle is not equal to the desired one, but it will stay stable because the speed reaches stability quickly. On the other hand, at $t = 7$ s, this increased sudden load is removed making the speed increase, and hence the angle, until the angle reaches the desired steady-state value and the speed stabilises again.

The terminal voltage is not affected by this sudden increase, however. This may not be realistic, since one expects a disruption in the terminal voltage when the rotational speed is disrupted. The reason behind that is the fact that the proposed model is too simplified that it is best suited for control design, and it may not be suitable for accurate simulations. Anyway, since we are interested in the behaviour of the controllers and the interaction between them to stabilize the states, this model probably suffices for the mentioned targets. Although the terminal voltage U is not affected by this sudden increase in the load torque, the field circuit voltage E_F increases to compensate for the decrease in the speed at $t = 5$ s. Conversely, E_F decreases at $t = 7$ s to compensate for the increase in the speed.

Now let us try to comment on the choice of the gains. By using the function *place*

in MATLAB, the gains $\mathbf{K_1}$ and $\mathbf{K_2}$ were obtained at the given poles to be:

$$
\begin{aligned}
\text{at poles} \quad [-5 \ -20 \ -5] &\longrightarrow \mathbf{K_1} = \begin{bmatrix} 100 & 25 & 0 \\ 0 & 0 & 5 \end{bmatrix} \\
\text{at poles} \quad [-5 \ -20 \ -25] &\longrightarrow \mathbf{K_2} = \begin{bmatrix} 500 & 45 & 0 \\ 0 & 0 & 5 \end{bmatrix}.
\end{aligned}
\tag{5.53}
$$

So, the elements of $\mathbf{K_1}$ corresponding to the angle and speed are less than those of $\mathbf{K_2}$. As can be seen in Fig. 5.2, when the load torque step is introduced at $t = 5$ s, the angle drops to a lower value with $\mathbf{K_1}$ because the speed takes longer time to retain its desired value, 1 p.u. Thus, the field circuit input voltage E_F increases to higher values. In opposition, increasing the gains of the angle and speed makes the speed retain its desired values quickly by increasing the first input control (m_ϕ). In a nutshell, reducing the gains of the angle and speed moves the burden of retaining the steady-state values from the first input control (m_ϕ) to the second input control (E_F). Hence, fine tuning is required to choose the gains because it is a trade-off between keeping the mass fuel input minimized and keeping the field circuit input at reasonable level. However, the most important advantage of this controller is the interconnection between the two input controls. That is to say, both inputs contribute to keeping the system stable at the same time and not separately.

Let us now see how the controllers would react if the terminal current is perturbed. A sudden increase of 0.25 p.u. in the terminal current in the $d-$axis i_d was introduced between $t = 5$ s and $t = 7$ s without feeding it forward to the controller. The results obtained are depicted in Fig. 5.3. It is shown in the figure that all states are affected by this step because the current affects the electromagnetic torque and the terminal voltage. The gains $\mathbf{K_1}$ and $\mathbf{K_2}$ have almost the same controlling effect. Increasing the gain increases the fuel mass input m_ϕ, and reduces the field circuit voltage E_F. One more time, we see that the two input controls are reacting with each other to keep stability.

5.4 Uncertainties

In this section, we discuss the performance of the proposed controller when the time delay and air/fuel ratio are taken in consideration.

5.4.1 Time Delay

On one hand, the model in (5.39) is highly non-linear, and the time delay is not constant due to its dependence on the speed, as can be noted from (5.21). Thus, techniques like *Smith predictor* may not be suitable, but time delay compensation for the model in (5.39) can be an interesting topic for future work. On the other hand, the time delay introduced by the Diesel engine is not so large, that is why it is usually neglected. What we try to show here, by simulations, is that small time delays will not affect the controller, significantly.
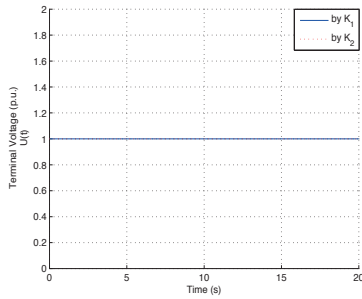
Generally speaking, a typical time delay of a Diesel engine is in order of milliseconds. The model in (5.40) with the control laws in (5.45) was simulated one more
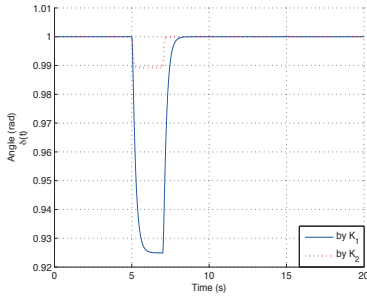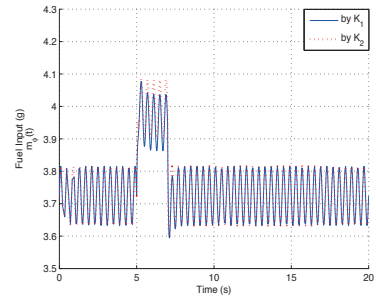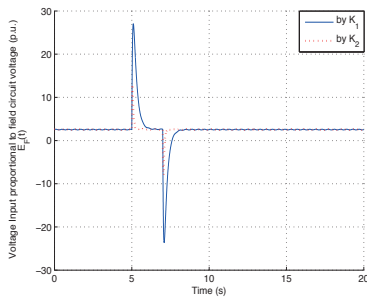
(a) Angle $x_1(t) = \delta$
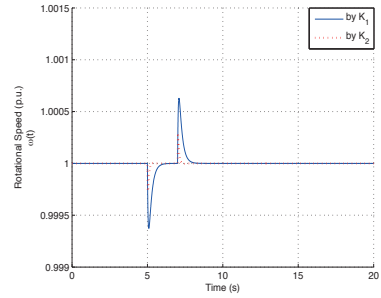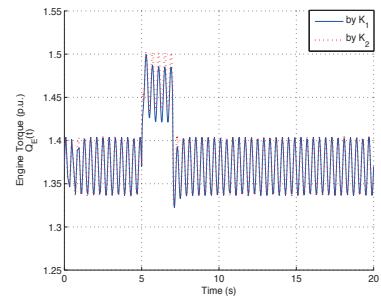
(b) Rotational speed $x_2(t) = \omega$

(c) Terminal Voltage $x_3(t) = U$
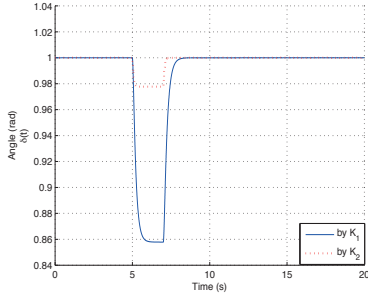
(d) Fuel mass input $u_1(t) = m_\phi$

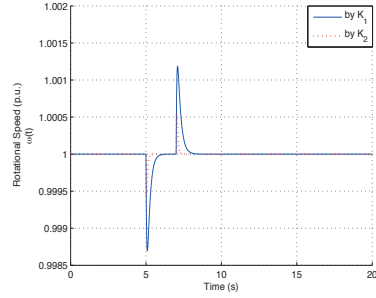(e) Field circuit input $u_2(t) = E_F$
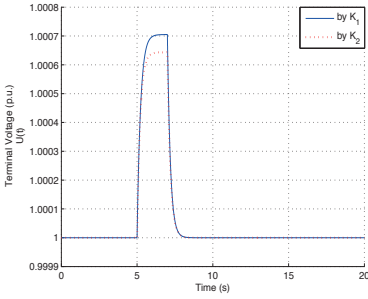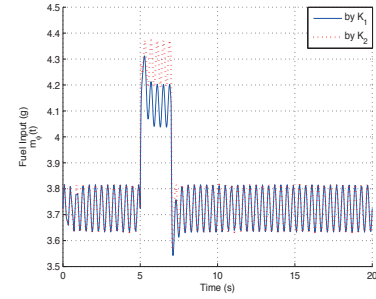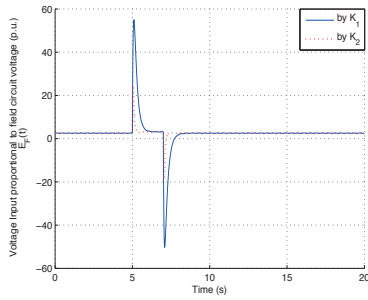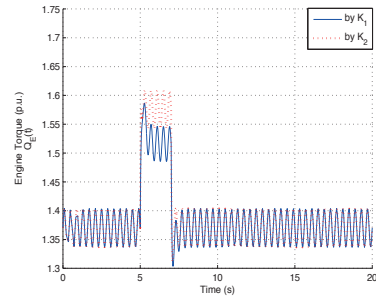
(f) Engine Torque $Q_E(t)$

Figure 5.2: Results of the simulation of the model in (5.40) with a sudden step in the load torque by the gains: $\mathbf{K_1}$ (solid) and $\mathbf{K_2}$ (dotted)
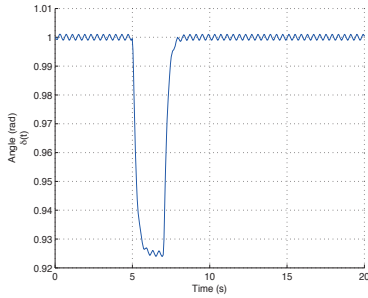
(a) Angle $x_1(t) = \delta$

(b) Rotational speed $x_2(t) = \omega$

(c) Terminal Voltage $x_3(t) = U$

(d) Fuel mass input $u_1(t) = m_\phi$

(e) Field circuit input $u_2(t) = E_F$

(f) Engine Torque $Q_E(t)$

Figure 5.3: Results of the simulation of the model in (5.40) with a sudden step in stator current by gains: $\mathbf{K_1}$ (solid) and $\mathbf{K_2}$ (dotted)

time, with the gain $\mathbf{K_1}$ in (5.53). Same parameters of the Diesel engine and synchronous machine, given in Tables 5.2 and 5.1 respectively, were used here as well. The disturbance stator current i_d was assumed as in (5.51). Two time delays of 0.01 s and 0.04 s were introduced in the engine torque as in (5.28). The same step in the load torque was introduced, as in the previous subsection. The states trajectories are shown in Fig. 5.4, without the control inputs signals because they are identical to those obtained in Fig. 5.2 with the gain $\mathbf{K_1}$.

As can be seen in Fig. 5.4, the proposed controller could keep the system stable around the desired outputs with a *limit cycle* due to the time delay. Intuitively, the amplitude of the limit cycle increases with the time delay. The argument about the states trajectories with the increase step in the load torque, in the previous subsection, is valid here, as well. However, we need to mention here that higher gains may lead to loss of stability, especially with increasing time delays. We can motivate that by the following argument. When the gain elements corresponding to the angle and speed increase, the first input control m_ϕ plays the major role in stabilizing the system. Thus, the the mass fuel input m_ϕ increases, and so does the engine torque. This magnifies the effect of the time delay since it is imposed by the engine torque. In fact, choosing the right gain is trickier now than before. Low gains of angle and speed would increase the field circuit input voltage E_F to unreasonable levels, whereas high gains of angle and speed would not only increase the fuel mass input m_ϕ, but it may also lead to losing the stability due to the time delay imposed by the engine torque.

5.4.2 Uncertainty from Air/Fuel Ratio

A combustion is called stoichiometric when "there is just enough oxygen for conversion of all the fuel into completely oxidized products [41]", and the air/fuel ratio corresponding to this situation is usually referred to as the stoichiometric air/fuel ratio. Generally speaking, air/fuel ratio is not desired to be too high or too low. High values of air/fuel ratio (usually called *lean* conditions) lead to incomplete combustion and hence decrease the efficiency, while low values (usually called *rich* condition)increase the emissions [36]. In *Spark Ignition* (SI) engines, a three-way *catalyst* is used to reduce the emissions. For this catalyst to perform efficiently, the air/fuel ratio should be in a narrow band around the stoichiometric value, i.e. the relative air/fuel ratio λ defined in (5.24) should be regulated to unity [36].

In opposition, in CI engines like Diesel engines, the situation is more complicated. Firstly, emission reduction is taken care of by controlling the air path through the EGR, which is not necessarily at the stoichiometric air/fuel ratio. Secondly, the torque developed by the engine depends on the air/fuel ratio. Thus, the air/fuel ratio in Diesel engines is not desired to be regulated around a fixed value, although this can be done through the EGR and turbocharger. For example, the authors in [1] suggested an air/fuel ratio controller for vehicles Diesel engines around optimal set points determined from a static engine map, function of the engine speed and the *mean effective pressure*, obtained from steady-state measurements [1]. Nevertheless, controllers of the modern Diesel engine usually include limiters to prevent the air/fuel ratio from reaching very high or low values, as explained above. Furthermore, the air/fuel ratio can be measured easily in modern engines by an air
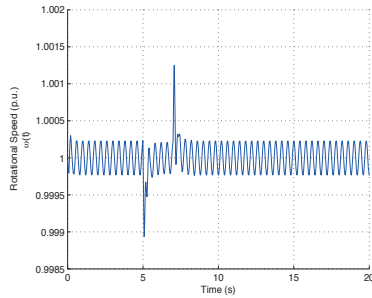
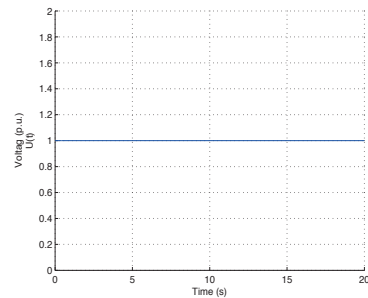(a) Angle $x_1(t) = \delta$, $\tau = 0.01$ s
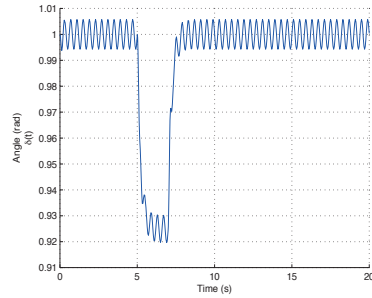
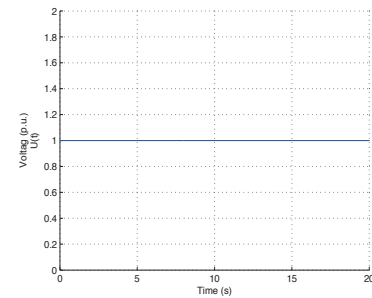(b) Angle $x_1(t) = \delta$, $\tau = 0.04$ s
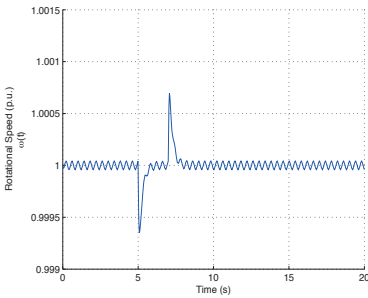
(c) Rotational speed $x_2(t) = \omega$, $\tau = 0.01$ s

(d) Rotational speed $x_2(t) = \omega$, $\tau = 0.04$ s

(e) Terminal Voltage $x_3(t) = U$, $\tau = 0.01$ s

(f) Terminal Voltage $x_3(t) = U$, $\tau = 0.04$ s

Figure 5.4: Results of the simulation of the model in (5.40) with a sudden step in the load torque for time delay of 0.01 s and 0.04 s

sensor in the exhaust manifold.

After the discussion above, one can say that the proposed model in (5.39) can be modified by dealing with the quantity Θ_E as a measurable disturbance, such as the current i_d. However, this may not be a clever choice, not only due to the time delay in the model but also because air/fuel ratio varies with fuel mass input. Thus, we propose dealing with the quantity Θ_E as an uncertain parameter with known nominal value, as follows:

$$\Theta_E = \Theta_{E_0} + \delta_{\Theta_E}, \tag{5.54}$$

where Θ_{E_0} is the measurable nominal value, and δ_{Θ_E} is unmeasurable. Several authors in the field of *robust control* have suggested techniques to treat the problem of the uncertain parameters. One of the most attractive techniques in the literature to deal with uncertain parameters with feedback linearisation is the one proposed in [53] for square MIMO systems, which was expanded later in [80] for non-square models. The authors in [53] suggested expanding the control law, found by feedback linearisation, by Taylor's series around the nominal values of the uncertain parameters. Then, instead of using pole placement to find the gain, the authors suggested using H_2/H_∞ synthesis and they proved, based on the work by [52], that the solution of the mixed H_2/H_∞ synthesis problem stabilizes the model with the uncertain parameters [53]. In spite of the elegance of their proposal, the authors did not provide enough information on choosing the weighting matrices, which makes the proposal cumbersome to apply, at least in our model.

Anyway, we show in the sequel, by simulations, that modelling Θ_E as in (5.54) does not influence the proposed controller. The model in (5.40) was simulated one more time, neglecting the time delay. Same parameters of the Diesel engine and synchronous machine, given in Tables 5.2 and 5.1 respectively, were used here as well. The disturbance stator current i_d was assumed as in (5.51). The same step in the load torque was introduced, as in the previous subsections. The nominal value Θ_{E_0} was assumed 0.4, and the variation δ_{Θ_E} was assumed a sinusoidal function as $0.2\sin(2\pi t)$. In order to mimic the situation, that the nominal value of the uncertain parameter is fed forward to the controller and variation is not, the control laws in (5.45) were calculated at the nominal value Θ_{E_0}. The states trajectories are shown in Fig. 5.5, with the control inputs signals.

Our experiments emphasized the fact we showed previously, that low gains make the second control input E_F reach high levels. The uncertainty make E_F increase even more. Hence, we had to increase the gains. Two gains are shown in the results depicted in Fig. 5.5, as follows:

$$\begin{aligned}
\text{at poles} \quad [-5 \ -20 \ -50] &\longrightarrow \mathbf{K_3} = \begin{bmatrix} 1000 & 70 & 0 \\ 0 & 0 & 5 \end{bmatrix} \\
\text{at poles} \quad [-5 \ -20 \ -100] &\longrightarrow \mathbf{K_4} = \begin{bmatrix} 2000 & 120 & 0 \\ 0 & 0 & 5 \end{bmatrix}.
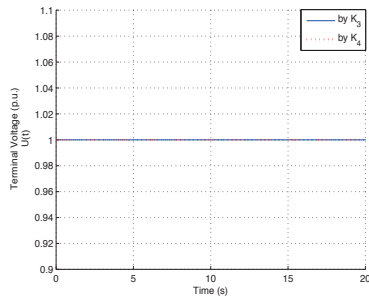\end{aligned} \tag{5.55}$$

Due to the uncertainty, limit cycles are noticed again in the trajectory of the angle and the speed. The amplitude of this limit cycle in the speed trajectory is less than 0.1%, which could be very conservative because usually in power systems variation
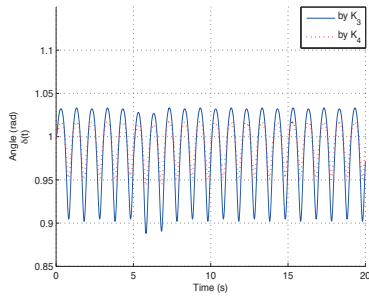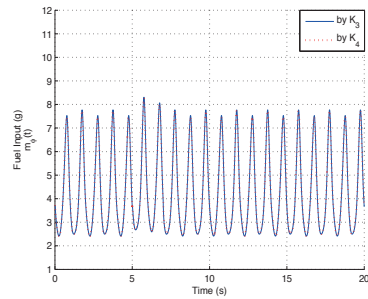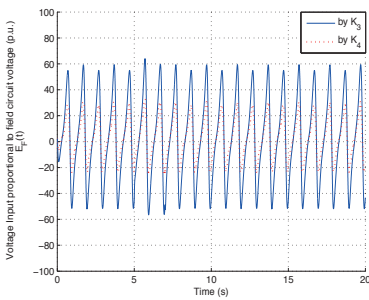
(a) Angle $x_1(t) = \delta$
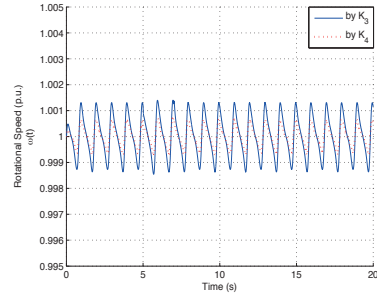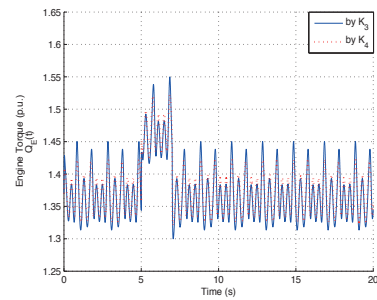
(b) Rotational speed $x_2(t) = \omega$

(c) Terminal Voltage $x_3(t) = U$

(d) Fuel mass input $u_1(t) = m_\phi$

(e) Field circuit input $u_2(t) = E_F$

(f) Engine Torque $Q_E(t)$

Figure 5.5: Results of the simulation of the model in (5.40) with a sudden step in the load torque by the gains: $\mathbf{K_3}$ (solid) and $\mathbf{K_4}$ (dotted) with Θ_E modelled as in (5.54)

of frequency less than 1% are acceptable. For the angle trajectory, the situation is worse, especially when we note that the setpoint to be tracked is shifted a little bit below 1. To sum up, we can say that the controller could keep the speed oscillating around the setpoint with a small amplitude, even when 50% uncertainty in Θ_E is applied to the model.

### 5.4.3   Uncertainty from Air/Fuel Ratio with Time Delay

We show finally the results of our simulations when the uncertainty is applied to the system and the time delay is taken in consideration. From the parameters in Table 5.2, and by using (5.21), we can say that for this particular Diesel engine the time delay does not exceed 0.01 s. Thus, we simulated the model again, with same step in the load torque, but this time with 0.01 s time delay in the torque model. Θ_E was modelled as in the previous subsection. The same gains $\mathbf{K_3}$ and $\mathbf{K_4}$ in (5.55) were used here, as well. The results are shown in Fig. 5.6.

We can note that the results shown in Fig. 5.6 are indifferent from those shown in Fig. 5.5, except for some disturbance in the angle and speed trajectories when the sudden step in the load torque was first introduced at $t = 5$ s and when it was removed at $t = 7$ s, when the higher gain $\mathbf{K_4}$ was used. Of course, this is expected, since increasing the gain manifests the dependence on the first control input m_ϕ which, in turn, increases the effect of the time delay, as explained before.

5.5   Conclusion

In this chapter a control-oriented model of the Genset, that comprises a Diesel engine and a synchronous generator, was proposed. The air/fuel ratio was modelled as an uncertain parameter of measurable nominal value, and unmeasurable variation. A controller to regulate the shaft speed and the terminal voltage, simultaneously, was designed by using feedback linearisation. The controller consists of two control inputs, the fuel mass input and the field circuit voltage. The suggested controller is supposed to coordinate the ordinary PID controllers, namely the AVR and the speed governor. Simulations of the proposed model with the proposed control laws were provided. The simulations show the following:

1. The suggested controller uses both control inputs to stabilize the system, simultaneously, in contrast to the ordinary PID where no direct communication exists between the speed governor and the AVR.

2. Choosing the gains of the control laws is a trade-off between reducing the fuel consumption and avoiding high values of field circuit voltage. Increasing the gain corresponding to the fuel input increases the fuel consumption, and reduces the dependence on the field circuit voltage, and vice versa.

3. Since the time delay in the engine torque model is not so large, with careful choosing of the gains, the proposed controller can still stabilize the states with limit cycles of very small amplitude.

4. The uncertainty of the air/fuel ratio increases the amplitude of the limit cycles reached by the states, and may change the desired set points to be tracked, especially for the angle state trajectory. However, the set point of the speed
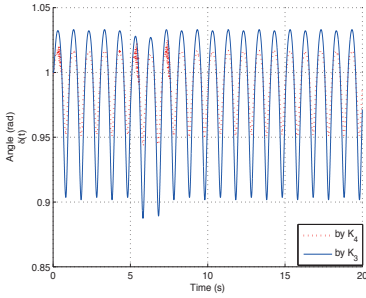
(a) Angle $x_1(t) = \delta$
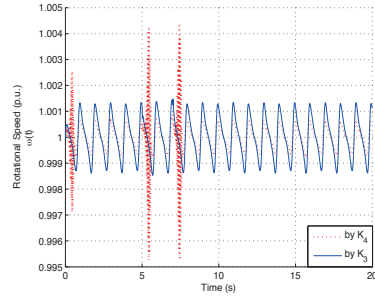
(b) Rotational speed $x_2(t) = \omega$
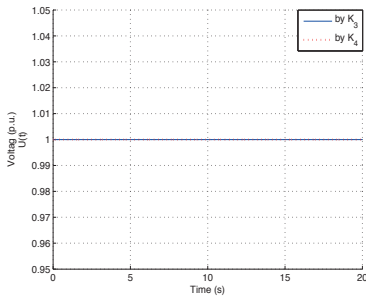
(c) Terminal Voltage $x_3(t) = U$
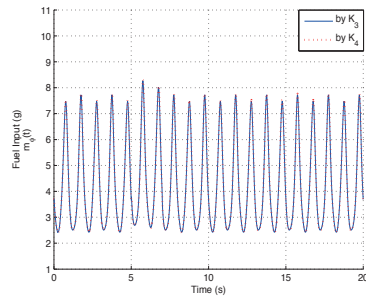
(d) Fuel mass input $u_1(t) = m_\phi$
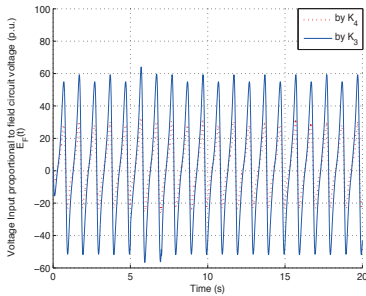
(e) Field circuit input $u_2(t) = E_F$
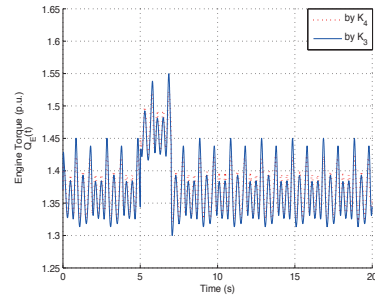
(f) Engine Torque $Q_E(t)$

Figure 5.6: Results of the simulation of the model in (5.40) with a sudden step in the load torque by the gains: $\mathbf{K_3}$ (solid) and $\mathbf{K_4}$ (dotted) with Θ_E modelled as uncertain parameter and time delay of 0.01 s

is not changed, and the amplitude of its limit cycle can be minimized with astute tuning of gains. Thus, the proposed controller can be considered robust to the uncertainties.

Chapter 6

Propellers and Genset

In some designs of power systems for marine vessels, a large-size or medium-size Diesel engine is used to drive one synchronous machine to generate electricity, and the main propeller, simultaneously, through a gear box. Such systems have several modes of operation. That is why the manufacturers refer to such shaft generators as PTI/PTO. In this chapter, we extend the model developed in the previous chapter by including the propeller, and hence the controller is modified to deal with this change.

6.1 Introduction

The power systems on marine vessels take many configurations and designs according to their purposes, and size. Companies in the field compete for designing and manufacturing better systems regarding efficiency, reliability, fuel saving, and environmental friendliness. The Diesel engine is most commonly used due to its efficiency, and low cost [35]. The propellers are driven either electrically (by a motor of any type), or mechanically (e.g. a Diesel engine) [11], [87]. In both cases, Diesel engines or any other prime movers are needed aboard to drive synchronous machines to generate electricity. Spurred by environmental reasons and the urge to save fuel, some systems that have been developed for marine vessels recently, comprise propellers that can be driven mechanically and/or electrically. Such designs exploit a large or medium-size Diesel engine that can drive both the main propeller, and synchronous machine as shown in Fig. 6.1. The synchronous machine in such designs works either as a motor or as a generator. In PTI mode, the machine is driven by the electric power from the bus as a motor, whereas when it is driven by the mechanical power produced by the Diesel engine to generate electric power, it is said to be working in the PTO mode. Hence, this system can be considered as a *Hybrid system*.

To the authors best knowledge, systems like the one in Fig. 6.1 have not been treated in the literature as one system, maybe due to its complexity. The complexity of the system arises from the fact that the propeller can be driven electrically, or mechanically. In addition, the objectives of the controller may change according to the mode of operation. For example, in PTO mode at least, the shaft speed
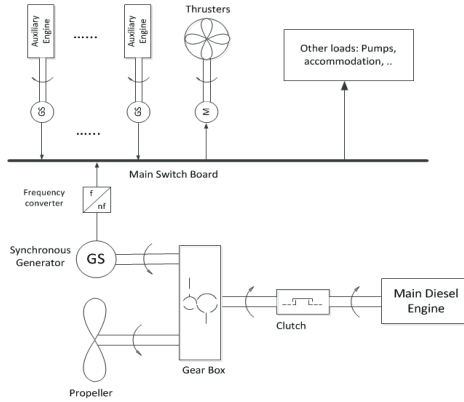
Figure 6.1: Schematic of the main network of the power systems on board subject of research.

must be kept constant to reduce large or undesired oscillations in the frequency of the electric power produced by the machine. In other modes, the target could be to control the propeller torque. In practice, the manufacturers depend on the low-level controllers, the AVR and speed governor, and high level controllers are used to achieve the control objectives depending on the mode of operation. Most of the authors in this arena differentiate between electrically-driven and mechanically driven propellers. For example, the authors in [66], [40], and [43] proposed different approaches to control the Genset, but all of them considered the propeller as part of the load since it is driven electrically. Also, the authors in [42], proposed a propulsion control strategy for *icebreakers*, but the propeller in their model was also driven electrically.

Inasmuch as we have proposed a model of the Genset in the previous chapter, we try to extend this model to include the propeller in the current chapter. Naturally, one would suggest to include the propeller torque in the load torque in the shaft dynamics in (5.30), and then consider the propeller torque as a measurable disturbance that can be fed forward to the controller. However, the propeller torque is not easy to measure in practice, and thus designing an observer can be considered. Alternatively, we propose in this chapter to include the propeller torque in our model, and hence the controller can be modified to compensate for it. Normally, the propellers used in such shaft generators are *Controllable Pitch Propeller* (CPP), for some reasons that will be explained later. Thus, the *pitch angle* of the propeller can be used as a control input to control the torque and thrust. In our proposal, the model proposed in [10] for propellers is utilised. In addition, we depended on the fact that high level controllers such as thrust allocation algorithms are usually used to determine the pitch angle. Hence, the pitch angle can be considered available for the proposed controller, as will be shown.

The model proposed is designed for one mode of operation, namely the PTO, in which the Diesel engine drives both the synchronous machine and the main propeller, simultaneously. Hence, our control objective, in addition to regulating the

terminal voltage, is to regulate the shaft speed to reduce undesired fluctuations in the frequency. We show by simulations that the proposed controller regulates the shaft speed and the terminal voltage, even when the propeller torque is changed abruptly by changing, e.g., the pitch angle. Besides, we show that the proposed controller can be considered robust to small uncertainties and very short time delays.

6.2 Shaft Dynamics

In the previous chapter, the shaft dynamics of the Genset was presented as in (5.30). In this section, we modify the model to include the propeller.
The Diesel engine in this mode drives both the synchronous machine and the main propeller. Thus, the propeller must be included in the shaft dynamics. To begin with, let the rotational speed of the propeller in mechanical `rad/s` be denoted by Ω_P, then this speed is related to the Diesel engine speed by:

$$\Omega_P = R_P \Omega_E, \tag{6.1}$$

where R_P is the gear ratio with respect to the propeller side. Then, by applying Newton's second law of rotating objects to the mechanical system, the model in (5.30) can be modified to:

$$(I_E + R_M^2 I_M + R_P^2 I_P)\dot{\Omega}_E = Q_E - R_M Q_S - R_M Q_D - R_P Q_P, \tag{6.2}$$

where I_P is the moment of inertia in `kg m`^{2} of the propeller and the *added mass*, and Q_P is the propeller torque in N.m that will be discussed in details in the next section. Now, mimicking the technique used previously, define the `p.u.` inertia constant $\tilde{H}_T$ as:

$$\tilde{H}_T = \frac{1}{2} \frac{(I_E + R_M^2 I_M + R_P^2 I_P)(\Omega_E^r)^2}{S_{\text{base}}}. \tag{6.3}$$

By using the same base quantities in (5.34), and by choosing the base quantity of the propeller torque as:

$$Q_{P_{\text{base}}} = S_{\text{base}}/\Omega_P^r = S_{\text{base}}/R_P \Omega_E^r, \tag{6.4}$$

one can obtain the following relation to describe the dynamics of the shaft:

$$\dot{\omega} = \frac{1}{2\tilde{H}_T}(q_E - q_P - q_S - q_D), \tag{6.5}$$

where q_P is the propeller torque in `p.u.` As was done before, we neglect in the above model the moment of inertia of the shaft and the frictional torque, although they can be easily added to the model without affecting its performance. However, dealing with the moment of inertia of the hydrodynamic added mass may not be straightforward.
When a force accelerates a body in a fluid, it must also accelerate the fluid around it. The additional force required to accelerate the fluid around the body is usually modelled by increasing the mass of the body by an additional virtual mass called

added mass. The added mass of a propeller depends on many factors such as
[94]: the shaft speed, advance speed, propeller submergence, and of course, the
dimensions of the propeller. Hence, complex models are needed to determine the
added mass, accurately. One can even think of observers to estimate the added
mass. However, since the added mass varies with the aforementioned factors, we
propose treating it as an uncertain parameter, as was done with the air/fuel ratio
in the previous chapter, that is to ay:

$$I_P = I_{P_0} + \delta_{I_P}, \tag{6.6}$$

where I_{P_0} is the nominal value of the moment of inertia and the added mass which
we assume measurable or observable, and δ_{I_P} is its uncertain variation. Needless to
say that the above model requires verification and identification by analysing ex-
perimental data. However, what we would like to show that the controller proposed
for the Genset in the previous chapter can be extended for the current system, and
another uncertainty in the model would not affect its performance, as will shown
in the sequel.

6.3   Propeller Model

In general, the propeller torque is modelled by [11], [87], and [82]:

$$Q_P = \rho D^5 |n_P| n_P K_Q \tag{6.7}$$

where n_P is the rotational speed in rps, ρ is the density of water in $\mathtt{kg/m^3}$, D is the
propeller diameter in m, and K_Q is the dimensionless propeller torque coefficient.
The coefficient K_Q is not constant, and it is usually modelled in open-water char-
acteristic as a function of the *advance ratio J*, and the *pitch angle* θ, depending on
the type of the propeller. The advance ratio J is defined by [25]:

$$J = \frac{V_a}{n_P D}, \tag{6.8}$$

where V_a is the advance speed in $\mathtt{m/s}$, that is the speed of the water in the wake
of the hull. The pitch angle θ is defined by [94]:

$$\theta = \arctan(\frac{P/D}{\pi}), \tag{6.9}$$

where P/D is the pitch to diameter ratio, and the pitch P is the distance travelled
by the propeller in the axial direction after one revolution. It is worth mentioning
that the propeller thrust is modelled as the torque, but with the 4th power of the
diameter D instead of the fifth in (6.7). From the model above, one can note that
the torque of the propeller, and hence the thrust, can be controlled by controlling
the rotational speed n_P, the pitch angle θ, or both. Regarding this aspect, the
propellers are classified into two main types: *Fixed Pitch Propeller* (FPP) and
CPP.

As the name suggests, for FPP the pitch is fixed, and thus control is done through
the speed only. On the other hand, for CPP two degrees of freedom are available.

For the shaft generator considered here, CPP are usually utilised. To elucidate, there are different modes of operation of the system shown in Fig. 6.1, as explained earlier. In this work we confined ourselves to one mode only, namely PTO in which the engine drives both the synchronous machine to generate electricity and the main propeller. As explained in the previous chapter, the rotational speed in this mode must be kept constant to regulate the frequency of the generated electricity. Thus, a CPP must be used to give one degree of freedom to control the thrust and torque of the propeller.

The author in [25], based on the work in [10], described a bilinear model of the torque of the CPP as:

$$Q_P = Q_0 n_p |n_P| + Q_{|n|n} |\theta| |n_P| n_P + Q_{|n|V_a} \theta |n_P| V_a, \tag{6.10}$$

where Q_0, $Q_{|n|n}$, and $Q_{|n|V_a}$ are constants that can be found from experimental data. The author in [10] stated that the above model is fairly accurate in steady ahead ($V_a > 0$, $n_P > 0$), and steady astern ($V_a < 0$, $n_P > 0$) cases, but not otherwise. For the purposes of our control design, we believe that this model is suitable. Furthermore, since we are interested in regulating the speed around a specific set point, the model above can be simplified to:

$$Q_P = Q_0 n_p^2 + Q_{|n|n} |\theta| n_P^2 + Q_{|n|V_a} \theta n_P V_a. \tag{6.11}$$

6.4 Control Design

In order to insert the propeller torque in (6.11) in the shaft dynamics in (6.5), we need to describe it in p.u. notation. Thus, by using (6.4) and by choosing $\Omega_{P_{\text{base}}} = R_P \Omega_E^r$, define the following parameters:

$$k_{P_1} = \frac{Q_0 R_P \Omega_E^r}{S_{\text{base}}} \left(\frac{R_P \Omega_E^r}{2\pi}\right)^2$$

$$k_{P_2} = \frac{Q_{|n|n} R_P \Omega_E^r}{S_{\text{base}}} \left(\frac{R_P \Omega_E^r}{2\pi}\right)^2$$

$$k_{P_3} = \frac{Q_{|n|V_a} R_P \Omega_E^r}{S_{\text{base}}} \left(\frac{R_P \Omega_E^r}{2\pi}\right). \tag{6.12}$$

Then, the final model of the shaft generator in PTO mode can be expressed as:

$$\dot{\mathbf{x}} = \tilde{f}(\mathbf{x}) + g_1(\mathbf{x})u_1 + g_2(\mathbf{x})u_2 + \begin{bmatrix} p_1 & p_2 & p_3 \end{bmatrix}(\mathbf{x}) \begin{bmatrix} d_1 \\ d_2 \\ d_3 \end{bmatrix}, \tag{6.13}$$

where the states $\mathbf{x}$, the control inputs u_1, u_2, and the functions $g_1(\mathbf{x})$, and $g_2(\mathbf{x})$ are as defined for the model in (5.40), after replacing H_T with $\tilde{H}_T$ in the function $g_1(\mathbf{x})$. The function $\tilde{f}(\mathbf{x})$ is now given by:

$$\tilde{f}(\mathbf{x}) = \begin{bmatrix} \Omega_S^r(x_2 - 1) \\ \frac{-1}{2\tilde{H}_T}\left(\frac{1}{2X_q}x_3^2 \sin(2x_1) + k_{f_1}x_2 + k_{f_2} + k_{P_1}x_2^2\right) \\ \left(\frac{R_a}{T_{d0}'X_d'} - \Omega_S^r(x_2 - 1)\right)x_3 \cot(x_1) \end{bmatrix}.$$

115

Further, just like the stator current i_d which we denote here by d_1, we model the pitch angle $|\theta|$ and the product θV_a as measured disturbances, denoted by d_2 and d_3, respectively. We can motivate this choice by the following. In marine control systems, there are usually high level controllers to control the propeller thrust or torque. Since, the advance speed V_a is required for those controllers, it must be estimated by special observers. Besides, the pitch angle θ is the only input that can be used to control the thrust or torque of the CPP, since the speed is fixed in this mode. Hence, it can be assumed available to be used as a feed forward signal. The function $p_1(\mathbf{x})$ is as same as $p(\mathbf{x})$ defined for the model in (5.40), after replacing H_T with $\tilde{H}_T$. Finally, the functions $p_2(\mathbf{x})$ and $p_3(\mathbf{x})$ are described by:

$$p_2(\mathbf{x}) = \begin{bmatrix} 0 \\ \frac{-1}{2\tilde{H}_T} k_{P_2} x_2^2 \\ 0 \end{bmatrix}, \ p_3(\mathbf{x}) = \begin{bmatrix} 0 \\ \frac{-1}{2\tilde{H}_T} k_{P_3} x_2 \\ 0 \end{bmatrix}.$$

Now, choosing the outputs as in (5.41), we apply the feedback linearisation technique again. Let $z_1 = h_1(\mathbf{x}) = x_1 - x_1^d$, the derivative is obtained to be:

$$\dot{z}_1 = L_{\tilde{f}} h_1(\mathbf{x}) = \Omega_S^r (x_2 - 1). \tag{6.14}$$

Let also $z_2 = \Omega_S^r (x_2 - 1)$, with the derivative given by:

$$\begin{aligned} \dot{z}_2 =& L_{\tilde{f}}^2 h_1(\mathbf{x}) + u_1 L_{g_1} L_{\tilde{f}} h_1(\mathbf{x}) + u_2 L_{g_2} L_{\tilde{f}} h_1(\mathbf{x}) \\ &+ d_1 L_{p_1} L_{\tilde{f}} h_1(\mathbf{x}) + d_2 L_{p_2} L_{\tilde{f}} h_1(\mathbf{x}) + d_3 L_{p_3} L_{\tilde{f}} h_1(\mathbf{x}), \end{aligned} \tag{6.15}$$

where $L_{g_2} L_{\tilde{f}} h_1 = 0$. Further, define the state $z_3 = h_2(\mathbf{x}) = x_3 - x_3^d$, and differentiate to get:

$$\begin{aligned} \dot{z}_3 =& L_{\tilde{f}} h_2(\mathbf{x}) + u_1 L_{g_1} h_2(\mathbf{x}) + u_2 L_{g_2} h_2(\mathbf{x}) \\ &+ d_1 L_{p_1} h_2(\mathbf{x}) + d_2 L_{p_2} h_2(\mathbf{x}) + d_3 L_{p_3} h_2(\mathbf{x}), \end{aligned} \tag{6.16}$$

where $L_{g_1} h_2 = L_{p_2} h_2 = L_{p_3} h_2 = 0$. Again, we notice that $r_1 + r_2 = 3$, and thus the model can be feedback linearised, exactly. Choosing the input control laws as follows:

$$u_1 = \frac{1}{L_{g_1} L_{\tilde{f}} h_1(\mathbf{x})} (-L_{\tilde{f}}^2 h_1(\mathbf{x}) - d_1 L_{p_1} L_{\tilde{f}} h_1(\mathbf{x}) - d_2 L_{p_2} L_{\tilde{f}} h_1(\mathbf{x}) - d_3 L_{p_3} L_{\tilde{f}} h_1(\mathbf{x}) + \tilde{v}_1)$$

$$u_2 = \frac{1}{L_{g_2} h_2(\mathbf{x})} (-L_{\tilde{f}} h_2(\mathbf{x}) - d_1 L_{p_1} h_2(\mathbf{x}) + \tilde{v}_2), \tag{6.17}$$

where the explicit Lie derivatives are given by:

$$L_{\tilde{f}}^2 h_1(\mathbf{x}) = \frac{-\Omega_S^r}{2H_T}\left(\frac{1}{2X_q}x_3^2\sin(2x_1) + k_{f_1}x_2 + k_{f_2} + k_{P_1}x_2^2\right)$$

$$L_{g_1}L_{\tilde{f}}h_1(\mathbf{x}) = \frac{\Omega_S^r k_E \Theta_E}{2\tilde{H}_T}(a_1 + \tilde{a}_2 x_2 + \tilde{a}_3 x_2^2)$$

$$L_{p_1}L_{\tilde{f}}h_1(\mathbf{x}) = \frac{-\Omega_S^r}{2\tilde{H}_T}x_3\sin(x_1)$$

$$L_{p_2}L_{\tilde{f}}h_1(\mathbf{x}) = \frac{-\Omega_S^r}{2\tilde{H}_T}k_{P_2}x_2^2$$

$$L_{p_3}L_{\tilde{f}}h_1(\mathbf{x}) = \frac{-\Omega_S^r}{2\tilde{H}_T}k_{P_3}x_2, \tag{6.18}$$

and

$$L_{\tilde{f}}h_2(\mathbf{x}) = (\frac{R_a}{T_{d0}'X_d'} - \Omega_S^r(x_2-1))U\cot(x_1)$$

$$L_{g_2}h_2(\mathbf{x}) = -\frac{R_a}{T_{d0}'X_d'\sin(x_1)}$$

$$L_{p_1}h_2(\mathbf{x}) = \frac{R_a X_d}{T_{d0}'X_d'\sin(x_1)}, \tag{6.19}$$

the external dynamics will be:

$$\dot{\mathbf{z}} = \begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \\ \dot{z}_3 \end{bmatrix} = A_z\mathbf{z} + B_z\tilde{\mathbf{v}} \tag{6.20}$$

where $\tilde{\mathbf{v}} = [\tilde{v}_1, \tilde{v}_2]^T$ is an auxiliary stabilizing input control vector, and A_z and B_z are as in (5.49). Finally, a control law:

$$\tilde{\mathbf{v}} = -\tilde{\mathbf{K}}_{\mathbf{p}}\mathbf{z}, \tag{6.21}$$

that stabilizes the system in (6.20) can be found by pole placement, as was done before.

6.4.1 Simulation Results

The model in (6.13) was simulated with the same parameters of the Diesel engine and synchronous machine given in Table 5.2 and Table 5.1, respectively. The disturbance stator current i_d was assumed as in (5.51). The time delay of the torque produced by the Diesel engine was assumed 0.01 s, as before. Θ_E was modelled as uncertain parameter, as in the previous chapter, with nominal value $\Theta_{E_0} = 0.4$, and variation $\delta_{\Theta_E} = 0.2\sin(2\pi t)$. The moment of inertia of the propeller and the added mass was modelled as in (6.6), with I_{P_0} and δ_{I_P} as in Table 6.1, which also contains the remaining parameters of the assumed propeller based on the data used in [10]. Thus, the p.u. inertia constant $\tilde{H}_T$ is obtained from (6.3) to be:

$$\tilde{H}_T = \tilde{H}_{T_0} + \delta_{\tilde{H}_T} = 2.55 + 0.01\sin(10\pi t). \tag{6.22}$$

Without loss of generality, a constant value of 5 $\mathtt{m/s}$ was assigned to V_a over the simulation horizon. In lieu of the sudden step in the load torque, we introduced a step in the pitch angle θ to see how the controller would respond to such a step. The pitch angle θ was assigned to a value of 0.3 over the simulation horizon, except for the period from $t = 5$ $\mathtt{s}$ till $t = 7$ $\mathtt{s}$, where the angle was increased to 0.8, and it was fed forward to the controller. In order to examine the robustness of the proposed controller to the uncertain parameters Θ_E and $\tilde{H}_T$, the control laws in (6.17) were calculated at the nominal value Θ_{E_0} and $\tilde{H}_{T_0}$. The same gains $\mathbf{K_3}$ and $\mathbf{K_4}$ in (5.55) were used here, and the states trajectories are shown in Fig. 5.5, with the control inputs signals.

I_{P_0} $\mathtt{kg\,m^2}$	δ_{I_P} $\mathtt{kg\,m^2}$	R_P	Q_0 $\mathtt{N\,m\,s^2/rad^2}$	$Q_{|n|n}$ $\mathtt{N\,m\,s^2/rad^2}$	$Q_{|n|V_a}$ $\mathtt{N\,m\,s^2/rad\,m}$
1000	$250\sin(10\pi t)$	0.2	1.22×10^4	5.375×10^3	-4.125×10^3

Table 6.1: The Propeller Parameters

One can note from Fig. 6.2 that the controller behaves as was shown in the previous chapter. The terminal voltage is not affected by the change of the load torque resulting from the change of the pitch angle. Despite the uncertain parameters and the time delay, the angle and speed are kept stable by the proposed controller, but with a limit cycle, as expected. The reference points required to be tracked are not changes, except for the angle where the set point reached by the controller is shifted a bit, especially when the pitch angle is changed. We can also see that both control inputs are acting to keep the system stable. As was explained before, increasing the gain on the angle and speed states reduces the magnitude of the limit cycle and decreases the dependence on the second control input E_F, on one hand. On the other hand, increasing the gain leads to undesired oscillations in the angle and speed when the load changes suddenly due to the time delay. Thus, fine tuning is required to choose the best gain.

6.5 Conclusion

In this chapter, the shaft generators known as PTI/PTO, that comprise a Diesel engine connected to a synchronous machine and the main propeller, are considered. The model presented in the previous chapter was extended to include the propeller to describe one mode of operation, namely the PTO in which the engine drives both the machine and propeller simultaneously, at constant speed. A controller was designed by using feedback linearisation to regulate the shaft speed and the terminal voltage. Simulations of the proposed model and controller were presented. In these simulations, the added mass around the propeller was modelled as an uncertain parameter. The results showed that the proposed controller could stabilize the system with very small limit cycles due to the time delay of the engine torque and the uncertainties imposed by the added mass and the air/fuel ratio.

We believe that the presented model can be extended for other modes of operation, and hence a supervisory controller can be thought of to choose the most appropri-
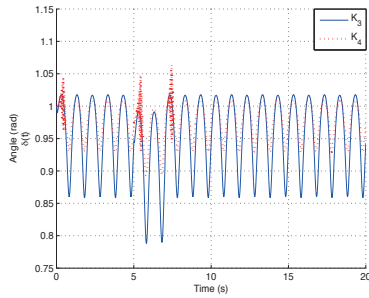
(a) Angle $x_1(t) = \delta$
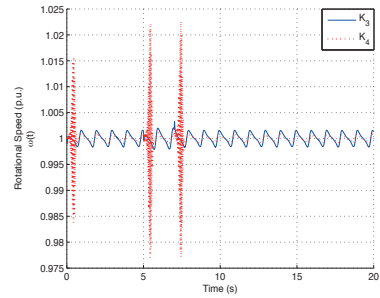
(b) Rotational speed $x_2(t) = \omega$
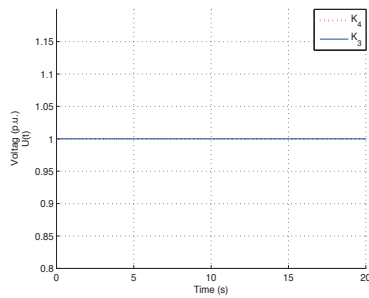
(c) Terminal Voltage $x_3(t) = U$
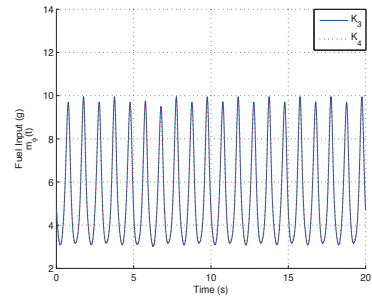
(d) Fuel mass input $u_1(t) = m_\phi$
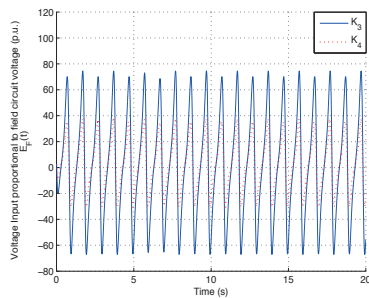
(e) Field circuit input $u_2(t) = E_F$
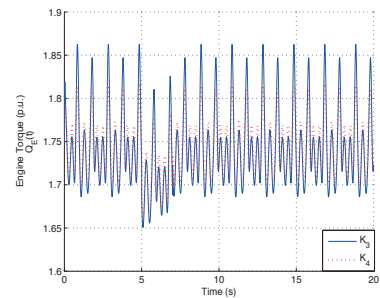
(f) Engine Torque $Q_E(t)$

Figure 6.2: Results of the simulation of the model in (6.13) with a sudden step in the pitch angle by the gains: $\mathbf{K_3}$ (solid) and $\mathbf{K_4}$ (dotted) with Θ_E and $\tilde{H}_t$ modelled as in (5.54) and (6.22), respectively, and with 0.01 $\mathbf{s}$ time delay

ate mode of operation and its controller. Definitely, this can be another project for future work.

Chapter 7

Concluding Remarks and Recommendations for Future Work

Management and control of isolated power systems is a hot area for research, nowadays. Isolated power systems are different from the on-land power systems in many aspects. First, the transmission lines in isolated power systems can usually be neglected. Then, the demand on such systems is highly unpredictable, in opposition to on-land power systems where the expectation of the demand can be accurately determined from the statistical data. Thus, special strategies must be developed to deal with isolated systems whether on the control level or on the management level. In this work, we tried to touch these general topics in a general framework.

Part I

Chapter 2 introduced basic results from the theory of IP and propositional calculus that was used throughout this work.

In **chapter 3**, a novel technique was proposed for minimization over PWL functions based on mixed-logical inequalities. In contrast to the regular SOS method, the proposed method can handle special class of discontinuous functions that are continuous form the left but not lower semi-continuous. Such functions may not have a minimum but they have an infimum. By using the proposed method, the infimum can be approximated to within arbitrarily small positive ϵ. Besides, the proposed model can be easily extended to approximate the supremum of the discontinuous PWL functions that are continuous from the right but not upper semi-continuous, when maximization problems are considered. The formulation of the model was proved to be sharp, but not ideal, and its convex-hull was proved to be full-dimensional. However, the proposed technique tends to increase the computational complexity. Thus, some strong inequalities were found and proved to be strong, i.e. facet-defining or cutting planes. The numerical results show that the proposed inequalities have a strong impact on reducing the computational time.

In **chapter 4**, we proposed a state-space model in discrete time that can be used to solve both the UC and ED problems, simultaneously. The ramping process, the

switching on/off process, and the up/down times change were combined in the proposed dynamic model. Then, the objective function was minimized with the constraints described by the state-space model, in addition to the regular constraints, namely, the power balance, power generation, spinning reserves, and minimum up/down times. Based on the suggested model, a new method to capture the start-up cost was presented, as well. Further, some strong inequalities were suggested and proved to be good cuts for the MIP. Numerical experiments were carried out on different systems comprising up to 100 units over a time horizon of 24 time slots. The objective function was taken to be the fuel consumption and the start-up cost. The proposed model was compared with a commonly used model introduced at the beginning of this chapter. The numerical results were carried out with the quadratic objective function, and with its PWL approximation, and they proved that:

1. the proposed model, without taking the start-up cost in consideration, gives better results in less time for small-sized problems, especially with quadratic objective function.

2. the proposed technique to capture the start-up cost is more realistic, and it enables the solver to reach optimal solutions to within smaller gaps when quadratic objective function is used.

3. with PWL the proposed model behaved slightly worse due to the increasing number of constraints used to describe the PWL function.

4. the effect of the proposed constraints on the commonly used models are stronger than that on the proposed model.

Regarding this chapter, we recommend the following for future work, if possible:

a) A better model of the ramping process can be considered. The ramping dynamics used in this work, was assumed of first order. We believe that this can be improved depending on the type of the generating unit.

b) Throughout this work, we assumed that the demand is known in advance. However, it is well-known that the demand on isolated power systems is usually unpredictable. Thus, should the MPC philosophy be exploited for the scheduling purposes, a deterministic or stochastic model can be thought of for the demand on the system, depending on the purpose of that isolated power system.

c) Depending on the proposed model of the demand, one would consider stochastic or robust optimization techniques to cope with the IP.

Part II

The Genset is controlled by the AVR and the speed governor, which are low-level PID controllers. The AVR responds to any change in the terminal voltage by increasing the field excitation circuit, while the governor responds to any change in the shaft speed by increasing the fuel mass input. The two controllers act separately. In **chapter5** a control-oriented model of the Genset, that comprises a Diesel engine and a synchronous generator, was proposed. The air/fuel ratio was modelled as an uncertain parameter of measurable nominal value, and unmeasurable variation. A

controller to regulate the shaft speed and the terminal voltage, simultaneously, was designed by using feedback linearisation. The simulations of the proposed model with the proposed controller were provided to show that the two control inputs can be manipulated simultaneously to respond to any change in the terminal voltage, the shaft speed or both. In addition, the simulations showed that the time delay imposed by the Diesel engine, and the unmeasurable uncertainty do not lead to loss of stability, as long as they are small, which is the case for the Diesel engine. The uncertainty and time delay can lead to limit cycles, but the set points to be tracked are almost not affected, when suitable gains of the controller are chosen. However, choosing the right values of the gains was not an easy task, because it is a trade-off between reducing the fuel consumption and avoiding high values of field circuit voltage. For future work, we recommend:

1. determining the gains in (5.50) by, instead of pole placement, more robust methods such as H_∞, H_2, mixed H_2/H_∞, or μ synthesis.

2. obtaining more accurate parameters for the indicated efficiency function in (5.23) by proper means of system identification.

3. trying the proposed controller on a more accurate model of the Genset. For example, one would think of a *discrete event model* of the Diesel engine.

Finally, in **chapter 6** the shaft generator known as PTI/PTO, that comprises a Diesel engine connected to a synchronous machine and the main propeller, was discussed. This hybrid system has been used in the industry for more than a decade. Such system can be operated in different modes. Based on the model presented in **chapter 5**, we proposed a method to describe one mode of operation, the PTO in which the engine drives both the machine and propeller simultaneously, at constant speed. Although one would suggest to design an observer for the propeller torque and use it as feed forward signal to the controller in **chapter 5**, we proposed modelling the propeller torque as a function of the speed and the pitch angle that can be obtained from the thrust controllers. The added mass around the propeller due to the hydraulic forces was modelled as another uncertain parameter of known nominal value and unknown variation. The results showed that the proposed controller could stabilize the system with very small limit cycles due to the time delay of the engine torque and the uncertainties imposed by the added mass and the air/fuel ratio. If opportunities are offered in the future, one would think of:

1. extending the current model to describe the other modes of operation.

2. proposing a supervisory controller to choose the most appropriate mode of operation, and ensure quiet transition from one controller to another.

Appendices

Appendix A

Proofs of Propositions in chapter 3

A.1 Proof of Proposition 3.2

Proof. The original variable of the set P^A is x. Let the feasible set of x be $S = [0, a_L]$. Actually, sharpness is guaranteed since, in the model P^A, the space of the original set was lifted up by introducing the new variables β_l and z_l. However, let us try the opposite. Specifically, let us try to obtain the projection of P^A_{LP} on the convex-hull of S, which is S itself in this case. To do that, we use the **Fourier-Motzkin elimination**. From the first constraint in (3.18), one can obtain:

$$\beta_l \leq \frac{a_L - x}{a_L - a_l}, \ \forall l \in \{1, ..., L-1\}. \tag{A.1}$$

Similarly, from the second constrain in (3.18), we obtain:

$$\beta_l \geq \frac{a_l - x + \epsilon}{a_l + \epsilon}, \ \forall l \in \{1, ..., L-1\}. \tag{A.2}$$

From the above two inequalities, we get:

$$\frac{a_l - x + \epsilon}{a_l + \epsilon} \leq \frac{a_L - x}{a_L - a_l}, \tag{A.3}$$

which simplifies to:

$$x(2a_l - a_L + \epsilon) \leq a_l(a_l + \epsilon). \tag{A.4}$$

Now, we have four cases that can be summarized in the equation below:

$$x \begin{cases} \leq \frac{a_l(a_l+\epsilon)}{2a_l-a_L+\epsilon} & \text{if} & 2a_l - a_L + \epsilon > 0 \\ \geq \frac{a_l(a_l+\epsilon)}{2a_l-a_L+\epsilon} & \text{if} & 2a_l - a_L + \epsilon < 0 \\ < \infty & \text{if} & 2a_l - a_L + \epsilon \longrightarrow 0^+ \\ > -\infty & \text{if} & 2a_l - a_L + \epsilon \longrightarrow 0^- \end{cases}. \tag{A.5}$$

1. In the case $2a_l - a_L + \epsilon \longrightarrow 0^+$, $x < \infty$ which is a redundant inequality because $x \leq a_L$.

2. In the case $2a_l - a_L + \epsilon \longrightarrow 0^-$, $x > -\infty$ which is a redundant inequality because $x \geq 0$.

3. In the case $2a_l - a_L + \epsilon < 0$, $x \geq \frac{a_l(a_l+\epsilon)}{2a_l-a_L+\epsilon}$ is also redundant because $x \geq 0$ and $\frac{a_l(a_l+\epsilon)}{2a_l-a_L+\epsilon} < 0$.

4. In the case $2a_l - a_L + \epsilon > 0$, $x \leq \frac{a_l(a_l+\epsilon)}{2a_l-a_L+\epsilon}$ is also redundant, as we show in the sequel. For this inequality to be redundant we need

$$\frac{a_l(a_l+\epsilon)}{2a_l - a_L + \epsilon} \geq a_L,$$

which simplifies, after some algebraic manipulations and by the positivity of $2a_l - a_L + \epsilon$, to:

$$(a_L - a_l) \geq \epsilon,$$

which is always true, because one can always find a positive and small enough ϵ that satisfies the inequality above. Hence, the inequality $x \leq \frac{a_l(a_l+\epsilon)}{2a_l-a_L+\epsilon}$ is redundant.

Similarly, the inequalities in (3.22) can be projected as follows. The LHS of (3.22a), and the RHS of (3.22b) are obviously redundant. Inserting the bounds in (A.1) and (A.2) in the RHS of (3.22a) and LHS of (3.22b), respectively, one can obtain after some manipulation:

$$x \left(\frac{a_l + \epsilon - a_L}{a_l + \epsilon} \right) \leq z_l \leq \frac{a_l}{a_L - a_l}(a_L - x). \tag{A.6}$$

Hence, x can be bounded as:

$$x \left[(a_L - a_l)\epsilon + a_l(a_l + \epsilon) - (a_L - a_l)^2 \right] \leq a_l(a_l + \epsilon)a_L$$
$$\Rightarrow x \left[a_L\epsilon - a_l\epsilon + a_l^2 + a_l\epsilon - a_L^2 + 2a_L a_l - a_l^2 \right] \leq a_l(a_l + \epsilon)a_L$$
$$\Rightarrow xa_L(2a_l - a_L + \epsilon) \leq a_l(a_l + \epsilon)a_L, \tag{A.7}$$

which is as same as the bound in (A.4) after dividing by a_L. The same argument of the resulting four cases presented above proves the result. So, the projection of P_{LP}^A is the original bounded set S. $\qquad \square$

A.2 Proof of Proposition 3.3

Proof. From Definition 2.3, to prove that $conv(P^A)$ is full-dimensional, we need to list down $2L$ affinely independent points in it. Consider the $2L$ points listed in Table A.1 $p_0, ..., p_L, p'_1, ..., p'_{L-1}$. It can be easily verified that all of the points given

are in $conv(P^A)$, where each point p'_l is given by:

$$p'_l = \begin{pmatrix} x \\ z_1 \\ \vdots \\ z_l \\ z_{l+1} \\ \vdots \\ z_{L-1} \\ \beta_1 \\ \vdots \\ \beta_l \\ \beta_{l+1} \\ \vdots \\ \beta_{L-1} \end{pmatrix} = \begin{pmatrix} a_l + \epsilon \\ 0 \\ \vdots \\ 0 \\ a_l + \epsilon \\ \vdots \\ a_l + \epsilon \\ 0 \\ \vdots \\ 0 \\ 1 \\ \vdots \\ 1 \end{pmatrix} \qquad (A.8)$$

Obviously, the points are affinly independent because the directions $p_1 - p_0, ..., p'_{L-1} - p_0$ are linearly independent, and the result follows. $\qquad \square$

	x	z_1	z_2	$\cdots$	$\cdots$	z_{L-1}	β_1	β_2	$\cdots$	$\cdots$	β_{L-1}
p_0	0	0	$\cdots$	$\cdots$	$\cdots$	0	1	1	$\cdots$	$\cdots$	1
p_1	a_1	a_1	$\cdots$	$\cdots$	$\cdots$	a_1	1	1	$\cdots$	$\cdots$	1
p_2	a_2	0	a_2	$\cdots$	$\cdots$	a_2	0	1	$\cdots$	$\cdots$	1
$\vdots$	$\vdots$	$\vdots$		$\ddots$		$\vdots$	$\vdots$		$\ddots$	$\vdots$	$\vdots$
p_{L-1}	a_{L-1}	0	$\cdots$	$\cdots$	0	a_{L-1}	0	$\cdots$	$\cdots$	0	1
p_L	a_L	0	$\cdots$	$\cdots$	$\cdots$	0	0	$\cdots$	$\cdots$	$\cdots$	0
p'_1	$a_1 + \epsilon$	0	$a_1 + \epsilon$	$\cdots$	$\cdots$	$a_1 + \epsilon$	0	1	$\cdots$	$\cdots$	1
p'_2	$a_2 + \epsilon$	0	0	$a_2 + \epsilon$	$\cdots$	$a_2 + \epsilon$	0	0	1	$\cdots$	1
$\vdots$	$\vdots$	$\vdots$		$\ddots$			$\vdots$		$\ddots$		$\vdots$
p'_{L-1}	$a_{L-1} + \epsilon$	0	0	$\cdots$	$\cdots$	0	0	0	$\cdots$	$\cdots$	0

Table A.1: List of feasible points in $conv(P^A)$.

A.3 Proof of Proposition 3.6

Proof. From Definition 2.5 we need to find $2L - 1$ affinely independent points in $conv(P^A)$ that satisfy the inequality at equality. Let us start with the first set of inequalities in (3.31) given by

$$z_{l+1} - z_l \leq (\beta_{l+1} - \beta_l)a_{l+1} \qquad \qquad \forall l \in \{1, ..., L - 2\}. \qquad (A.9)$$

We have $L - 2$ inequalities. It is enough to prove that one of them is facet-defining since for the other inequalities the same technique can be used to generate the $2L - 1$ affinely independent points. Let us take for example the first one:

$$z_2 - z_1 \leq (\beta_2 - \beta_1)a_2. \tag{A.10}$$

The points $\{p_0, ..., p'_{L-1}\}$ in Table A.1, except the point p'_1, satisfy the above inequality at equality. These are $2L - 1$ affinely independent points in $conv(P^A)$ that satisfy (A.10) tightly. The result follows from definition.

Now, consider the inequality:

$$z_{l+1} - z_l \geq (\beta_{l+1} - \beta_l)(a_l + \epsilon) \qquad \forall l \in \{1, ..., L - 2\}. \tag{A.11}$$

One more time, it is enough to prove that the inequality:

$$z_2 - z_1 \geq (\beta_2 - \beta_1)(a_1 + \epsilon), \tag{A.12}$$

is facet-defining. The points $\{p_0, ..., p'_{L-1}\}$ in Table A.1, except the point p_2, satisfy the above inequality at equality. These are $2L - 1$ affinely independent points in $conv(P^A)$ that satisfy (A.12) tightly. The result follows from definition. $\square$

Appendix B

Proofs of Propositions in chapter 4

B.1   Proof of Proposition 4.2

Proof. As was done in the proof of Proposition 4.1, we investigate the four possible cases for the two successive binary indicators $\alpha_j(k)$ and $\alpha_j(k+1)$ in Table 4.1. We want to prove that for each case the two formulations will result in the same constraints.

1. When $\alpha_j(k) = \alpha_j(k+1) = 1$. The unit stays on during the successive time slots, implying that $\beta_j(k)$ and $\gamma_j(k)$ are zero. Hence, the model in (4.27) will boil down to $p_j(k+1) = p_j(k) + \Delta_T r_j(k)$. From (4.28), $-RD_j \leq r_j(k) \leq RU_j$, which can be rewritten by the substitution $\Delta_T r_j(k) = p_j(k+1) - p_j(k)$, as:

$$-\Delta_T RD_j \leq p_j(k+1) - p_j(k) \leq \Delta_T RU_j,$$

which is exactly the same constraint in (4.17) after inserting $\beta_j(k) = \gamma_j(k) = 0$ and $\alpha_j(k+1) = \alpha_j(k) = 1$.

2. When $\alpha_j(k) = \alpha_j(k+1) = 0$. The unit stays off during the successive time slots, implying that $\beta_j(k)$ and $\gamma_j(k)$ are zero. Hence, the model in (4.27) will boil down to $p_j(k+1) = p_j(k) + \Delta_T r_j(k)$. From (4.28), $r_j(k) = 0$, which can be rewritten by the previous substitution as:

$$p_j(k+1) - p_j(k) = 0,$$

which is exactly the same constraint in (4.17) after inserting $\beta_j(k) = \gamma_j(k) = 0$ and $\alpha_j(k+1) = \alpha_j(k) = 0$.

3. When $\alpha_j(k) = 0$ and $\alpha_j(k+1) = 1$. The unit is off and is to be turned on in the $(k+1)$th slot implying that $\beta_j(k) = 1$ and $\gamma_j(k) = 0$. Recall that $p_j(k) = 0$ because $\alpha_j(k) = 0$. Hence, the model in (4.27) will boil down to $p_j(k+1) = \Delta_T R_j^{SU}$, because from the first inequality in (4.28) $r_j(k) = 0$. Now, insert $\beta_j(k) = 1$, $\gamma_j(k) = 0$, $\alpha_j(k+1) = 1$ and $\alpha_j(k) = p_j(k) = 0$ in (4.17), we get:

$$-\Delta_T RD_j \leq p_j(k+1) \leq \Delta_T R_j^{SU}.$$

Then, from the constraint in (4.6) we have $\underline{P}_j \leq p_j(k+1) \leq \overline{P}_j$. Thus, the only feasible solution to the last two constraints is $p_j(k+1) = \Delta_T R_j^{SU}$ if $\Delta_T R_j^{SU} = \underline{P}_j$. Note that if $\Delta_T R_j^{SU} < \underline{P}_j$, no feasible solution exists.

4. When $\alpha_j(k) = 1$ and $\alpha_j(k+1) = 0$. The unit is on and is to be turned off in the $(k+1)$th slot implying that $\beta_j(k) = 0$ and $\gamma_j(k) = 1$. Recall that $p_j(k+1) = 0$ because $\alpha_j(k+1) = 0$. Hence, the model in (4.27) will boil down to $p_j(k) = \Delta_T R_j^{SD}$, because from the second inequality in (4.28) $r_j(k) = 0$. Now, insert $\beta_j(k) = 0$, $\gamma_j(k) = 1$, $\alpha_j(k+1) = p_j(k+1) = 0$ and $\alpha_j(k) = 1$ in (4.17), we get:

$$-\Delta_T RU_j \le p_j(k) \le \Delta_T R_j^{SD}.$$

Then, from the constraint in (4.6) we have $\underline{P}_j \le p_j(k) \le \overline{P}_j$. Thus, the only feasible solution for the last two constraints is $p_j(k) = \Delta_T R_j^{SD}$ if $\Delta_T R_j^{SD} = \underline{P}_j$. Note also that if $\Delta_T R_j^{SD} < \underline{P}_j$, no feasible solution exists.

$\square$

B.2 Proof of Proposition 4.3

Proof. Let us begin by showing that the start-up cost will be zero when the unit is not scheduled to be started up. Actually, if $\beta_j(k) = 0$, $r_j^{off}(k) = 0$ from (4.38) which is less than CT_j, so $z_j^{CT}(k) = 0$. Then, from (4.52) one gets:

$$\begin{aligned} c_j^{SU}(k) &\ge 0 \\ c_j^{SU}(k) &\ge -cc_j \\ c_j^{SU}(k) &\ge 0 \\ c_j^{SU}(k) &\le 0, \end{aligned}$$

so the start-up cost will be zero. On the other hand, if $\beta_j(k) = 1$, we have three cases for $r_j^{off}(k)$:

1. When $r_j^{off}(k) < HT_j$.
   In this case $z_j^{CT}(k) = 0$ because of the constraints in (4.51), so the constraints in (4.52) will look like:

   $$\begin{aligned} c_j^{SU}(k) &\ge hc_j \\ c_j^{SU}(k) &\ge 0 \\ c_j^{SU}(k) &\ge \left[r_j^{off}(k) - HT_j\right] m_j^{SU} + hc_j \\ c_j^{SU}(k) &\le cc_j. \end{aligned}$$

   Since $r_j^{off}(k) < HT_j$, the third inequality above will result in a value less than hc_j, definitely. Thus, the minimum feasible solution of the above constraints is hc_j.

2. When $HT_j \le r_j^{off}(k) \le CT_j$:
   In this case $z_j^{CT}(k) = 0$ also because of the constraints in (4.51), so the

constraints in (4.52) will be:

$$c_j^{SU}(k) \geq hc_j$$
$$c_j^{SU}(k) \geq 0$$
$$c_j^{SU}(k) \geq \left[r_j^{off}(k) - HT_j \right] m_j^{SU} + hc_j$$
$$c_j^{SU}(k) \leq cc_j.$$

Since $r_j^{off}(k) > HT_j$, the minimum feasible solution of the above constraints is $\left[r_j^{off}(k) - HT_j \right] m_j^{SU} + hc_j$.

3. When $r_j^{off}(k) > CT_j$:
   In this case $z_j^{CT}(k) = 1$ because of the constraints in (4.51), so the constraints in (4.52) will write:

$$c_j^{SU}(k) \geq hc_j$$
$$c_j^{SU}(k) \geq cc_j$$
$$c_j^{SU}(k) \geq \left[r_j^{off}(k) - HT_j - cK \right] m_j^{SU} + hc_j$$
$$c_j^{SU}(k) \leq cc_j$$
$$\forall j \in \mathcal{J}, \quad \forall k \in \{0, ..., K-1\}. \tag{B.1}$$

Since $r_j^{off}(k) < cK$ as assumed in (4.38), the third inequality above will result in a value less than hc_j, definitely. Thus, the minimum feasible solution of the above constraints is cc_j.

$\square$

References

[1] E. Alfieri, A. Amstutz, and L. Guzzella. Gain-scheduled model-based feedback control of the air/fuel ratio in diesel engines. *Control Engineering Practic*, (17):1417–1425, 2009.

[2] J. M. Arroyo and A. J. Conejo. Optimal response of a thermal unit to an electricity spot market. *IEEE Transactions on Power Systems*, 15(3):1098–1104, 2000.

[3] J. M. Arroyo and A. J. Conejo. Modeling of start-up and shut-down power trajectories of thermal units. *IEEE Transaction on Power Systems*, 19(3):1562–1568, 2004.

[4] E. Balas. Projection, lifting and extended formulation in integer and combinatorial optimization. *Annals of Operations Research*, 140:125–161.

[5] E. Balas and W. R. Pulleyblank. The perfectly matchable subgraph polytope of an arbitrary graph. *Combinatorica*, 9:321–337.

[6] E. M. L. Beale and J. Tomlin. Special facilities for nonconvex problems using ordered sets of variables. *Proceedings of the 5th International Conference on Operational Research, (J. Lawrence ed., Tavistock Publications)*, 69:447–454, 1970.

[7] A. Bemporad and M. Morari. Control of systems integrating logic, dynamics, and constraints. *Automatica*, 35:407–427, 1999.

[8] L. T. Biegler and I. E. Grossmann. Part ii: Future perspective on optimization. *Computers and Chemical Engineering*, 28:1193–1218, 2004.

[9] L. T. Biegler and I. E. Grossmann. Retrospective on optimization. *Computers and Chemical Engineering*, 28:1169–1192, 2004.

[10] M. Blanke. *Ship Propulsion Losses Related to Automatic Steering and Prime Mover Control*. PhD thesis, Technical University of Denmark, 1981.

[11] M. Blanke, K. P. Lindegaard, and T. I. Fossen. Dynamic model for thrust generation of marine propellers. *IFAC Conference of Maneuvering and Control of Marine craft (MCMC)*, pages 363–368, 2000.

[12] P. Bonami, M. Kilinç, and J. Linderoth. Algorithms and software for convex mixed integer nonlinear programs. In J. Lee and S. Leyffer, editors, *Mixed Integer Nonlinear Programming, The IMA Volumes in Mathematics and its Applications*, volume 154, pages 1–39. Springer, 2012.

[13] M. Carrión and J. M. Arroyo. A computationally efficient mixed-integer linear formulation for the thermal unit commitment problem. *IEEE Transactions on Power Systems*, 21(3):1371–1378, 2006.

[14] G. M. Casolino, A. Losi, and M. Russo. Design choices for combined cycle units and profit-based unit commitment. *Electrical Power and Energy Systems*, 42:693–700, 2012.

[15] K. Chandrasekaran, S. Hemamalini, S. P. Simon, and N. P. Padhy. Thermal unit commitment using binary/real coded artificial bee colony algorithm. *Electric Power Systems Research*, 84:109–119, 2012.

[16] A. I. Cohen and V. R. Sherkat. Optimization-based methods for operations scheduling. *Proceedings of the IEEE*, 75:1574–1591, 1987.

[17] G. C. Contaxis and J. Kabouris. Short term scheduling in a wind/diesel autonomous energy system. *Transactions on Power Systems*, 6(3):1161–1167, Aug 1991.

[18] K. L. Croxton, B. Gendron, and T. L. Magnanti. A comparison of mixed-integer programming models for non-convex piecewise linear cost minimization problems. *Management Science*, 49(9):1268–1273, 2003.

[19] G. B. Dantzig. On the significance of solving linear programming problems with some integer variables. *Econometrica*, 28(1):30–44, 1960.

[20] D. Dasgupta and D.R. McGregor. Thermal unit commitment using genetic algorithms. *IEE Proceedings, Generation, Transmission, Distribution*, 141(5):459–465, Sep 1994.

[21] I. R. de Farias Jr., M. Zhao, and H. Zhao. A special ordered set approach for optimizing a discontinuous separable piecewise linear function. *Operations Research Letters*, 36:234–238, 2008.

[22] H. W. Dommel and W. F. Tinney. Optimal power flow solutions. *IEEE Transactions on Power Apparatus and Systems*, PAS-87(10):1866–1876, Oct 1968.

[23] X. Fang, S. Misra, G. Xue, and D. Yang. Smart grid-the new and improved power grid: A survey. *IEEE Communications Surveys & Tutorials*, 14(4):944–980, 2012.

[24] F. A. C. C. Fontes, D. B. M. M. Fontes, and L. A. Roque. An optimal control approach to the unit commitment problem. In *IEEE 51st Conference on Decision and Control (CDC)*, pages 7069–7074, 2012.

[25] T. I. Fossen. *Guidance and Control of Ocean Vehicles*. John Wiley & Sons, 1994.

[26] A. Frangioni and C. Gentile. Perspective cuts for a class of convex 0-1 mixed integer programs. *Mathematical Programming*, 106:225–236, 2006.

[27] C. L. Frenzen, T. Sasao, and J. T. Butler. On the number of segments needed in a piecewise linear approximation. *Journal of Computational and Applied Mathematics*, 234:437–446, 2010.

[28] J. Garcia-González and J. Barquin. Self-unit commitment of thermal units in a competitive electricity market. *IEEE Power Engineering Society Summer Meeting*, 4:2278–2283, 2000.

[29] L. L. Garver. Power generation scheduling by integer programming-development of theory. *AIEE Transactions on Power Apparatus and Systems*, 81:730–735, Feb 1963.

[30] M. Gerdts. A variable time transformation method for mixed-integer optimal control problems. *Optimal Control Applications and Methods*, 27:169–182, 2006.

[31] K. B. Goh, S. K. Spurgeon, and N. B. Jones. Higher-order sliding mode control of a diesel generator set. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, 217:229–241, 2003.

[32] H. B. Gooi, D. P. Mendes, K. R. W. Bell, and D. S. Kirschen. Optimal scheduling of spinning reserve. *IEEE Transactions on Power Systems*, 14(4):1485–1492, Nov 1999.

[33] X. Guan, P. B. Luh, and H. Yan. An optimization-based method for unit commitment. *Electric Power and Energy Systems*, 14(1):9–17, 1992.

[34] Y. Gui, C. H. Kim, C. C. Chung, and Y. C. Kang. Intra-day unit commitment for wind farm using model predictive control method. *IEEE Power and Energy Society General Meeting*, pages 1–5, 2013.

[35] L. Guzzella and A. Amstutz. Control of diesel engines. *IEEE Control Systems*, 18:53–71, 1998.

[36] L. Guzzella and C. H. Onder. *Introduction to Modeling and Control of Internal Combustion Engine Systems*. Springer, 2nd edition, 2010.

[37] X. S. Han, H. B. Gooi, and D. S. Kirschen. Dynamic economic dispatch: Feasible and optimal solutions. *IEEE Transactions on Power Systems*, 16(1):22–28, Feb 2001.

[38] X.S. Han and H.B. Gooi. Effective economic dispatch model and algorithm. *Electrical Power and Energy Systems*, 29:113–120, 2007.

[39] J. F. Hansen. *Modelling and Control of Marine Power Systems*. PhD thesis, Norwegian University of Science and Technology, 2000.

[40] J. F. Hansen, A. K. Ådnanes, and T. I. Fossen. Mathematical modeling of diesel-electric propulsion systems for marine vessels. *Mathematical and Computer Modeling of Dynamical Systems*, 7(1):1–33, 2001.

[41] J. B. Heywood. *Internal Combustion Engine Fundamentals*. McGraw-Hill, 1988.

[42] W. A. Hill, G. Creelman, and L. Mischke. Control strategy for an icebreaker propulsion system. *IEEE Transactions on Industry Applications*, 28(4):887–892, July/Aug 1992.

[43] M. L. Huang. Robust control research of chaos phenomenon for diesel-generator set on parallel connection. *Applications of Nonlinear Control, Intec.*, 2012.

[44] A. Isidori. *Nonlinear Control Systems*. 2nd edition.

[45] J. P. Jensen, A. F. Kristensen, S. C. Sorenson, N. Houbak, and E. Hendricks. Mean value modeling of a small turbocharged diesel engine. *Society of Automotive Engineers (SAE)*, (910070), 1991.

[46] M. Jünger and G. Reinelt. *Facets of Combinatorial Optimization: Festschrift for Martin Grötschel*. Springer, 2010.

[47] M. Kao and J. J. Moskwa. Engine load and equivalence ratio estimation for control and diagnostics via nonlinear sliding observer. In *Proceedings of the American Control Conference*, pages 1574–1578, June 1994.

[48] Y. L. Karnavas and D. P. Papadopoulos. Maintenance oriented algorithm for economic operation of an autonomous diesel-electric station. *Electric Power Systems Research*, 51:109–122, 1999.

[49] S. A. Kazarlis, A.G. Bakirtzis, and V. Petridis. A genetic algorithm solution to the unit commitment problem. *IEEE Transactions on Power Systems*, 11(1):83–92, 1996.

[50] A. B. Keha, I. R. de Farias Jr., and G. L. Nemhauser. Models for representing piecewise linear cost functions. *Operations Research Letters*, 32:44–48, 2004.

[51] A. B. Keha, I. R. de Farias Jr., and G. L. Nemhauser. A branch-and-cut algorithm without binary variables for nonconvex piecewise linear optimization. *Operations Research*, 54(5):847–858, 2006.

[52] P. P. Khargonekar and M. A. Rotea. Mixed h_2/h_∞ control: A convex optimization approach. *IEEE Transactions on Automatic Control*, 36(7):824–837, July 1991.

[53] S. N. Kolavennu, S. Palanki, and J. C. Cockburn. Robust controller design for multi-variable non-linear systems via multi-model H_2/H_∞ synthesis. *Chemical Engineering Science*, 56:4339–4349, 2001.

[54] S. Kontogiorgis. Practical piecewise-linear approximation for monotropic optimization. *INFORMS Journal on Computing*, 12(4):324–340, 200.

[55] B. Korte and J. Vygen. *Combinatorial Optimization: Theory and Algorithms*. Springer, fifth edition, 2010.

[56] P. Kundur. *Power System Stability*. McGraw-Hill, Inc., 1994.

[57] H. G. Kwatny and T. E. Bechert. On the structure of optimal area controls in electric power networks. *IEEE Transactions on Automatic Control*, 34:167–172, Apr 1973.

[58] J. Lee, J. Leung, and F. Margot. Min-up/min-down polytopes. *Discrete Optimization*, 1:77–85, 2004.

[59] J. Lee and D. Wilson. Polyhedral methods for piecewise-linear functions i: the lambda method. *Discrete Applied Mathematics*, 108:269–285, 2001.

[60] T. Li and M. Shahidehpour. Price-based unit commitment: A case of lagrangian relaxation versus mixed integer programming. *IEEE Transaction on Power Systems*, 20(4):2015–2025, Nov 2005.

[61] T. Li and M. Shahidehpour. Dynamic ramping in unit commitment. *IEEE Transactions on Power Systems*, 22(3):1379–1381, Aug 2007.

[62] J. Á. López, R. N. Gómez, and I. G. Moya. Commitment of combined cycle plants using a dual optimization-dynamic programming approach. *IEEE Transaction on Power Systems*, 26(2):728–737, May 2011.

[63] M. Lorenzen, M. Bürger, G. Notarstefano, and F. Allgöwer. A distributed solution to the adjustable robust economic dispatch problem. In *4th IFAC Workshop on Distributed Estimation and Control in Networked Systems*, pages 75–80, Sep 2013.

[64] J. Machowski, J. W. Bialek, and J. R. Bumby. *Power System Dynamics: Stability and Control*. John Wiley & Sons, 2nd edition, 2008.

[65] T. T. Maifeld and G. B. Sheble. Genetic-based unit commitment algorithm. *IEEE Transactions on Power Systems*, 11:1359–1370, Aug 1996.

[66] H. Man-lei. Research on h$_2$ speed governor for diesel engine of marine power station. *Journal of Marine Science and Application*, 6(3):51–57, Sep 2007.

[67] A. H. Mantawy, Y. L. Abdel-Magid, and S. Z. Selim. Unit commitment by tabu search. *IEE Proceedings Generation, Transmission and Distribution*, 145(1), 1998.

[68] H. M. Markowitz and A. S. Manne. On the solution of discrete programming problems. *Econometrica*, 25(1):84–110, 1957.

[69] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert. Constrained model predictive control: Stability and optimality. *Automatica*, 36:789–814, 2000.

[70] D. J. McCowan, D. J. Morrow, and M. McArdle. A digital pid speed controller for a diesel generating set. In *IEEE Power Engineering Society General Meeting*, volume 3, pages 1472–1477, July 2003.

[71] D. J. McGowan, D. J. Morrow, and B. Fox. Integrated governor control for a diesel-generating set. *IEEE Transactions on Energy Conversion*, 21(2):476–483, June 2006.

[72] G. Morales-España, J. M. Latorre, and A. Ramos. Tight and compact milp formulation of start-up and shut-down ramping in unit commitment. *IEEE Transactions on Power Systems*, 28(2):1288–1296, 2013.

[73] J. A. Muckstadt and R. C. Wilson. An application of mixed-integer programming duality to scheduling thermal generating systems. *IEEE Transactions on Power Apparatus and Systems*, PAS-87(12):1968–1978, 1968.

[74] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, second edition, 2006.

[75] M. P. Nowak and W. Römisch. Stochastic lagrangian relaxation applied to power scheduling in a hydro-thermal system under uncertainty. *Annals of Operations Research*, 100:251–272, 2000.

[76] A. Olsbu, P. A. Loeken, and I. E. Grossmann. A mixed-integer programming model for the design and planning of the power systems in oil production platforms. *Engineering Costs and Production Economics*, 14:281–296, 1988.

[77] J. Ostrowski, M. F. Anjos, and A. Vannelli. Tight mixed integer linear programming formulations for the unit commitment problem. *IEEE Transactions on Power Systems*, 27(1):39–46, 2012.

[78] M. Padberg. Approximating separable nonlinear functions via mixed zero-one programs. *Operations Research Letters*, 27:1–5, 2000.

[79] N. P. Padhy. Unit commitment-a bibliographical survey. *IEEE Transactions on Power Systems*, 19(2):1196–1205, 2004.

[80] S. Palanki, J. C. Cockburn, and S. N. Kolavennu. Robust state feedback synthesis for control of non-square multivariable nonlinear systems. *Journal of Process Control*, 13:623–631, 2003.

[81] A. Parisio, E. Rikos, and L. Glielmo. A model predictive control approach to microgrid operation optimization. *EEE Transaction on Control Systems Technology*, 22(5):1813–1827, Sep 2014.

[82] L. Pivano, T. A. Johansen, Ø. N. Smogeli, and T. I. Fossen. Nonlinear thrust controller for marine propellers in four-quadrant operations. In *Proceedings of the American Control Conference*, pages 900–905, 2007.

[83] D. Radan. *Integrated Control of Marine Electrical Power Systems*. PhD thesis, Norwegian University of Science and Technology, 2008.

[84] M. A. Rahman, A. M. Osheiba, T. S. Radwan, and E. S. Abdin. Modelling and controller design of an isolated diesel engine permanent magnet synchronous generator. *IEEE Transactions on Energy Conversion*, 11(2):324–330, June 1996.

[85] D. Rajan and S. Takriti. Minimum up/down polytopes of the unit commitment problem with start-up costs. *IBM Research Report*, June 2005.

[86] J. B. Rawlings. Tutorial overview of model predictive control. *IEEE Control Systems Magazine*, 20:38–52, June 2000.

[87] A. J. Sørensen and Ø. Smogeli. Torque and power control of electrically driven marine propellers. *Control Engineering Practice*, 17(9):1053–1064, 2009.

[88] D. W. Ross and S. Kim. Dynamic economic dispatch of generation. *IEEE Transactions on Power Apparatus and Systems*, PAS-99(6):2060–2068, 1980.

[89] S. Saneifard, N. R. Prasad, and H. A. Smolleck. A fuzzy logic approach to unit commitment. *IEEE Transactions on Power Systems*, 12(2):988–995, May 1997.

[90] T. Senjyu, K. Shimabukuro, K. Uezato, and T. Funabashi. A fast technique for unit commitment problem by extended priority list. *IEEE Transactions on Power Systems*, 18(2):882–888, May 2003.

[91] G. B. Sheble and G. N. Fahd. Unit commitment literature synopsis. *IEEE Transactions on Power Systems*, 9(1):128–135, 1994.

[92] H. D. Sherali. On mixed-integer zero-one representations for separable lower-semicontinuous piecewise-linear functions. *Operations Research Letters*, 28:155–160, 2001.

[93] C. K. Simoglou, P. N. Biskas, and A. G. Bakirtzis. Optimal self-scheduling of a thermal producer in short-term electricity markets by milp. *IEEE Transactions on Power Systems*, 25(4):1965–1977, 2010.

[94] Ø. N. Smogeli. *Control of Marine Propellers from Normal to Extreme Conditions*. PhD thesis, Norwegian University of Science and Technology, 2006.

[95] S. Sridhar, J. Linderoth, and J. Luedtke. Locally ideal formulations for piecewise linear functions with indicator variables. *Operations Research Letters*, 41:627–632, 2013.

[96]   S. Takriti and J. R. Birge. Using integer programming to refine lagrangian-based unit commitment solutions. *IEEE Transactions on Power systems*, 15(1):151–156, Feb 2000.

[97]   S. Takriti, J. R. Birge, and E. Long. A stochastic model for the unit commitment problem. *IEEE Transactions on Power Systems*, 11(3):1497–1508, Aug 1996.

[98]   S. Takriti, B. Krasenbrink, and L. S. Y. Wu. Incorporating fuel constraints and electricity spot prices into the stochastic unit commitment problem. *Operations Research*, 48(2):268–280, 2000.

[99]   M. Tanaka. Real-time pricing with ramping costs: A new approach to managing a steep change in electricity demand. *Energy Policy*, 34:3634–3643, 2006.

[100] T. O. Ting, M. V. C. Rao, and C. K. Loo. A novel approach for unit commitment problem via an effective hybrid particle swarm optimization. *IEEE Transactions on Power Systems*, 21(1):411–418, Feb 2006.

[101] M. Tuffaha and J. T. Gravdahl. Mixed-integer formulation of unit commitment problem for power systems: Focus on start-up cost. In *IEEE 39th Annual Conference of the Industrial Electronics Society (IECON)*, pages 8160–8165, Vienna-Austria, Nov 2013.

[102] M. Tuffaha and J. T. Gravdahl. Mixed integer minimization of the cost function of the unit commitment problem for isolated power systems. In *IEEE 52nd Annual Conference on Decision and Control (CDC)*, pages 421–428, Firenze-Italy, Dec 2013.

[103] M. Tuffaha and J. T. Gravdahl. Modeling and control of a marine diesel engine driving a synchronous machine and a propeller. In *2014 IEEE International conference on Control Applications (CCA) Part of 2014 IEEE Multi-conference on Systems and Control*, pages 897–904, Antibes-France, Oct 2014.

[104] M. Tuffaha and J. T. Gravdahl. Control-oriented model of a generating set comprising a diesel engine and a synchronous generator. *Modelling, Identification and Control*, 36(4):199–214, 2015.

[105] M. Tuffaha and J. T. Gravdahl. Discrete state-space model to solve the unit commitment and economic dispatch problems. Submitted to Energy Systems (ENSY), https://www.dropbox.com/s/f6vb254pv1evzzo/SSModelUC.pdf?dl=0, 2015.

[106] M. Tuffaha and J. T. Gravdahl. Dynamic formulation of the unit commitment and economic dispatch problems. In *IEEE International Conference on Industrial Technology (ICIT)*, pages 1294–1298, Seville-Spain, March 2015.

[107] M. Tuffaha and J. T. Gravdahl. On the optimization of discontinuous piecewise linear functions. Submitted to: Central European Journal of Operations Research (CJOR), https://www.dropbox.com/s/8jbdic0qfm5fxow/OnDiscPWLFunctions.pdf?dl=0, 2015.

[108] J. P. Vielma, S. Ahmed, and G. Nemhauser. Mixed-integer models for non-separable piecewise-linear optimization: Unifying framework and extensions. *Operations Research*, 58(2):303–315, 2010.

[109] J. P. Vielma and G. L. Nemhauser. Modeling disjunctive constraints with a logarithmic number of binary variables and constraints. *Mathematical Programming*, 128:49–72, 2011.

[110] C. Wang and S. M. Shahidehpour. Ramp-rate limits in unit commitment and economic dispatch incorporating rotor fatigue effect. *IEEE Transactions on Power Systems*, 9(3):1539–1545, 1994.

[111] J. Warrington, P. Goulart, S. Mariéthoz, and M. Morari. Policy-based reserves for power systems. *IEEE Transaction on Power Systems*, 28(4):4427–4437, Nov 2013.

[112] L. A. Wolsey. *Integer Programming*. John Wiley & Sons Inc., 1998.

[113] A. J. Wood and B. F. Wollenberg. *Power Generation, Operation, and Control*. Wiley, 2nd edition, 1996.

[114] X. Xia, J. Zhang, and A. Elaiw. An application of model predictive control to the dynamic economic dispatch of power generation. *Control Engineering Practice*, 19:638–648, 2011.

[115] H. Yan, P. B. Luh, X. Guan, and P. M. Rogan. Scheduling of hydrothermal power systems. *IEEE Transactions on Power Systems*, 8(3):1358–1365, 1993.

[116] F. Zhuang and F.D. Galiana. Unit commitment by simulated annealing. *IEEE Transactions on Power systems*, 5(1):311–318, Feb 1990.