الجمهورية الجزائرية الديمقراطية الشعبية

وزارة التعليم العالي والبحث العلمي

People's Democratic Republic of Algeria

Ministry of Higher Education and Scientific Research

University of Saad Dahleb Blida1

Faculty of Science

Department of Mathematics

thesis memory for obtaining a Master's degree in Mathematics

Specialty : Recherche Operationelle

Presented by:

BOUAICHAOUI souad

OTSMANE amira

THEME

A Centralized Home Energy Management System to Minimize Consumer's Electricity Bill

In front of the jury composed of :

Mr.	M.Boukhari	Univ-blida 1	President
Mr.	R.Frihi	Univ-blida 1	Examiner
Mr.	R. Boudjemaa	Univ-blida 1	Supervisor
Ms.	B.Zerrouki	Sonelgaz-blida	Advisor

2023/2024

Acknowledgments

*First of all, I thank **ALLAH** the Almighty for giving me the patience, the will, the energy, and the health to pursue this work.*
*We would like to thank all those who have contributed, directly or indirectly, to the development of this present dissertation, especially our supervisor, **Mr. Redouane Boudjemaa**, for his valuable advice that facilitated our task and his recommendations. To our mentor the entire SONELGAZ team for accepting us into their company, and to the jury members, **Mr. Mohamed Boukhari** and **Mr. Redouane Frihi** .*
We do not forget our parents for their support and patience, our relatives, and our friends who have supported and encouraged us.

Dedication

I want to thank **Allah** who has enabled me to reach the highest ranks and has fulfilled my dream that I had wished for.

I would like to dedicate this humble work to: To my dear **Parents** who supported and encouraged me a lot and who made me who I am today;

And to **myself**, of course, for all the effort and struggle I have made for everything that has passed.

BOUAICHAOUI - SOUAD

Dedication

I would like to dedicate this modest work to:

To my dearest **parents** who have greatly supported and encouraged me, and who have made me who I am today.

To my dear sister **Aya**, I wish you a life full of happiness and prosperity.

To my best friends **Meriem** and **Mouna**, I thank you immensely and wish you a life full of happiness.

OTSMANE - AMIRA

ملخص

الهدف من هذه الأطروحة هو استكشاف نهج رياضي لنمذجة وتحسين استهلاك الطاقةللأجهزة المنزلية. يعتمد هذا النهج على نظام إدارة الطاقة المنزلية المركزي الذي يقوم بجدولة استخدام الأجهزة المنزلية ليتزامن مع انخفاض تعريفة الكهرباء مما يؤدي إلى انخفاض فاتورة الكهرباء. تم تصميم المشكلة المدروسة على أنها مزيج من البرمجة الخطية والبرمجة الخطية المختلطة للأعداد الصحيحة وتم حلها باستخدام البرامج المجانية والتجارية. يتم توفير النتائج العددية للسيناريوهات المحاكاة لإظهار إمكانية تطبيق هذاالنهج. وبعد دراسة هذه المسألة لجأنا إلى استخدام طرق حل مشكلة البرمجةالخطية للأعداد الصحيحة وأخيراً قدمنا برنامجاً يسهل هذا العمل.

Abstract

The aim of this thesis is to explore a mathematical approach to modeling and optimizing the energy consumption of household appliances. The approach is based on a Centralized Home Energy Management System (HEMS) which schedules the use of home appliances to coincide with a lower electricity tariff resulting in a reduced electricity bill. The studied problem is modeled as a combination of Linear Programming (LP) and Mixed Integer Linear Programming (MILP) and solved using both free and commercial softwares. Numerical results for simulated scenarios are provided to showcase the applicability of the approach.
After studying this issue, we resorted to using methods to solve the integer linear programming problem and finally presented a program that facilitates this work.

Keywords:
Home Energy Management System, Mixed Integer Linear Programming, Energy Minimization, Home Appliances Scheduling.

Résumé

L'objectif de cette thèse est d'explorer une approche mathématique pour modéliser et optimiser la consommation énergétique des appareils ménagers. L'approche repose sur un système de gestion centralisée de l'énergie domestique (HEMS) qui planifie l'utilisation des appareils ménagers pour coïncider avec un tarif d'électricité plus bas, entraînant une réduction de la facture d'électricité.
Le problème étudié est modélisé comme une combinaison de la programmation linéaire (PL) et de la programmation linéaire en nombres entiers mixtes et est résolu à l'aide de logiciels gratuits et commerciaux. Des résultats numériques pour des scénarios simulés sont fournis pour démontrer l'applicabilité de l'approche.
Après avoir étudié ce problème, nous avons utilisé des méthodes pour résoudre le problème de programmation linéaire en nombres entiers et enfin présenté un programme qui facilite ce travail.

Keywords:
système de gestion centralisée de l'énergie domestique, programmation linéaire ,programmation linéaire en nombres entiers mixtes .

Contents

List of Figures

List of Tables

Abreviation

LP: Linear Programming.

ILP: Integer Linear Programming.

MILP: Mixed integer linear programming.

HEMS: Home Energy Management System.

MPC: Model Predictive Control.

General introduction

The domestic use of electricity has evolved dramatically over the decades, reflecting technological, economic, and cultural changes. Today, electricity is ubiquitous in our daily lives, and its use extends far beyond lighting and heating. It powers a multitude of devices and systems that enhance our comfort, productivity, and well-being. However, this ubiquity is accompanied by increasing energy consumption, posing significant challenges in terms of sustainability and energy efficiency.

Modern households are equipped with an impressive array of electrical appliances: refrigerators, washing machines, televisions, computers, air conditioning and heating systems, not to mention the countless electronic gadgets that simplify and enrich our lives.

A centralized Home Energy Management System (HEMS) aimed at minimizing consumers electricity bills constitutes a central challenge for a company like the Distribution Center Region, which is part of SONELGAZ, acronym for "National Company for Electricity and Gas", is a company responsible for the production, transportation, and distribution of electricity and gas in Algeria. The company was established on July 28, 1969, replacing the previous entity Electricity and Gas of Algeria (EGA), and was granted a monopoly on the distribution and sale of natural gas in the country, as well as on the production, distribution, import, and export of electricity. Previously organized as an industrial group consisting of 39 subsidiaries and 5 joint venture companies engaged in core businesses, works, and peripherals, the new organization approved in February 2017, established SONELGAZ as an industrial group consisting of 16 subsidiaries. Its core business subsidiaries are responsible for the production, transportation, and distribution of electricity as well as the transportation and distribution of gas.

In order to acquire experience with the proposed problem, which is a centralized system for managing domestic energy to minimize consumers' electricity bills, we completed a six-month internship at the Distribution Center Region at Blida. During this time, we attempted to conduct a research study on the problem and the different state of the art mathematical models.

This company faces several issues, most notably the problem of daily consumption of household appliances. We observe that the average daily consumption has increased from 5.29 kwh in 2006 to 8.43 kwh in 2012. This trend can be attributed to various factors such as population growth, urbanization, and increased use of electrical appliances. We also notice that high-power appliances with long usage periods, such as the electric oven (1800 w) and the electric heater (2700 w), significantly contribute to the total annual consumption in kilowatt-hours (kwh) and Algerian dinars (DA). In particular, the electric heater shows the highest consumption in terms of annual cost (3402 DA) despite being used for only half an hour a day, highlighting its substantial impact on energy consumption. On the other hand, less frequently used appliances, such as the microwave and the hairdryer, despite their high power, have much lower annual energy consumption and costs. The total indicates an annual consumption of 3349.5 kwh, costing 13,398 DA, underscoring the importance of monitoring and managing the use of household appliances to improve energy efficiency and reduce expenses.

In the first chapter, we discuss optimization models such as Linear Programming, Mixed-Integer Linear Programming and others, which provide powerful tools to solve a variety of problems, whether related to maximizing profits, minimizing costs, resource allocation, or scheduling.

In the second chapter, we provide information on the Branch and Bound and Branch and Cut methods,for solving different MIP and MILP optimization problems.

In chapter three, we focus on efficient energy consumption management in the context of the ongoing global energy transition. The critical challenge posed by energy management for consumers and electricity providers in a rapidly changing environment are highlighted.

The importance of improving energy consumption is emphasized, pointing out the benefits for both end-users and electricity providers. The need to find effective solutions for managing energy demand in a smart and sustainable manner is underscored.

Specific strategies, models, or techniques used in the field of energy management are introduced while providing insights into how optimization can contribute to addressing current challenges related to energy consumption and the transition towards more sustainable and efficient energy systems.

In the fourth chapter,a MATLAB numerical implementation of the studied mathematical model is provided, showing by simulation results a reductions in electricity bill and peak power consumption. The algorithm allowed for dynamic scheduling of appliances by solving an optimization problem at each time step, providing flexibility for the user to operate non-thermal appliances at any time with minimal costs. Non-thermal appliances were able to be turned off and rescheduled to periods of lower electricity prices while adhering to the imposed timelines. In the last chapter, we conclude our thesis by providing some suggestions on future work.

Chapter 1

Constrained optimization

1.1 Introduction

We will talk in this chapter about specific constrained optimization problems and everything that revolves around linear programming (LP) applied to solving complex problems. More advanced versions of (LP) through the presence of integer variables are also discussed. Conversely, incorporating integer variables enables the modeling of much more complex and sophisticated systems. This capability is a key reason why integer programming problems, particularly mixed-integer programming problems, are among the most prevalent in practical applications.This chapter is mostely based on chapter (3,5) of oliveira lecture notes [Oliveira].Please refers this lecture notes for farther information and proofs.

1.2 Linear Programming(LP)

Also called linear optimization, is a method to achieve the best outcome (such as maximum profit or lowest cost) in a mathematical model whose requirements and objective are represented by linear relationships. Linear programming is a special case of mathematical programming (also known as mathematical optimization).

More formally, linear programming is a technique for the optimization of a linear objective function, subject to linear equality and linear inequality constraints. Its feasible region is a convex polytope, which is a set defined as the intersection of finitely many half spaces, each of which is defined by a linear inequality. Its objective function is a real-valued affine (linear) function defined on this polytope. A linear programming algorithm finds a point in the polytope where this function has the largest (or smallest) value if such a point exists Wiese [2016]. Linear programs are problems that can be expressed in standard form as

$$\min C^T x \tag{1.1}$$

$$s.t.\ Ax = b \tag{1.2}$$

$$x \in \mathbb{R}^n_+ \tag{1.3}$$

There are other general forms of LPs, for example the canonical one, including inequality constraints instead of equalities $a^i x \leq\geq b^i, i \in [1, n]$.

Here the components of (x) are the variables to be determined, $C \in \mathbb{R}^n$ and $b \in \mathbb{R}^m$ are given vectors, and $A \in \mathbb{R}^{mxn}$ is a given matrix. The function whose value is to be maximized, is called the objective function.

A key point is that *any* linear programming problem can be converted to standard form using nonnegative *slack variables*. For example, a feasibility set initially given by

$$P = \{x \in \mathbb{R}^n : A_1 x \leq b_1, A_2 x \geq b_2, x \geq 0\}$$

can be transformed into an equivalent standard-form polyhedral set. This involves introducing slack variables $s_1 \geq 0$ and $s_2 \geq 0$ so that

$$P = \{(x, s_1, s_2) \in \mathbb{R}^{(n+|b_1|+|b_2|)} : A_1 x + s_1 = b_1, A_2 x - s_2 = b_2, (x, s_1, s_2) \geq 0\},$$

where $|u|$ denotes the cardinality of the vector $\mathbf{u}$.

Theorem 1.1 (Linear independence and basic solution). Consider the constraints $Ax = b$ and $x \geq 0$, with A having m linearly independent (LI) rows $\mathbf{I} = \{1, \ldots, m\}$. A vector $\overline{x} \in \mathbb{R}^n$ is a basic solution if and only if $A\overline{x} = b$ and there are indices $B(1), \ldots, B(m)$ such that

(1) The columns $A_{B(1)}, \ldots, A_{B(m)}$ of A are LI.

(2) For $j \neq B(1), \ldots, B(m)$, we have $\overline{x}_j = 0$.

1.2.1 Forming bases for standard-form linear programming problems

Theorem 1.1 provides us with a way to develop a simple procedure to generate all basic solutions of a linear programming problem in standard form.

1. Choose m LI columns $A_{B(1)}, \ldots, A_{B(m)}$;
2. Let $x_j = 0$ for all $j \notin \{B(1), \ldots, B(m)\}$;
3. Solve the system $Ax = b$ to obtain $x_{B(1)}, \ldots, x_{B(m)}$.

1.2.2 Adjacent basic solutions

Let us begin by formally defining the concept of an adjacent basic solution.
Definition 1.2 (Adjacent basic solutions). Two basic solutions are adjacent if they share $n-1$ LI active constraints. Alternatively, two bases B_1 and B_2 are adjacent if all but one of their columns are the same.

1.2.3 Redundancy and degeneracy

Theorem 1.3 (Redundant constraints). Let $P = \{x \in \mathbb{R}^n : Ax = b, x \geq 0\}$, where A is $m\mathbf{x}n$ matrix with rows $\{a_i\}_{i\in I}$ and $I = \{1, \ldots, m\}$. Suppose that $(A) = k < m$ and that the rows $a_{i_1}, \ldots, a_{i_k}$ are LI. Then P is the same set as $Q = \{x \in \mathbb{R}^n : a_{i_1}^\top x = b_{i_1}, \ldots, a_{i_k}^\top x = b_{i_k}, x \geq 0\}$.

1.3 Optimality of extreme points

1.3.1 The existence of extreme points

First, we define the condition ensuring the existence of extreme points within a polyhedral set. Without this condition, discovering an optimal solution becomes unfeasible.
Definition 1.4 (Existence of extreme points). A polyhedral set $P \subset \mathbb{R}^n$ contains a line if it is non-empty and there exists a nonzero vector $d \in \mathbb{R}^n$ such that $x + \lambda d \in P$ for

all $\lambda \in \mathbb{R}$.

Theorem 1.5 (Existence of extreme points). Let $P = \{x \in \mathbb{R}^n : a_i^\top x \geq b_i, i = 1, \dots, m\} \neq \emptyset$ be a polyhedral set. The following statements are equivalent:

(1) P contains at least one extreme point;

(2) P does not contain a line;

(3) There exist n linearly independent vectors among $\{a_i\}_{i=1}^m$.

Theorem 1.6 (Optimality of extreme points). Let $P = \{x \in \mathbb{R}^n : Ax \geq b\}$ be a polyhedral set and $c \in \mathbb{R}^n$. Consider the problem

$$z = \min\{c^\top x : x \in P\}.$$

Suppose P contains at least one extreme point and there exists an optimal solution. Then, there exists an optimal solution that is also an extreme point of P.

1.3.2 Finding optimal solutions

In practice, optimization methods typically iterate through the following steps:

1. Begin with an initial (often feasible) solution;
2. Locate a neighboring solution with improved value;
3. If none are found, revert to the best-known solution.

Definition 1.7 (Feasible directions). Let $x \in P$, where $P \subset \mathbb{R}^n$ is a polyhedral set. A vector $d \in \mathbb{R}^n$ is a feasible direction at x if there exists $\theta > 0$ for which $x + \theta d \in P$.

1.3.3 Optimality conditions

Having identified promising directions for improvement, we have inadvertently established a framework for assessing the optimality of a given basic feasible solution (BFS)

Theorem 1.8 (Optimality conditions)

Consider the problem $P : \min\{c^\top x : Ax = b, x \geq 0\}$. Let x be the BFS associated with a basis B and let $\bar{c}$ denote the corresponding vector of reduced costs.

(1) If $\overline{c} \geq 0$, then x is optimal.

(2) If x is optimal and nondegenerate, then $\overline{c} \geq 0$.

.

1.4 Mixed integer linear programming (MILP)

MILP is the name for the mathematical problem of minimizing a linear function over a subset of $\mathbb{R}^n$, that can be described by using only linear equalities and requiring a subset of the variables to take only integer values. MILP evolved from Linear Programming (LP), in the mid-sixties over the years, MILP has become a technology and reliable and robust commercial codes for solving MILPs are available.

Mixed integer linear programming (MILP) is the state-of-the-art mathematical framework for optimization of energy systems. The capability of solving rather large problems that include time and space discretization is particularly relevant for planning the transition to a system where non-dispatchable energy sources are key. Here, one of the main challenges is to realistically describe the technologies and the system boundaries: on the one hand the linear modeling, and on the other the number of variables that can be handled by the system call for a trade-off between level of details and computational time Vielma [2015].

An integer programming problem is a mathematical optimization or feasibility problem where some or all of the variables are constrained to be integers. Often, this term specifically refers to integer linear programming (ILP), where both the objective function and the constraints (excluding the integer constraints) are linear. A notable special case is 0–1 integer linear programming, where the variables are binary and only the constraints need to be satisfied. We consider a mixed-integer linear programming (MILP) problem of the general form:

$$\min C^T x \tag{1.4}$$

$$s.t.\ Ax = b \tag{1.5}$$

$$x \in \mathbb{R}_+^{n-p} \cup \mathbb{Z}_+^p \tag{1.6}$$

where $A \in \mathbb{R}^{mxn}, b \in \mathbb{R}^m$ and $C \in \mathbb{R}^n$ In this form, we thus have n variables, corresponding to the columns of the matrix A, and m equality constraints, $a^i x = b^i, i \in [1, n]$,corresponding to its row vectors $a^i, i \in [1, n]$. The feasible region is completed by the fact that the variables have to be non-negative, and that the last p of them have to take integer values.

MILP is thus a non-convex optimization problem. In the spacial case of a purely integer Linear Program (ILP), where $(p = n)$, any feasible solution is thus a completely integer vector. However, also when $(p \leq n)$, we will call a solution to (1.4) - (1.6), i.e., a vector whose last p components only are integer values, an integer solution.

1.5 Duality

In mathematical optimization theory, duality or the duality principle is the principle that optimization problems may be viewed from either of two perspectives, the primal problem or the dual problem. If the primal is a minimization problem then the dual is a maximization problem (and vice versa)., any feasible solution to the primal (minimization) problem is at least as large as any feasible solution to the dual (maximization) problem. Therefore, the solution to the primal is an upper bound to the solution of the dual, and the solution of the dual is a lower bound to the solution of the primal, this fact is called weak duality.

In general, the optimal values of the primal and dual problems need not be equal, their difference is called the duality gap. For convex optimization problems, the duality gap is zero under a constraint qualification condition. This fact is called strong duality Güzelsoy et al. [2010].

Every linear program has associated with it another linear program called the dual. A given linear program and its dual will be related in important ways. For example, a feasible solution for one will provide a bound on the optimal objective function value of the other. Also, if one has an optimal solution, then the other will have an optimal solution as well and the objective function values of both will be the same. In particular, if one problem has an optimal solution, then a "certificate" of optimality can be obtained from the corresponding dual problem verifying the optimality.

The theory related to the relationship between a linear program and its dual is called duality theory, and has important consequences for optimization and is not only of theoretical, but of practical importance as well. This chapter will develop and explore the implications and economic interpretations of duality theory and its role in optimal algorithm design and sensitivity analysis. With the development of duality theory, a variant of the simplex method called the dual simplex method is developed which can enhance the computation of optimal solutions of linear programs that are modifications of an existing problem Klamroth et al. [2004].

1.6 Formulating duals

1.6.1 Motivation

Let us establish the notation to be used in the upcoming chapters. As previously defined, let $c \in \mathbb{R}^n$, $b \in \mathbb{R}^m$, $A \in \mathbb{R}^{mxn}$, and P represent the standard form linear programming problem:

$$\begin{aligned} (P) : \min\ & c^\top x \\ s.t :& Ax = b \\ & x \geq 0, \end{aligned}$$

which we will refer to as the primal problem. In mathematical programming, a constraint is said to be relaxed if it is removed from the set of constraints. Considering this, let us examine a relaxed version of P, where $Ax = b$ is replaced with a violation penalty term $p^\top (b - Ax)$. This leads to the following problem:

$$g(p) = \min_{x \geq 0} \left\{ c^\top x + p^\top (b - Ax) \right\},$$

which eliminates explicit equality constraints and represents them implicitly through a penalty term. This term penalises constraint infeasibility, guiding the relaxed problem's solution towards that of P. Since our primary goal is to solve P, we are interested in determining the values (or prices) $p \in \mathbb{R}^m$ that make P and $g(p)$ equivalent.

Let $\overline{x}$ be the optimal solution to P. Note that for any $p \in \mathbb{R}^m$, we have:

$$g(p) = \min_{x \geq 0} \left\{ c^\top x + p^\top (b - Ax) \right\} \leq c^\top \overline{x} + p^\top (b - A\overline{x}) = c^\top \overline{x},$$

meaning that $g(p)$ serves as a *lower bound* on the optimal value $c^\top \overline{x}$. This inequality is true because, although $\overline{x}$ is optimal for P, it might not be optimal for $g(p)$ for a given p. The equality on the right results from $\overline{x} \in P$, meaning the feasibility of $\overline{x}$ ensures that $A\overline{x} = b$.

Table 1.1 provides a summary which allows one to identify the resulting formulation of the dual problem based on the primal formulation, in particular regarding its type (minimisation or maximisation), constraint types and variable domains. To convert a

Primal (dual)	**Dual (primal)**
minimise	maximise
Independent terms	Obj. function coef.
Obj. function coef.	Independent terms
i-th row of constraint coef.	i-th column of constraint coef.
i-th column of constraint coef.	i-th row of constraint coef.
Constraints	**Variables**
≥ 0	≥ 0
≤ 0	≤ 0
$= 0$	$\in \mathbb{R}$
Variables	**Constraints**
≥ 0	≤ 0
≤ 0	≥ 0
$\in \mathbb{R}$	$= 0$

Table 1.1: Primal-dual conversion table

minimisation primal problem into a maximisation dual, the table should be read from left to right. In this case, the independent terms (b) become the objective function coefficients, and greater or equal constraints turn into nonnegative variables, among other changes. Conversely, if the primal problem is a maximisation problem, the table should be read from right to left. For instance, less-or-equal-than constraints in the primal would become nonnegative variables in the dual, and so forth. With some practice, one can become familiar with this table, which is a valuable tool for deriving dual formulations from primal problems .

It is important to note that the conversion of primal problems into duals is symmetric. This means that by reapplying the rules in Table 1.1, one can revert the obtained dual back to the original primal problem. This property of linear programming problems is known as being *self dual*. Additionally, equivalent reformulations in the primal problem result in equivalent duals. Specifically, transformations such as replacing variables $x \in \mathbb{R}$ with $x^+ - x^-$, where $x^+, x^- \geq 0$, introducing nonnegative slack variables, or removing redundant constraints, all lead to equivalent duals.

1.7 Duality theory

1.7.1 Weak duality

Weak duality pertains to the bounding property of dual feasible solutions, as stated in the Theorem 1.9 .

Theorem 1.9 (Weak duality). Let x be a feasible solution to $(P) : \{c^\top x : Ax = b, x \geq 0\}$ and p be a feasible solution to $(D) : \{p^\top b : p^\top A \leq c^\top\}$, the dual problem of P. Then $c^\top x \geq p^\top b$.

Corollary 1.10 (Consequences of weak duality). The following implications derive directly from Theorem (1.9):

(1) If the optimal value of P is $-\infty$ (i.e., P is unbounded), then D must be infeasible.

(2) If the optimal value of D is ∞ (i.e., D is unbounded), then P must be infeasible.

(3) Let x and p be feasible solutions to P and D, respectively. If $p^\top b = c^\top x$, then x is optimal for P and p is optimal for D. Oliveira

1.7.2 Strong duality

This bounding property can act as a verification of optimality when the values coincide. This idea is referred to as *strong duality*, a fundamental characteristic of linear programming problems, formally stated in Theorem 1.11.

Theorem 1.11 (Strong duality). If $(P) : \min\left\{c^\top x : Ax = b, x \geq 0\right\}$ has an optimal solution, then its dual $(D) : \max\left\{p^\top b : p^\top A \leq c^\top\right\}$ also has an optimal solution, and their optimal values are equal.

1.7.3 Complementary slackness

One point that must be noticed is that, for the constraints that are not active at the optimal point $\overline{x}$. That is, we have that

$$p^\top b = \sum_{i \in I} p_i b_i = \sum_{i \in I} p_i (a_i^\top \overline{x}) = c^\top \overline{x},$$

which again implies the optimality of p (cf. Corollary 1.10 (3)). This geometrical insight leads to another key result for linear programming duality, which is the notion of *complementary slackness*

Theorem 1.12 (Complementary slackness). Let x be a feasible solution for

$$(P) : \{c^\top x : Ax = b, x \geq 0\}$$

and p be a feasible solution for

$$(D) : \{p^\top b : p^\top A \leq c^\top\}.$$

The vectors x and p are optimal solutions to P and D, respectively, if and only if $p_i(a_i^\top x - b_i) = 0, \forall i \in I$, and $(c_j - p^\top A_j)x_j = 0$, $\forall j \in J$.

1.7.4 Dual feasibility and optimality

Combining the preceding discussions, the conditions for a primal-dual pair (x, p) to be optimal for their respective primal (P) and dual (D) problems are given by

$$a_i^\top x \geq b_i, \forall i \in I \qquad \text{(primal feasibility)} \tag{1.7}$$

$$p_i = 0, \forall i \notin I^0 \qquad \text{(complementary conditions)} \tag{1.8}$$

$$\sum_{i \in I} p_i^\top a_i = c \qquad \text{(dual feasibility I)} \tag{1.9}$$

$$p_i \geq 0, \qquad \text{(dual feasibility II)} \tag{1.10}$$

where $I^0 = \{i \in I \mid a_i^\top x = b_i\}$ denotes the set of active constraints.

1.8 Conclusion

In this chapter, we explored linear programming (LP), mixed-integer linear programming (MILP), and Duality with its components Motivation for Duality and Forming the Dual Problem and Weak and Strong Duality Theory, These approaches allowed us to effectively and accurately model various aspects of Constrained optimization.

Chapter 2

Solving constrained optimization problems

2.1 Introduction

This chapter is devoted to the resolutions of constrained optimization problems, especially in the context of Mixed-Integer Linear Programming (MILP), has seen significant advancements with the development of sophisticated algorithms such as branch and bound (separation and evaluation) and branch and cut (separation and cutting). These techniques are particularly effective for addressing complex MILP problems where an exhaustive search of all possible solutions is impractical due to the exponential growth of the search space.This chapter primarily draws from chapters (4, 9, 10, and 11) of Oliveira lecture notes Oliveira. For additional details and proofs, please refer to these lecture notes.

2.2 Calculating step sizes

Consider the linear programming problem P in its standard form

$$(P) : \min\{c^\top x : Ax = b, x \geq 0\}.$$

Using the concepts defined in Chapter 1, solving P with the simplex method involves these steps:

1. Begin with a nondegenerate basic feasible solution (BFS).
2. Identify a negative reduced cost component $\bar{c}_j$. If $\bar{c} \geq 0$, the current solution is optimal.
3. Proceed in the feasible direction $d = (d_B, d_N)$, where $d_j = 1$, $d_{N\setminus\{j\}} = 0$, and $d_B = -B^{-1}A_j$.

2.3 Moving between adjacent bases

Once the optimal step size $\overline{\theta}$ is determined, we transition to a new BFS $\overline{x}$.

Theorem 2.1 (Adjacent bases) Consider A_j as the column of the matrix A related to the chosen nonbasic variable with index $j \in I_N$.where $A_{B(i)}$ is its corresponding column in A. Then:

(1) The columns $A_{B(i)}$ and A_j are linearly independent, implying that $\overline{B}$ is a basic matrix.

(2) The vector $\overline{x} = x + \overline{\theta}d$ represents a BFS corresponding to $\overline{B}$.

Theorem 2.2(Convergence of the simplex method) Assume that P has at least one feasible solution and that all BFS are nondegenerate. Then, the simplex method terminates after finitely many iterations, resulting in one of these outcomes:

(1) The basis B and the associated BFS are optimal; or

(2) There exists a direction d such that $Ad = 0$, $d \geq 0$, and $c^\top d < 0$, leading to an optimal value of $-\infty$.

.

Algorithm 1 Simplex method Oliveira

initialise. Initial basis B, associated BFS x, and reduced costs $\overline{c}$.
while $\overline{c}_j < 0$ for some $j \in I_N$ **do**
  Choose some j for which $\overline{c_j} < 0$. Calculate $u = B^{-1}A_j$.
  if $u \leq 0$ **then**
    return $z = -\infty$.
  else
    $\overline{\theta} = \min_{i \in I_B : u_i > 0}\{\frac{x_{B(i)}}{u_i}\}$ and $l = \arg\min_{i \in I_B : u_i > 0}\{\frac{x_{B(i)}}{u_i}\}$
    Set $x_j = \overline{\theta}$ and $x_B = x - \overline{\theta}u$. Form new basis $I_B = I_B \setminus \{l\} \cup \{j\}$.
    Calculate $\overline{c}_j = c_j - c_B^\top B^{-1}A_j$ for all $j \in I_N$.
  end if
end while
return optimal basis I_B and optimal solution x.

The matrix B consists the current basis matrix, consisting of the columns of A corresponding to the basic variables. x is the current basic feasible solution (BFS), $\overline{c}$ the vector of reduced costs, I_N the set of indices of the non-basic variables, u the direction vector, calculated as $u = B^{-1}A_j$, z The objective function value. $\overline{\theta}$ is The step length, calculated as $\overline{\theta} = \min_{i \in I_B, u_i > 0}\{\frac{x_B(i)}{u_i}\}$, x_B The current values of the basic variables, I_B The set of indices of the basic variables, c_j the cost coefficient for the j variable, c_B the cost coefficients for the basic variables, A_j the j column of the matrix A.

2.4 Relaxations

Consider an integer programming problem expressed as

$$z = \min_x\{c^\top x : x \in X \subseteq \mathbb{Z}^n\}.$$

Most methods for proving optimality rely on bounding the optimal solution. This involves constructing an increasing sequence of lower bounds

$$\underline{z}_1 < \underline{z}_2 < \cdots < \underline{z}_s \leq z$$

and a decreasing sequence of upper bounds

$$\overline{z}_1 > \overline{z}_2 > \cdots > \overline{z}_t \geq z$$

to achieve the tightest possible lower ($\underline{z} \leq z$) and upper ($\overline{z} \geq z$) bounds.

Definition 2.1 (Relaxation) A problem

$$(RP) : z_{RP} = \min\{\overline{c}^\top x : x \in \overline{X} \subseteq \mathbb{R}^n\}$$

is a relaxation of the problem

$$(P) : z = \min\{c^\top x : x \in X \subseteq \mathbb{R}^n\}$$

if $X \subseteq \overline{X}$ and $\overline{c}^\top x \leq c^\top x$ for all $x \in X$.

Proposition 2.3. If RP is a relaxation of P, then z_{RP} serves as a dual bound for z.

Proposition 2.4. The following statements hold:

1. If the relaxation RP is infeasible, then P is also infeasible.
2. Let x^* be an optimal solution for RP. If $x^* \in X$ and $\overline{c}^\top x^* = c^\top x^*$, then x^* is also an optimal solution for P.

2.4.1 Linear programming relaxation

In the context of solving (mixed-)integer programming problems, we use the concept of linear programming (LP) relaxations. Although we briefly discussed this in the previous chapter, we will now provide a precise definition.

Definition 2.5 (Linear programming (LP)relaxation) The LP relaxation of an integer programming problem $\min\{c^\top x : x \in P \cap \mathbb{Z}^n\}$ with $P = \{x \in \mathbb{R}^n_+ : Ax \leq b\}$ is the linear programming problem $\min\{c^\top x : x \in P\}$.

Proposition 2.6. Let P_1 and P_2 be formulations of the integer programming problem

$$\min_x\{c^\top x : x \in X\} \text{ with } X = P_1 \cap \mathbb{Z}^n = P_2 \cap \mathbb{Z}^n.$$

Assume P_1 is a tighter formulation than P_2 (i.e., $P_1 \subset P_2$).
Let $z^i_{LP} = \min\{c^\top x : x \in P_i\}$ for $i = 1, 2$. Then $z^1_{LP} \geq z^2_{LP}$ for any cost vector c.

2.5 Branch and bound method

The working principle behind this strategy is based on Proposition (2.7).

Proposition 2.7 Let $K = \{1, \ldots, |K|\}$ and $\bigcup_{k\in K} S_k = S$ be a decomposition of S. Let $z^k = \max_{\cdot x}\{c^T x : x \in S_k\}$ for all $k \in K$. Then

$$z = \max_{k\in K}\{z^k\}.$$

2.5.1 Bounding in enumerative trees

The main principle behind pruning branches in enumerative search trees is summarized in Proposition (2.8).

Proposition 2.8 Consider the problem P and let $S = \bigcup_{k\in K} S_k$ be a decomposition of S into smaller sets. Let $z^k = \max_{\cdot x}\{c^T x : x \in S_k\}$ for $k \in K$, and let $\overline{z}^k$ ($\underline{z}^k$) be an upper (lower) bound on z^k. Then

$$\overline{z} = \max_{k\in K}\{\overline{z}^k\} \quad \text{and} \quad \underline{z} = \max_{k\in K}\{\underline{z}^k\}.$$

2.5.2 Linear-programming-based branch-and-bound

Branch-and-bound refers to methods that solve relaxations of subproblems and use bounding information to prune branches in the enumerative search tree preemptively as shown in (Algorithm(2)).

Algorithm 2 LP-relaxation-based branch-and-bound Oliveira

1: **initialise.** $\mathcal{L} \leftarrow \{S\}$, $\underline{z} \leftarrow -\infty$, $\overline{x} \leftarrow -\infty$
2: **while** $\mathcal{L} \neq \emptyset$ **do**
3: select problem S_i from $\mathcal{L}$. $\mathcal{L} \leftarrow \mathcal{L} \setminus \{S_i\}$.
4: solve LP relaxation of S_i over P_i, obtaining z^i_{LP} and x^i_{LP}. $\overline{z}^i \leftarrow z^i_{LP}$.
5: **if** $S_i = \emptyset$ **then** return to step(2).
6: **else if** $\overline{z}^i \leq \underline{z}$ **then** return to step (2).
7: **else if** $x^i_{LP} \in \mathbb{Z}^n$ **then** $\underline{z} \leftarrow \max\{\underline{z}, \overline{z}^i\}$, $\overline{x} \leftarrow x^i_{LP}$; and return to step (2).
8: **end if**
9: select a fractional component x_j and create subproblems S_{i1} and S_{i2} with formulations P_{i1} and P_{i2}, respectively, such that

$$P_{i1} = P_i \cup \{x_j \leq \lfloor \overline{x}_j \rfloor\} \text{ and } P_{i2} = P_i \cup \{x_j \geq \lceil \overline{x}_j \rceil\}.$$

10: $\mathcal{L} \leftarrow \mathcal{L} \cup \{S_{i1}, S_{i2}\}$.
11: **end while**
12: **return** $(\overline{x}, \underline{z})$.

$\mathcal{L}$ the list of subproblems to be solved. S the initial problem, $\underline{z}$ the current lower bound on the optimal value, $\overline{z}$ the current upper bound on the optimal value, S_i the selected subproblem from the list $\mathcal{L}$, P_i the feasible region associated with subproblem S_i , z^i_{LP} the optimal value of the LP relaxation of subproblem S_i , x^{*i}_{LP} the optimal solution of the LP relaxation of subproblem S_i , $\mathbb{Z}^n$ the set of integer vectors of dimension n,x_j the selected fractional component of the solution x^{*i}_{LP}, S_{i1}, S_{i2} the subproblems created by branching on x_j, P_{i1}, P_{i2} the feasible regions associated with subproblems S_{i1} and S_{i2}, respectively. $\lfloor x_j \rfloor$ the greatest integer less than or equal to x_j, $\lceil x_j \rceil$ the smallest integer greater than or equal to x_j.

2.6 Valid inequalities

Cutting-plane methods work by iteratively approximating the set of inequalities $\tilde{A}x \leq \tilde{b}$ through the addition of constraints to the formulation P of IP. These added constraints are known as *valid inequalities*, which we define more precisely in Definition

Definition 2.9 (Valid inequality) An inequality $\pi^\top x \leq \pi_0$ is valid for $X \subset \mathbb{R}^n$ if $\pi^\top x \leq \pi_0$ holds for all $x \in X$.

2.7 The Chvátal-Gomory procedure

To establish a systematic procedure for generating valid inequalities in the context of solving integer programming problems, we will use a two-step process.

Proposition 2.10 (Valid inequalities for polyhedral sets) An inequality $\pi^\top x \leq \pi_0$ is valid for $P = \{x \in \mathbb{R}^n : Ax \leq b, x \geq 0\}$, if and only if $P \neq \emptyset$ and there exists $u \geq 0$ such that $u^\top A \geq \pi$ and $u^\top b \leq \pi_0$.

Proposition 2.11 (Valid inequalities for integer sets) Let $X = \{y \in \mathbb{Z}^1 : y \leq b\}$. The inequality $y \leq \lfloor b \rfloor$ is valid for X.

Definition 2.12 (Chvátal-Gomory procedure) Consider the integer set $X = P \cap \mathbb{Z}^n$ where $P = \{x \in \mathbb{R}^n_+ : Ax \leq b\}$, A is an $m\mathbf{x}n$ matrix with columns $\{A_1, \dots, A_n\}$ and $u \in \mathbb{R}^m_+$.

The Chvátal-Gomory procedure consists of the following set of steps to generate valid inequalities for X:

1. $\sum_{j=1}^n u^\top A_j x_j \leq u^\top b$ is valid for P as $u \geq 0$;

2. $\sum_{j=1}^n \lfloor u^\top A_j \rfloor x_j \leq u^\top b$ is valid for P as $x \geq 0$;

3. $\sum_{j=1}^n \lfloor u^\top A_j \rfloor x_j \leq \lfloor u^\top b \rfloor$ is valid for X as $\lfloor u^\top b \rfloor$ is integer.

Theorem 2.13 Any inequality that holds for the set X can be derived by executing the Chvátal-Gomory process a finite number of times.

2.8 The cutting-plane method

In any scenario where we have a collection of inequalities and a procedure for solving the separation problem, we can formulate a general cutting-plane method. This is outlined in algorithm 3 the rationale for utilizing cutting-plane techniques.

Algorithm 3 Cutting-plane algorithm Oliveira

```
initialise. let F ⊆ {(π, π_0) : π^⊤ x ≤ π_0 is valid for X}. k = 0.
while x^k_LP ∉ Z^n do
    solve the LP relaxation over P, obtaining the optimal objective value z^k_LP
    and optimal solution x^k_LP.
    if x^k_LP ∉ Z^n then find (π^k, π^k_0) ∈ F such that π^k^⊤ x^k_LP > π^k_0.
    else
        return (x^k_LP, z^k_LP).
    end if
    P ← P ∪ {π^k^⊤ x ≤ π^k_0}. k = k + 1.
end while
return (x^k_LP, z^k_LP).
```

$\mathcal{F}$ set of valid inequalities $\{(\pi, \pi_0) : \pi^\top x \leq \pi_0$ is valid for $X\}$, X set of all feasible integer solutions, x^k_{LP} optimal solution to the linear programming (LP) relaxation at iteration k. z^k_{LP} optimal objective value of the LP relaxation at iteration k. k iteration counter.(π^k, π^k_0) a valid inequality from the set $\mathcal{F}$ used to cut off the current LP solution x^k_{LP} if it is not integral. P the feasible region of the problem defined by the current set of constraints.

2.9 Modern mixed-integer linear programming solvers

In this context, we will explore some of the key features common to most professional-grade mixed-integer programming (MIP) solvers. It will become evident that MIP solvers are composed of a complex array of techniques developed over the past few decades.

Continuous enhancement and the development of new techniques have led to performance improvements surpassing those achieved by hardware advancements alone. This is a dynamic and exciting research area, with new features being regularly proposed and integrated into these solvers with frequent updates.

The main distinction between MIP solver implementations lies in the specific "tricks" and techniques they employ. Often, the details of these techniques are not fully disclosed, as high-performing solvers are commercial products protected by trade secrets. Fortunately, some open-source options (such as CBC and HiGHS) and free-to-use alternatives (such as SCIP) are available ,(CBC,HiGHS and SCIP are mathematical programming solvers used to solve optimization problems) though they do not yet match the performance of commercial implementations Oliveira.

We will focus on the most important techniques that constitute a professional-grade MIP solver implementation. Most MIP solvers offer substantial tuning capabilities and on-off toggling of these techniques. Therefore, understanding the most crucial techniques and their functions can be beneficial for configuring MIP solvers to meet specific needs.

Most MIP solvers implement a method called branch-and-cut, which combines the linear-programming (LP)-based branch-and-bound method and cutting-plane method (discussed this previously) employed at the root node (or the first subproblem LP relaxation) and potentially at later nodes as well. Figure (2.1) shows the typical flowchart of a MIP solver algorithm. The initial phase consists of preprocessing, known as presolve. Here, the problem formulation is analyzed to identify and remove redundant constraints or "loose" variables. More advanced techniques may be used to infer the optimal value of some variables through logic or to tighten their bounds. For simpler problems, the presolve phase might return an optimal solution or certify that the problem is infeasible or unbounded.

Subsequently, the main solution loop begins, similar to the branch-and-bound method. A node selection method is used, and the LP relaxation is solved. Then, branching is applied, and the process continues until an optimal solution is found.

The primary difference, however, is the inclusion of the extra cuts and heuristics phases. Along with the presolve, these phases likely vary the most between MIP solver implementations. The cut phase involves applying a cutting-plane method to the current

LP relaxation to either obtain an integer solution (pruning the branch by optimality) or strengthen the LP relaxation formulation. Each solver uses its own set of cuts during this phase, often employing a collection of them simultaneously. The heuristics phase is combined to find primal feasible solutions from the LP relaxations (possibly augmented by cuts) so that primal bounds (integer and feasible solution values) can be obtained and shared across the search tree, promoting pruning by bound. Oliveira

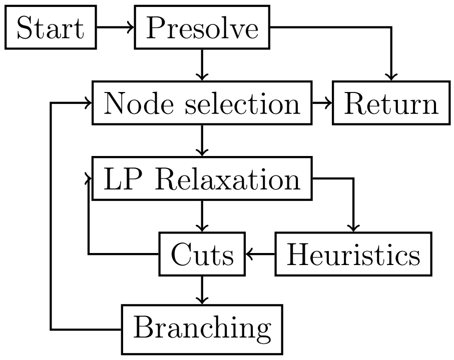

Figure 2.1: The flowchart of a typical MIP solver. The nodes represent phases of the algorithm

2.10 Conclusion

In this chapter, we presented the methods of simplex,branch and bound and branch and cut in mixed integer linear programming.

Chapter 3

Modeling Home Energy Management System (HEMS)

3.1 Introduction

In the context of the ongoing global energy transition, the efficient management of energy consumption has become a critical challenge for both consumers and electricity providers. The increasing penetration of renewable energy sources, such as solar and wind energy, into the electrical grid introduces significant variability and unpredictability in energy supply. Consequently, there is a growing need for advanced energy management strategies that can optimize energy consumption, reduce costs, and ensure the reliability of electrical systems. One promising approach to address these challenges is the application of predictive control in scheduling household appliances and Home Energy Management Systems (HEMS). The mathematical model presented in our thesis is heavily based on the work of Nagpal et al. [2020].

3.2 Description of the Problematic

This study introduces an HEMS architecture aimed at reducing peak power consumption and electricity costs through automated appliance scheduling. It relies on bi-directional communication with the power grid for real-time data exchange. The system operates in two modes based on whether appliances are thermal or non-thermal, using linear programming for optimization. It maintains peak power consumption within specified limits set by the utility. The scheduling algorithm adapts in real-time, and users set deadlines for non-thermal appliances, receiving warnings if deadlines are too short Halvgaard et al. [2012b].

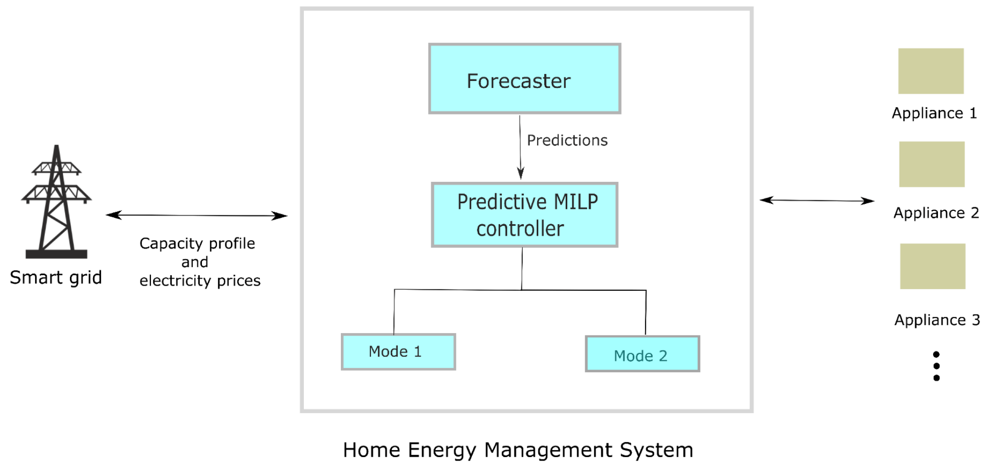

Figure 3.1: System architecture of HEMS, MILP = mixed-integer-linear-programming.

- The figure shows how the HEMS integrates weather data, electricity prices, and appliance characteristics into a centralized system that uses predictive control and optimization to manage household energy consumption efficiently. The MILP framework allows the system to handle the complexity of scheduling different types of appliances under various constraints, achieving significant reductions in energy costs and peak power consumption while maintaining user comfort , mode 1 and mode 2 refer to two different modes of operation for the home energy management system (HEMS) based on load category (thermal/non-thermal).Mode 1 a mix of thermal and non-thermal appliances are operated together.Mode 2 only the thermal appliances are activated.The capacity profile in this context likely refers to the maximum power consumption allowed for the appliances at any given time. The double-sided arrow typically represents a bidirectional communication link between the HEMS and the power grid, enabling data exchange and control signals between the two entities.

3.3 System Architecture

3.3.1 Load Categorization

The load categorization in the Home Energy Management System (HEMS) is divided into two main types based on the dynamics of the appliances:

- Thermal Load
  - Examples: Heat pumps, room heaters, water heaters, refrigerators.
  - Dynamics: Includes temperature as a system state.
- Non-Thermal Load
  - Examples: Washing machines, clothes dryers, dishwashers.
  - Characteristics:
  - Preemptive: Can be paused or interrupted as needed.
  - Fixed Duration: Operates for a fixed time to complete tasks.
  - Deadline: Must complete tasks by a user-defined deadline.

3.3.2 System Layout

The system layout of the Home Energy Management System (HEMS) presented in the Figure3.1 includes the following components:

- Forecaster: This component is responsible for making predictions about external weather conditions and heat gains due to solar irradiance in the house. Data-driven prediction techniques are commonly used based on historical data

- Predictive MILP Controller: The key component of the HEMS and operates in two modes based on electrical load categories. It solves different mathematical optimization problems in each mode using predictions from the forecaster about weather conditions. The controller receives electricity prices and consumption-capacity profiles from the smart-grid operator via smart-meter communication.

3.4 Appliance Dynamics Modeling and Setup

3.4.1 State-Space Model

The scheduling algorithm requires mathematical modeling of appliances into a discrete linear state-space representation. This model encapsulates control inputs, outputs, state variables, and external disturbances and is written as in Equation(3.1).

$$x_{k+1} = Ax_k + B\mathbf{u}_k + E\mathbf{d}_k \tag{3.1}$$

$$y_k = Cx_k.$$

- $x \in \mathbb{R}^n$ is the state vector
- $\mathbf{u} \in \mathbb{R}^m$ is the control vector
- $y \in \mathbb{R}^p$ is called the output vector
- $\mathbf{d} \in \mathbb{R}^r$ is called the disturbance vector
- $A \in \mathbb{R}^{nxn}$ is the state matrix
- $B \in \mathbb{R}^{nxm}$ is the control matrix
- $C \in \mathbb{R}^{pxn}$ is the output matrix
- $E \in \mathbb{R}^{rxn}$ is the disturbance matrix
- k denotes the time instant.

3.4.2 Thermal Load Appliances Modeling

The house uses a water source heat pump for indoor heating via floor heating pipes and a solar water heater system for hot water. Both systems' performance depends on solar irradiance and ambient temperature.

- Heat pump:a reduced order model of a residential building equipped with a heat-pump is developed in Halvgaard et al. [2012b] Staino et al. [2016] using the heat balance equations:

$$C_{p,r}\,\dot{T}_r = (UA)_{fr}(T_f - T_r) - (UA)_{ra}(T_r - T_a) + (1-p)\phi_s \tag{3.2}$$

$$C_{p,f}\,\dot{T}_f = (UA)_{wf}(T_w - T_f) - (UA)_{fr}(T_f - T_r) + p\phi_s$$

$$C_{p,w}\,\dot{T}_w = \eta W_c - (UA)_{wf}(T_W - T_f).$$

  - $C_{p,r}$, $C_{p,f}$ and $C_{p,w}$ are thermal capacities for the room, floor, and water, respectively.
  - T_r , T_f and T_w are the temperatures of the room, floor, and water, respectively.
  - T_r represents a variable that is a derivative with respect to time. Therefore, T_r is a temperature variable that changes over time, and $\dot{T}_r$ its derivative represents the rate of change of temperature with respect to time.
  - $(UA)_{fr}$, $(UA)_{wf}$ and $(UA)_{ra}$ are heat transfer coefficients between different components.
  - η is the efficiency of the heat pump compressor.
  - Equation (3.2) discretized with appropriate sample time using zero-order-hold model.
  - Converted into state-space representation as shown in Equation (3.1).

- The first sub-equation in Equation (3.2) describes the indoor environment's thermal dynamics of the building, depending on floor temperature T_f , ambient temperature T_a, and solar gains ϕ_s .

- The second sub-equation represents heat transfer between the floor and the underfloor heating system.

- The last sub-equation relates the floor temperature T_f to the work done W by the compressor.

- These equations describe the heat balance within the building. The first equation captures the thermal dynamics of the indoor environment, the second equation represents the heat transfer between the floor and the heating system, and the third equation relates the water temperature to the work done by the compressor.

Following that, a state-space model for the heat pump is established, we get

$$\mathbf{x} = \begin{bmatrix} T_r & T_f & T_w \end{bmatrix}^T, \quad \mathbf{u} = W_c,$$
$$\mathbf{d} = \begin{bmatrix} T_a & \phi_s \end{bmatrix}^T, \quad \text{and} \quad \mathbf{y} = T_r.$$

- Solar water tank:

$$C_t\dot{T}_t = \eta_h P_h + \phi_s - Q_c - (UA)_t(T_t - T_i) \tag{3.3}$$

  - Equation (3.3) captures the thermal dynamics of the tank.

  - It is a function of:

    * Inlet water temperature T_i.

    * Power consumption of the heating element P_h.

    * Solar gains ϕ_s.

After discretization and rewriting Equation (3.3) in state-space representation, we get $\mathbf{x} = \mathbf{y} = T_t$, $\mathbf{u} = P_h$, and $\mathbf{d} = \begin{bmatrix} Q_c & \phi_s & T_i \end{bmatrix}^T$.

3.4.3 Non-Thermal Load Appliances Modeling

The dynamics of a non-thermal load appliance are modeled using the following equation:

$$\zeta_{k+1} = \zeta_k + \psi_k, \tag{3.4}$$

- ζ is the variable that tracks the duration for which the appliance has run.
- ψ is a binary integer variable ($\psi \in \{0,1\}$) indicating the operational status of the appliance at time k (1 if running, 0 if halted).

This model can be represented in the state-space form, where:

- $\mathbf{x} = \mathbf{y} = \zeta$
- $\mathbf{u} = \psi$

Whenever ψ is 1, the appliance consumes a certain amount of power.

3.5 Problem Formulation of Thermal Appliances

In this section, the task of scheduling appliances is formulated as an MILP. The appropriate constraints associated with the appliances are also described. The objective of the program is to reduce the total electricity cost and peak power consumption at the same time. As mentioned previously, an MPC (is an advanced method of control that emerges from application in process industry in late 70s and early 80s Carlos E. García †) framework is applied to operate the appliances.Nagpal et al. [2020] At each time step an MILP is formulated with a prediction horizon of N time steps ahead from the current time step.

3.5.1 Energy Cost

The electricity cost associated with the energy consumption of thermal appliances is expressed mathematically as:

$$J_{\mathcal{P}} = \sum_{k \in \mathcal{N}} \sum_{i \in \mathcal{P}} c_k \mathbf{u}_{k,i} \tag{3.5}$$

where

- $\mathcal{N}$ is the prediction horizon, representing the set of future time steps.
- $\mathcal{P}$ denotes the set of thermal appliances.
- c_k is the real-time electricity price at k^{th} time step.
- $\mathbf{u}_{k,i}$ represents the power consumption of the i^{th} thermal appliance at the time step k.

3.5.2 Comfort Zone Constraints

The user's preference is to maintain the temperature of the building or water tank within a specified comfort zone. These constraints can be imposed as:

$$\mathbf{y}_{k,\min,i} \leq \mathbf{y}_{k,i} \leq \mathbf{y}_{k,\max,i}, \quad k \in \mathcal{N},\ i \in \mathcal{P}, \tag{3.6}$$

where

- $\mathbf{y}_{\max,i}$ and $\mathbf{y}_{\min,i}$ are upper and lower bound on the temperature of the i^{th} thermal appliance.

3.5.3 Maximum Power Consumption Constraint

Each thermal appliance is assumed to have a maximum and minimum power consumption limit at each time step, ensuring that the power consumed does not exceed certain thresholds. This constraint can be expressed as:

$$\mathbf{u}_{\min,i} \leq \mathbf{u}_{k,i} \leq \mathbf{u}_{\max,i} \quad k \in \mathcal{N}, i \in \mathcal{P}, \tag{3.7}$$

where

- $\mathbf{u}_{\min,i}$ and $\mathbf{u}_{\max,i}$ represent the upper and lower bound on the power consumption of i^{th} thermal appliance.

3.6 Problem Formulation of Non-Thermal Appliances

3.6.1 Energy Cost

The energy cost associated with non-thermal appliances is similar to that of thermal appliances. It can be represented mathematically as follows:

$$J_{\mathcal{Q}} = \sum_{k \in \mathcal{N}} \sum_{j \in \mathcal{Q}} c_k P_{k,j} \psi_{k,j} \quad \psi = \{0, 1\}. \tag{3.8}$$

- $J_{\mathcal{Q}}$ is the total energy cost for operating the activated non-thermal appliances.
- c_k is the real-time electricity price at k^{th} time step.
- $P_{k,j}$ is the power consumed by the j^{th} non-thermal appliance at the k^{th} time step.
- $\psi_{k,j}$ is a binary variable associated with the j^{th} non-thermal appliance at the k^{th} time step, indicating whether the appliance is ON (ψ=1) or OFF (ψ=0).

3.6.2 Deadline Constraint

Each non-thermal appliance must complete its task within a user-defined deadline, which must be longer than the appliance's running time. This is ensured by the equation

$$\Omega_j \leq (e_j - b_j) \quad j \in \mathcal{Q}, \tag{3.9}$$

where

- Ω_j is the running time
- e_j is the deadline
- b_j is the start time

The Home Energy Management System (HEMS) enforces this by scheduling the appliance to start at the next time slot and warning the user if the deadline is too short.

3.6.3 Start-Up Cost

The non-thermal appliance start-up cost is represented by the equation:

$$J_s = \sum_{k \in \mathcal{N}} \sum_{j \in \mathcal{Q}} \gamma_j \psi_{k,j} \tag{3.10}$$

where

- J_s is the total start-up cost of non-thermal appliances.
- γ_j refers to the start-up cost of the j^{th} non-thermal appliance

3.6.4 Operation Time Constraint

To complete the assigned task, non-thermal appliances must operate for a fixed duration Ω This requirement can be expressed with the following constraint:

$$\sum_{k=b_j}^{e_j} \psi_{k,j} = \Omega_j \quad j \in \mathcal{Q}, \psi = \{0, 1\}. \tag{3.11}$$

The constraint in Equation (3.11) guarantees that the appliance operates for its entire duration between its activation time and its deadline. Nagpal et al. [2020]

3.6.5 Total Capacity Constraint

The Total Capacity Constraint limits the maximum total power consumption by all appliances at each time step and is represented by the equation:

$$\sum_{i \in \mathcal{P}} \mathbf{u}_{k,i} + \sum_{j \in \mathcal{Q}} P_{k,j} \psi_{k,j} \leq C_k, \quad k \in \mathcal{N}, i \in \mathcal{P}, j \in \mathcal{Q}, \psi \in \{0, 1\} \tag{3.12}$$

C_k represents the maximum power available for consumption at time step k. This capacity constraint is set by the utility company and can vary over time based on electricity demand and grid operational costs. This is a hard constraint, meaning that if it is not met, the optimization problem will stop, which is undesirable. To address this, a soft constraint on total capacity is introduced in the next subsection.

3.7 Soft Constraints

Soft Constraints are introduced to allow for some violation of relevant constraints in the optimization problem. Slack variables are used to enable compromise between optimal scheduling and constraint violation.Nagpal et al. [2020] The Soft Constraints are represented by the following equations:

$$\mathbf{y}_{k,\min,i} \leq \mathbf{y}_{k,i} + \mathbf{v}_{k,i} \quad k \in \mathcal{N}, i \in \mathcal{P}, \tag{3.13}$$

$$\mathbf{y}_{k,\max,i} \geq \mathbf{y}_{k,i} - \mathbf{v}_{k,i} \quad k \in \mathcal{N}, i \in \mathcal{P}, \tag{3.14}$$

$$\sum_{i \in \mathcal{P}} \mathbf{u}_{k,i} + \sum_{j \in \mathcal{Q}} p_{k,j} \psi_{k,j} \leq C_k - \mathbf{w}_k \quad k \in \mathcal{N}, i \in \mathcal{P}, j \in \mathcal{Q}. \tag{3.15}$$

where

- $\mathbf{v}$ and $\mathbf{w}$ are slack variables introduced to allow for constraint violation.

3.8 Scheduling Algorithm

Upon activation, the (HEMS) first verifies the status of non-thermal appliances by checking their activation flags, which indicate either 1 (on) or 0 (off). The HEMS then determines its mode of operation based on these activation flags.

3.8.1 Non-Thermal

In the scheduling algorithm for (HEMS), Mode 1 can be mathematically represented as follows:

$$\min_{\mathbf{u},\mathbf{v},\mathbf{w}} \mathcal{J}_{\mathcal{P}} + \mathcal{J}_{\mathcal{Q}} + \mathcal{J}_{\mathcal{S}} + \sum_{k \in \mathcal{N}} (\beta_{\mathbf{w}}(\mathbf{w}_k) + \sum_{i \in \mathcal{P}} \alpha_{\mathbf{v},i}(\mathbf{v}_{k,i})) \tag{3.16}$$

subject to

$$\mathbf{u}_{min,i} \leq \mathbf{u}_{k,i} \leq \mathbf{u}_{max,i} \quad k \in \mathcal{N}, i \in \mathcal{P} \tag{3.17}$$

$$\sum_{k=b_j}^{e_j} \psi_{k,j} = \Omega_j, \quad j \in \mathcal{Q}, \ \psi = \{0, 1\}, \tag{3.18}$$

$$\mathbf{y}_{k,min,i} \leq \mathbf{y}_{k,i} + \mathbf{v}_{k,i} \quad k \in \mathcal{N}, i \in \mathcal{P} \tag{3.19}$$

$$\mathbf{y}_{k,\max,i} \geq \mathbf{y}_{k,i} - \mathbf{v}_{k,i} \quad k \in \mathcal{N}, i \in \mathcal{P} \tag{3.20}$$

$$\sum_{i \in \mathcal{P}} \mathbf{u}_{k,i} + \sum_{j \in \mathcal{Q}} P_{k,j}\psi_{k,j} \leq C_k - \mathbf{w}_k \quad k \in \mathcal{N}, i \in \mathcal{P}, j \in \mathcal{Q}. \tag{3.21}$$

Here

- $\alpha_{\mathbf{v}}(\mathbf{v}) \geq 0$ and $\beta_{\mathbf{w}}(\mathbf{w}) \geq 0$ are convex penalty cost functions associated with the slack variables $\mathbf{v}$ and $\mathbf{w}$.

3.8.2 Thermal

In the scheduling algorithm for Home Energy Management Systems (HEMS), Mode 2 can be mathematically represented as follows:

$$\min J_{\mathcal{P}}, \tag{3.22}$$

subject to

$$\mathbf{u}_{min,i} \leq \mathbf{u}_{k,i} \leq \mathbf{u}_{max,i} \quad k \in \mathcal{N}, i \in \mathcal{P} \tag{3.23}$$

$$\mathbf{y}_{k,min,i} \leq \mathbf{y}_{k,i} + \mathbf{v}_{k,i} \quad k \in \mathcal{N}, i \in \mathcal{P} \tag{3.24}$$

$$\mathbf{y}_{k,\max,i} \geq \mathbf{y}_{k,i} - \mathbf{v}_{k,i} \quad k \in \mathcal{N}, i \in \mathcal{P} \tag{3.25}$$

$$\sum_{i \in \mathcal{P}} \mathbf{u}_{k,i} \leq C_k - \mathbf{w}_k \quad k \in \mathcal{N}, i \in \mathcal{P} \tag{3.26}$$

- In both modes and at each time step, a control decision policy $\{\mathbf{u}_k^*\}_{k=1}^N$ is obtained for next N time steps by solving the formulated optimization. MILP and only the first control input $\{\mathbf{u}_1^*\}$ which will be applied to the system and while rest of the control inputs $\{\mathbf{u}_k^*\}_{k=2}^N$ are discarded.

3.9 Conclusion

In this chapter, we discussed modeling our problem and the main concepts and theories related to household energy management.

Chapter 4

Numerical results

4.1 Introduction

In this chapter, we discuss some numerical evaluations of our problem while providing some analysis on the results obtained.Using MATLAB 2021b combined with the YALMIP package Lofberg [2004], CPLEX, MOSEK and INTLINPROG solvers Lofberg [2004], several numerical case studies were conducted.

4.2 Simulation Studies

4.2.1 Simulation Setup

- Thermal Appliances:Heat pumps and solar water heaters are considered.Table(4.1) contains the values of the parameters used in modeling of thermal appliances as described in reference [Halvgaard et al. [2012b], Halvgaard et al. [2012a]] such as desired indoor air temperature (18°C, 22°C) and water temperature range (50°C, 70°C) are specified. Maximum power consumption for the heat pump is 1 KW, and for the water heater, it is 2 KW. Daily water withdrawal demand profiles are generated using the dwhcalc toolbox Jordan and Vajen [2005].

Table 4.1: Thermal appliance parameters.

Parameter	Value	Unit	Parameter	Value	Unit
$C_{p,f}$	3315	$KJ/°C$	$(UA)_{fr}$	624	$KJ/°Ch$
$C_{p,r}$	810	$KJ/°C$	$(UA)_{fr}$	28	$KJ/°Ch$
$C_{p,w}$	836	$KJ/°C$	$(UA)_{wf}$	28	$KJ/°Ch$
C_t	3881	$KJ/°C$	$(UA)_{wf}$	29,84	$KJ/°Ch$
η	3	-	p	0,2	-
η_h	1	-			

- Non-Thermal Appliances:a washing machine and a dishwasher are included.These appliances are activated at random times each day with random deadlines.We used in case 2 the data from table (4.2).
  We have made some modifications and additions to this program, notably by adding an air conditioner, which is a non-thermal device.We have also adjusted the operating time and the rated power for both the washing machine and the dishwasher.We used the data in table (4.3) in both cases 1 and 3 :

Table 4.2: Non-thermal appliance parameters.

Appliance	Running Time (Ω)	Rated-Power (P)
Washing machine	2h	3KW
Dishwasher	2.5h	4KW

Table 4.3: Non-thermal appliance parameters after adjustments.

Appliance	Running Time (Ω)	Rated-Power (P)
Washing machine	1.5h	2 KW
Dishwasher	1.5h	2 KW
air conditioner	2h	3 KW

- Data Sources: Ambient temperature and solar radiation data are sourced from ASHRAE IWEC weather data files for Dublin, IrelandStaino et al. [2016] Electricity prices are taken from the wholesale Danish Energy Market Ma and Jørgensen.

4.2.2 Simulation Details

- Implementation: The setup is simulated over a 7-day period with a 15 minute slots prediction horizon. The problems are are solved using multiple solvers such as INTLINPROG ,MOSEK and CPLEX Lofberg [2004] integrated with the YALMIP toolboxLofberg [2004] .Table (4.4) shows the number of constraints and optimization methods associated with each mode.

Table 4.4: Non-thermal appliance parameters.

Mode	Number of Constraints	Optimization Problem
Mode 1 (WM)	7003	MILP
Mode 1 (DW)	7003	MILP
Mode 1 (AC)	7003	MILP
Mode 1 (WM and DW)	7290	MILP
Mode 1 (WM and AC)	7290	MILP
Mode 1 (DW and AC)	7290	MILP
Mode 1 (WM and DW and AC)	8140	MILP
Mode 2	6716	LP

  - WM:washing machine , DW:dishwasher , AC: air conditioner.

- Capacity Constraint: A maximum capacity constraint of 4 kw is imposed on the total power consumption of all appliances.

4.3 Results and Discussion

Case 1: We present the results only if devices non-thermal are used:

- This Figure 4.1 shows the power consumption of a washing machine over the week (72 hours). The power consumption values are consistently higher than other consumption values. Figure 4.2 displays power consumption data for the dishwasher, and Figure 4.3 shows power consumption data for the air conditioner. All Figures provide insights into energy usage patterns for these household appliances.

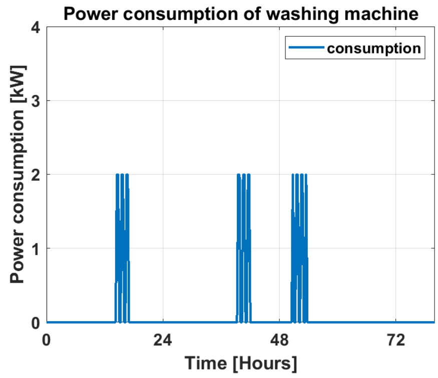

Figure 4.1: power consumption of washing machine

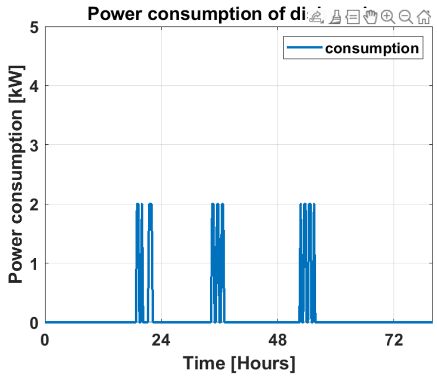

Figure 4.2: power consumption of dishwasher

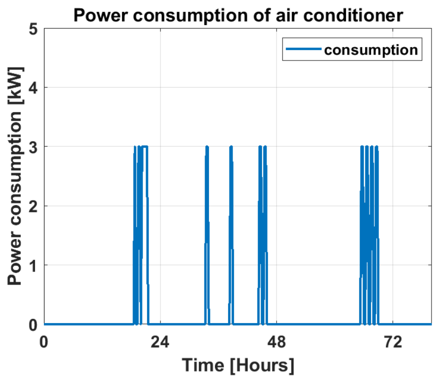

Figure 4.3: power consumption of air conditioner

- Figure 4.4 represents the total power consumption over time.The x-axis is labeled (Time [Hours]) ranging from 0 to 168 hours.The y-axis represents Power Consumption [KW].The solid blue bars indicate actual total power consumption values for all three appliances. There's a dashed horizontal line representing a capacity constraint set at 4 KW.

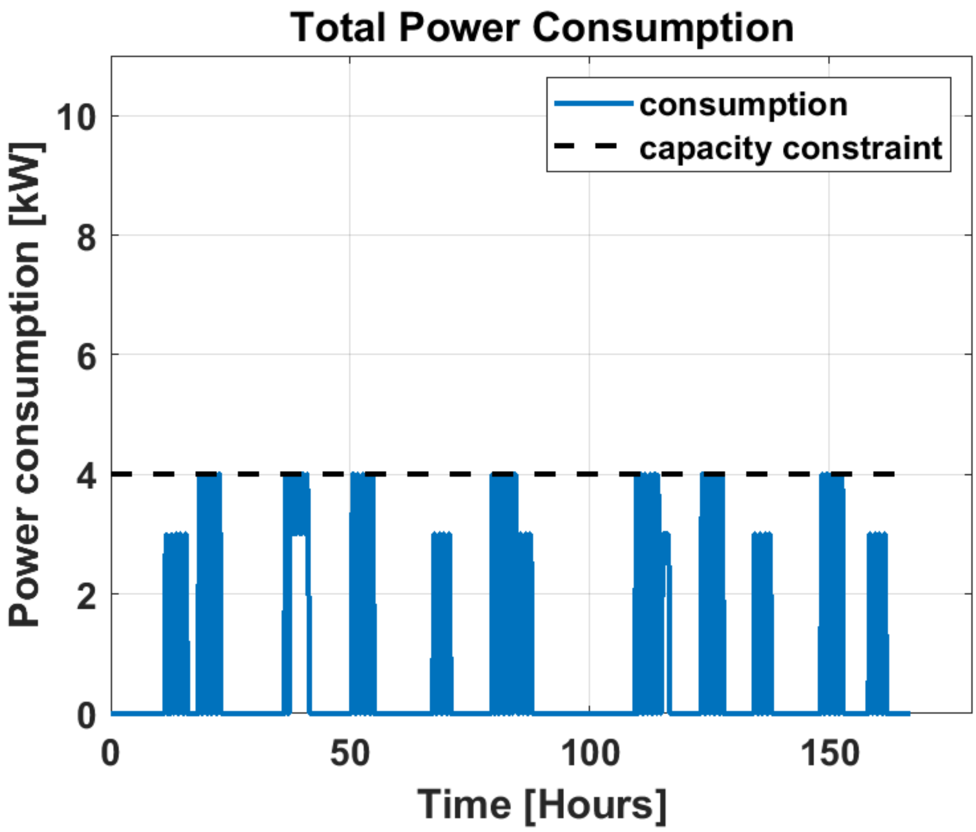

Figure 4.4: Total power consumption

Case 2: In this scenario, we are providing results of a combination of 2 thermal devices and two non-thermal appliances as used in the original work of Nagpal et al. [2020].

- Figure(4.5) illustrates the changes in indoor air temperature with HEMS and Figure (4.6) the power consumption of the heat pump with HEMS needed to maintain that temperature. Throughout the simulation, the indoor temperature remains within the specified comfort zone limits, and the power consumption constraint of the heat pump is consistently met.

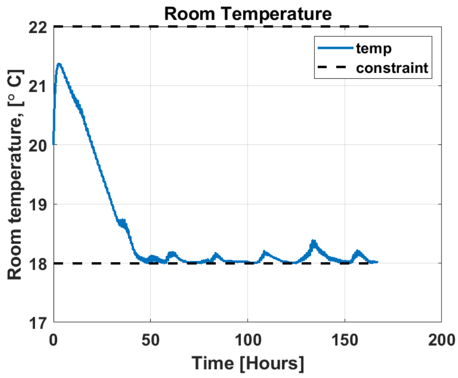

Figure 4.5: Room temperature

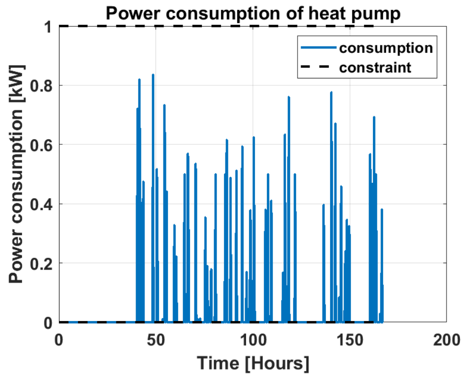

Figure 4.6: power consumption of heat pump

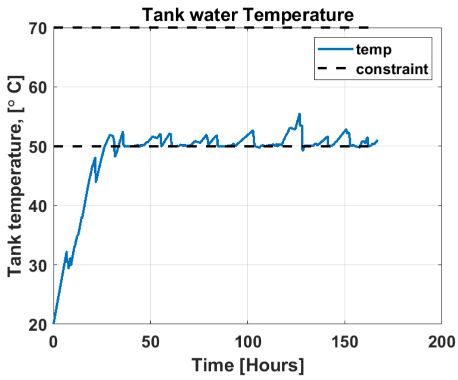

Figure 4.7: tank water temperature

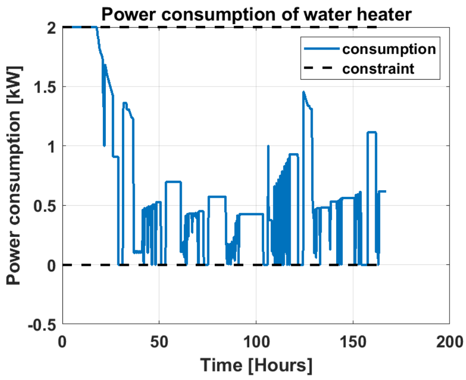

Figure 4.8: power consumption of water heater

- Figure (4.7) depicts the fluctuation of water temperature in the tank with HEMS and Figure(4.8) corresponding power consumption of the water heater with HEMS. Initially, the water temperature is set at 20°C (as shown in Figure 4.7). Over time, the temperature gradually rises over the next few hours and then remains within the specified range. This explains why the power consumption of the water heater is higher at the beginning of the simulation period, as illustrated in Figure (4.8).

- Power consumption profiles of the washing machine and dishwasher are displayed in Figures (4.9) and Figures (4.10) respectively.These appliances are intermittently interrupted during operation to minimize electricity costs and adhere to capacity constraints. Importantly, the Home Energy Management System (HEMS) operates without prior knowledge of when washing or dishwashing requests will occur. Requests are generated randomly to assess the HEMS's ability to handle immediate demand scenarios.

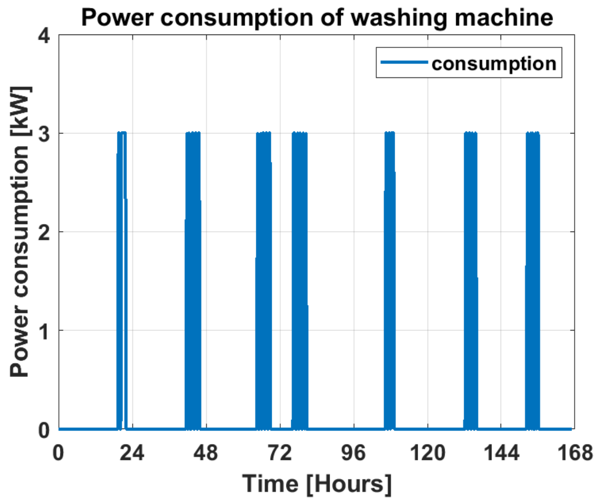

Figure 4.9: power consumption of washing machine

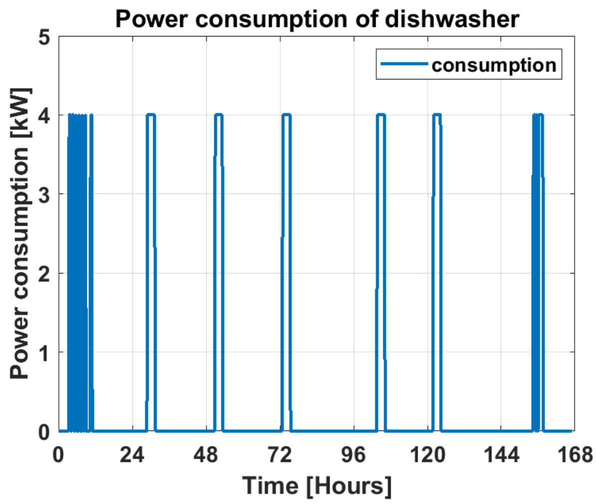

Figure 4.10: power consumption of dishwasher

- In Figure (4.11), the advantages of employing HEMS are evident, showcasing the total power consumption of all household appliances. Initially, Figure (4.12) illustrates that the total power consumption exceeds the 6 KW capacity limit.

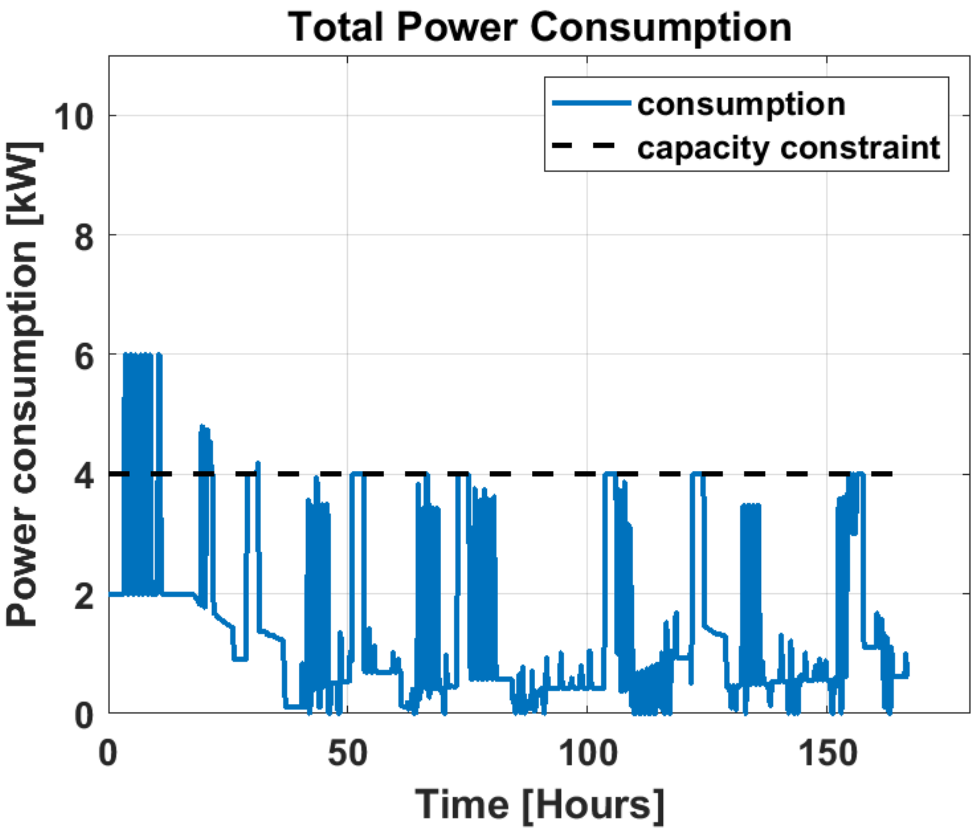

Figure 4.11: Total power consumption

Case 3: In this case we model a scenario similar to case 2 with the addition of a third non-thermal appliance, i.e. the air conditioner. The parameters related to the appliances operation times and power consumptions are modified as presented in Table [4.2] :

- The two presented graphs are titled room temperature Figure (4.12) and heat pump energy consumption Figure (4.13). The line on the graph starts slightly above 21°C, drops sharply to just below 19°C within the first few hours, then fluctuates slightly while generally trending downward Figure (4.12). The bars on Figures (4.13) represent different energy consumption values at different times hours of the 7 days.

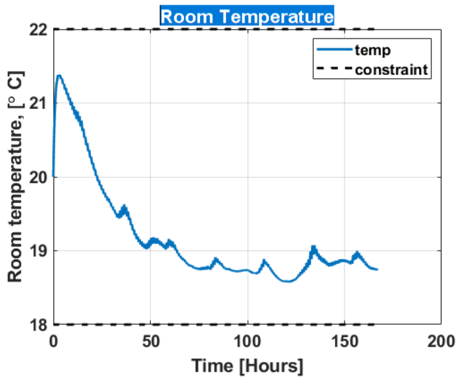

Figure 4.12: Room temperature

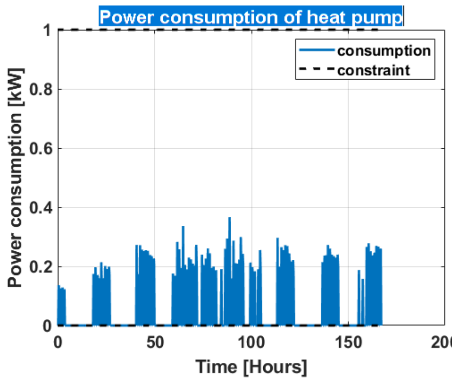

Figure 4.13: power consumption of heat pump

- The graphs show the water temperature in the tank in degrees Celsius over time (in hours) (Figure 4.14) and the energy consumption in kilowatts (kW) of the water heater (Figure 4.15). The temperature fluctuates around 60 degrees Celsius. A constant constraint is set at approximately 50 degrees Celsius (Figure 4.14). The consumption varies considerably over time, with peaks followed by periods of no consumption. A constant constraint is set at approximately 2 KW (Figure 4.15).

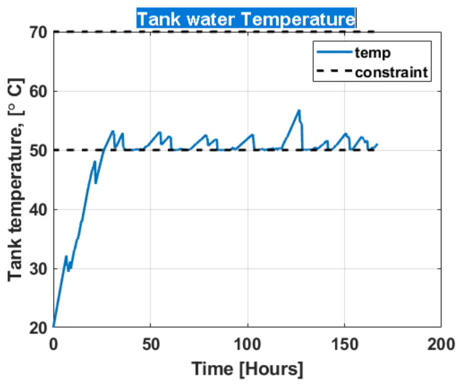

Figure 4.14: tank water temperature

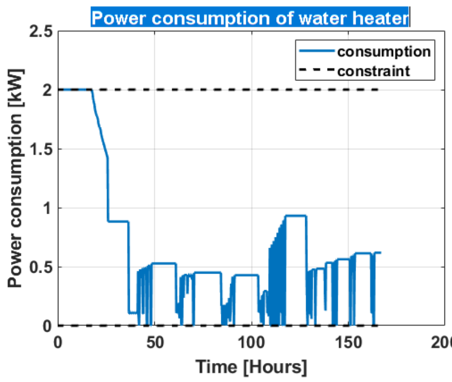

Figure 4.15: power consumption of water heater

- Figure 4.16 shows energy consumption peaks at different times. This suggests that the washing machine is used at specific times of the day or over a period of several days. Figure 4.17 presents similar patterns, but for the dishwasher. The peaks likely indicate the times when the dishwasher is running. Figure 4.18 also shows peaks, but this time for the air conditioner. Understanding these peaks can help adjust the temperature more efficiently, thereby reducing the electricity bill. In summary, these graphs provide insight into the energy consumption habits of household appliances.

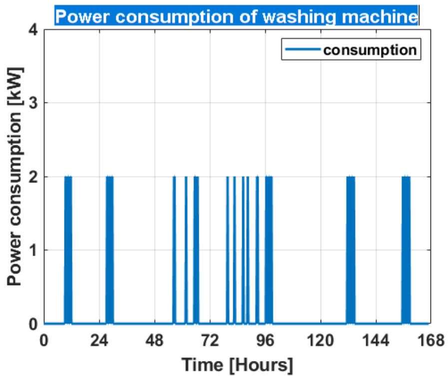

Figure 4.16: power consumption of wahing machine

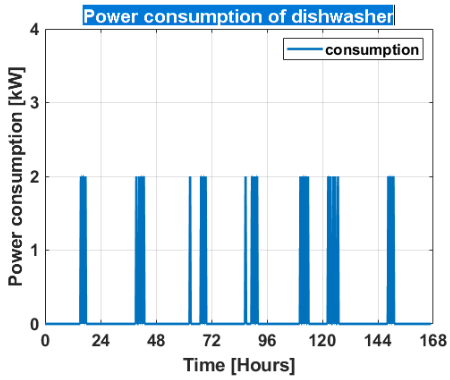

Figure 4.17: power consumption of dishwasher

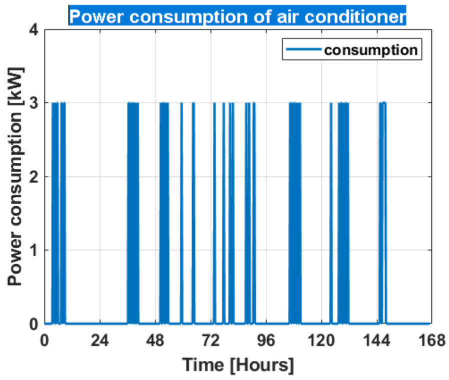

Figure 4.18: power consumption of air conditioner

- Figure 4.19, titled Total Energy Consumption, shows the trend of electricity consumption over time. The solid blue line represents the actual energy consumption over time, while the dashed black line represents the maximum available capacity.

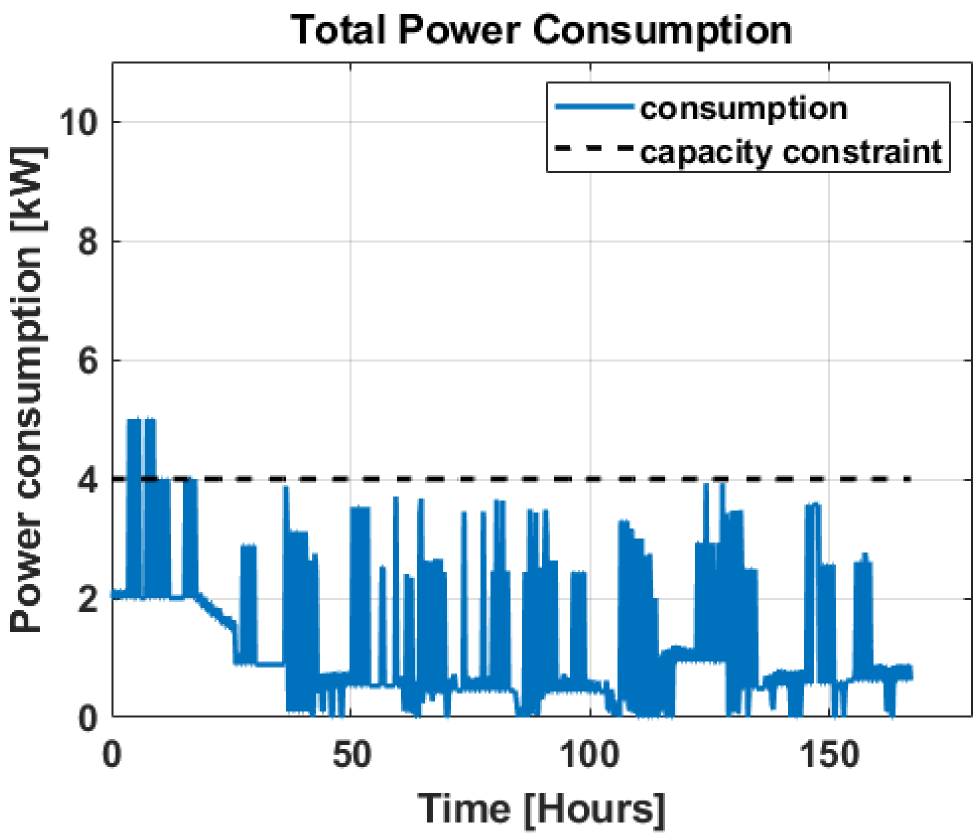

Figure 4.19: Total power consumption

- This table (4.5) presents the performances of different solvers for each case. The elapsed time and total cost are indicated for each case.INTLINPROG is optimal for scenarios involving only non-thermal devices due to its low cost and moderate elapsed time.CPLEX offers the best balance for mixed device scenarios, providing the fastest solution time and reasonable cost. MOSEK is not recommended due to its high elapsed time and cost, making it the least efficient option.Overall, CPLEX is the most versatile and efficient solver for a mixed device environment, while INTLINPROG is preferable for non-thermal-only applications.

Table 4.5: the performances of different solvers.

Case	Solver	Elapsed time(s)	Total cost
1	INTLINPROG	226,24	1,7137
2	CPLEX	180,51	3,2327
3	MOSEK	65415,28	7,9561

4.4 Conclusion

To reduce the household electricity bill using the HEMS system, we find that these results are satisfactory and can reduce electricity consumption.An existing program has been used for case 2 and modified accordingly in case 1 and case 3.In future work,we could compare all three solvers in all three cases.

General conclusion

In this thesis, we explored the potential of Home Energy Management Systems (HEMS) to reduce household electricity consumption. In particular, we implemented optimization techniques based on linear programming (LP),mixed-integer linear programming (MILP)constraints. These approaches allowed us to effectively and accurately model various aspects of the energy management problem.

To solve the optimization models, we used advanced methods such as branch and bound and branch and cut. These methods proved particularly effective in handling the inherent complexities of our problem, including nonlinear constraints and discrete variables. By applying these techniques, we were able to develop optimized solutions that minimize energy consumption while respecting the comfort and usage constraints of household electrical appliances.

Our results show that the use of HEMS, combined with sophisticated optimization techniques, enables significant energy savings for households. The simulations and tests carried out confirm that our approach effectively reduces electricity bills while maintaining user comfort levels. Furthermore, the resolution methods employed, such as branch and bound and branch and cut, demonstrated their robustness and efficiency in managing the complex constraints of the problem.

In conclusion, this work makes a significant contribution to research on domestic energy management. It demonstrates that HEMS, when combined with advanced optimization techniques, represent a promising solution for reducing household electricity consumption. The results obtained are encouraging and pave the way for practical applications and future research in this field. We hope that this study will encourage wider adoption of HEMS and continuous optimization of domestic energy management methods for a more sustainable and energy-efficient future .

Bibliography

M. M. Carlos E. García †, David M. Prett ‡. Model predictive control: Theory and practice—a survey.

M. Güzelsoy, T. K. Ralphs, and J. Cochran. Integer programming duality. In *Encyclopedia of Operations Research and Management Science*, pages 1–13. Wiley Hoboken, NJ, USA, 2010.

R. Halvgaard, P. Bacher, B. Perers, E. Andersen, S. Furbo, J. B. Jørgensen, N. K. Poulsen, and H. Madsen. Model predictive control for a smart solar tank based on weather and consumption forecasts. *Energy Procedia*, 30:270–278, 2012a.

R. Halvgaard, N. K. Poulsen, H. Madsen, and J. B. Jørgensen. Economic model predictive control for building climate control in a smart grid. In *2012 IEEE PES innovative smart grid technologies (ISGT)*, pages 1–6. IEEE, 2012b.

U. Jordan and K. Vajen. Dhwcalc: Program to generate domestic hot water profiles with statistical means for user defined conditions. In *Proceedings of the ISES Solar World Congress, Orlando, FL, USA*, volume 12. Citeseer, 2005.

K. Klamroth, J. Tind, and S. Zust. Integer programming duality in multiple objective programming. *Journal of Global Optimization*, 29:1–18, 2004.

J. Lofberg. Yalmip: A toolbox for modeling and optimization in matlab. In *2004 IEEE* *international conference on robotics and automation (IEEE Cat. No. 04CH37508)*, pages 284–289. IEEE, 2004.

Z. Ma and B. N. Jørgensen. Energy flexibility of the commercial greenhouse growers.

H. Nagpal, A. Staino, and B. Basu. Application of predictive control in scheduling of domestic appliances. *Applied Sciences*, 10(5):1627, 2020.

F. Oliveira. Optimisation notes. URL `https://github.com/gamma-opt/optimisation-notes`.

A. Staino, H. Nagpal, and B. Basu. Cooperative optimization of building energy systems in an economic model predictive control framework. *Energy and Buildings*, 128:713–722, 2016.

J. P. Vielma. Mixed integer linear programming formulation techniques. *Siam Review*, 57(1):3–57, 2015.

S. Wiese. On the interplay of mixed integer linear, mixed integer nonlinear and constraint programming. 2016.