Optimal design of local multi-energy systems : mixed-integer linear programming models and bi-level decomposition approaches

Bingqian Liu

université
PARIS-SACLAY

Optimal design of local multi-energy systems: mixed-integer linear programming models and bi-level decomposition approaches

Conception optimale de systèmes multi-énergies locaux : modèles de programmation linéaire à nombre entier mixte et approches de décomposition à deux niveaux

Thèse de doctorat de l'université Paris-Saclay

École doctorale n°580 : sciences et technologies de l'information et de la communication (STIC)
Spécialité de doctorat : Informatique
Graduate School : Informatique et sciences du numérique
Référent : Faculté des sciences d'Orsay

Thèse préparée dans l'unité de recherche **Laboratoire interdisciplinaire des sciences du numérique** (Université Paris-Saclay, CNRS), sous la direction de **Dominique QUADRI**, Professeure, le co-encadrement de **Céline GICQUEL**, Maîtresse de conférences, la co-supervision de **Côme BISSUEL**, Ingénieur chercheur et la co-supervision de **François COURTOT**, Docteur

Thèse soutenue à Paris-Saclay, le 2 juin 2022, par

Bingqian LIU

Composition du jury

Safia KEDAD-SIDHOUM Professeure, Conservatoire Nationale des Arts et Métiers	Présidente
François CLAUTIAUX Professeur, Université de Bordeaux	Rapporteur & Examinateur
Pierre FOUILHOUX Professeur, Université Sorbonne Paris Nord	Rapporteur & Examinateur
Frédéric BABONNEAU Professeur, KEDGE Business School	Examinateur
Sandra NGUEVEU Docteur, Toulouse INP	Examinatrice
Guillaume SANDOU Professeur, Université Paris-Saclay	Examinateur
Dominique QUADRI Professeure, Université Paris-Saclay	Directrice de thèse

Thèse de doctorat

NNT : 2022UPASG046

université PARIS-SACLAY

ÉCOLE DOCTORALE
Sciences et technologies de l'information et de la communication (STIC)

Titre: Conception optimale de systèmes multi-énergies locaux : modèles de programmation linéaire à nombre entier mixte et approches de décomposition à deux niveaux
Mots clés: Système multi-énergie, Programmation linéaire en nombres entiers, Optimisation de planification

Résumé: Un système multi-énergie local (LMES) est un système énergétique décentralisé produisant de l'énergie sous de multiples formes pour satisfaire les besoins énergétiques de clients situés dans son voisinage. Les LMES présentent de nombreux avantages, par exemple une efficacité de production plus élevée et des coûts de maintenance plus faibles. Toutefois, afin de fournir leurs meilleures performances potentielles, les LMES doivent être conçus avec soin.

Le problème de conception d'un LMES, en tant que problème d'optimisation combinatoire, est difficile à résoudre en raison de sa grande taille. De nombreux outils numériques pour aider les gens à concevoir des LMES reposent sur des hypothèses fortes et des simplifications, et ils limitent généralement la taille des instances considérées afin de pouvoir proposer des solutions dans un temps de calcul acceptable.

Cette thèse de doctorat se concentre sur la modélisation et la résolution des problèmes de conception optimale des LMES. En termes de modélisation du problème, nous permettons de construire le système en plusieurs phases d'investissement et de considérer l'opération détaillée aux pas de temps horaires. Le problème d'optimisation qui en résulte, sous la forme d'un grand programme linéaire à nombres entiers mixtes, est résolu par deux approches de décomposition : un algorithme de décomposition hiérarchique étendu et un algorithme de décomposition de Benders généralisé.

Les résultats numériques obtenus à partir de trois études de cas réels montrent d'abord que les algorithmes de décomposition proposés sont plus performants que les approches de résolution traditionnelles. Nos résultats montrent également que, même si certaines approximations sont appliquées, les plans de déploiement obtenus sont de très bonne qualité.

Title: Optimal design of local multi-energy systems: mixed-integer linear programming models and bi-level decomposition approaches
Keywords: Multi-energy system, Mixed-integer linear programming, Planning optimisation

Abstract: A local multi-energy system (LMES) is a decentralized energy system producing multiple forms of energy to satisfy the needs of customers in its vicinity. LMESs display many advantages, e.g., higher production efficiency and lower maintenance costs. However, to provide their best potential performance, LMESs should be carefully designed.

The design problem of an LMES, as a combinatorial optimization problem, is difficult to solve due to its large size. Many off-the-shelf numerical decision-aid tools rely on strong assumptions and simplifications, and they usually limit the size of the considered instances to enable the resolution process to terminate within an acceptable computation time.

This PhD thesis focuses on the modelling and resolution of optimal design problems of LMESs. In terms of problem modelling, we allow to construct the system within multiple investment phases and consider the detailed operation with hourly timestep. The resulting optimization problem, as a large mixed-integer linear program, are solved by two decomposition approaches: an extended hierarchical decomposition algorithm and a generalized Benders' decomposition algorithm.

The numerical results obtained from three real-life case studies first show that the proposed decomposition algorithms significantly outperform the traditional solving approaches. Our results also show that, even if some approximations are applied, the obtained deployment plans are of very good quality.

Acknowledgement

During the past three years of my doctoral study, I was fortunate to receive help and support from many people. This thesis could not have been realized without their help. I would like to express my sincere gratitude to them.

First, I would like to express my deepest gratitude to my supervisors of thesis: Dominique Quadri, Céline Gicquel, Côme Bissuel and François Courtot. Despite the difficulties caused by COVID-19, they helped me with patience and kindness, which allows me to carry out my work remotely. Especially the weekly meetings with Céline and Côme, the discussions and inspirations during these meetings are the key for me to accomplish this thesis.

I would also like to thank François Clautiaux and Pierre Fouilhoux for having accepted to be the reporters of my thesis. My gratitude also goes to Frédéric Babonneau, Safia Kedad-Sidhoum, Sandra Ngueveu and Guillaume Sandou for being part of the jury for my doctoral thesis.

Special thanks to Wenhui Du, the director of EDF R&D China, for proposing this excellent opportunity to me and for supporting my work during my doctoral study.

I would also like to thank my colleagues in EDF R&D and I appreciate the great time we had working together.

Finally, I would like to dedicate this thesis to my wife Xinran Wang, without whom I could not have accomplished this long journey. I would also like to thank my mother Linmei Zhang. Words cannot express my gratitude to her consistent encouragement and selfless love. I am also extremely grateful to my father Yuming Liu, who is the first and the best teacher in my life. I would also like to say to my two younger sisters, Qianqian Liu and Youyou Liu, I love you.

Abstract

Over the last decades, the energy industry has been striving to improve the energy production efficiency, to lower the greenhouse gas emissions related to energy production and distribution and to better integrate renewable energy resources. Local multi-energy systems (LMESs) are an interesting alternative to meet these challenging objectives. Basically, an LMES is a decentralized energy system producing energy within multiple forms to satisfy the energy needs of customers located in its vicinity. The customers correspond to a set of buildings belonging e.g. to a university campus, a hospital complex or a city district. LMESs have a higher production efficiency and a lower maintenance cost than traditional individual (i.e., single-building) energy systems.

LMESs thus display many practical advantages. However, in order to provide their best potential performance, they should be carefully designed. Designing an LMES essentially consists in selecting the energy conversion and storage devices making up the system. The obtained LMES should be able to satisfy the fluctuating energy demand at all time and to minimize the total construction and operation cost of the system over its lifetime usually spanning several decades.

Many off-the-shelf numerical decision-aid tools already exist to assist people in the design of LMESs. However, most of these tools rely on strong assumptions and simplifications. For example, they assume that the capacity of an energy conversion device may take any value within a predefined continuous range, while this capacity should in fact be selected within a discrete list of values corresponding to the available models produced by equipment manufacturers. Moreover, these tools usually strongly limit the size of the considered instances to enable the resolution process to terminate within an acceptable computation time.

This PhD thesis focuses on the problem of optimally designing an LMES involving both energy conversion and storage devices. In terms of problem modelling, we improve the current state-of-the art in several directions. First, we allow to choose the capacity of the installed devices within a predefined discrete list of value. Second, we consider the fact that building an LMES is not a one-step but rather a multi-step process in which investment decisions are made little by little to adjust the system layout to the long-term increase of the energy demand. We thus seek to build a multi-phase strategic deployment plan for the LMES. Third, we incorporate in the objective function the operation cost of the system over its lifetime. To estimate this cost as accurately as possible, we build detailed daily operation schedules using hourly time steps for a set of representative days. These schedules take into account several realistic complicating features such as the partial load efficiency and the minimum working load of energy conversion devices.

The resulting optimization problem is formulated as a very large mixed-integer linear program. To solve it efficiently, we develop two new decomposition algorithms exploiting the specific bi-level structure of the problem. The first algorithm extends a previously published hierarchical decomposition algorithm, the second one is a generalized Benders' decomposition algorithm.

The proposed modelling and solving approach are applied to three real-life case studies located in China. Our numerical results first show that the proposed decomposition algorithms significantly outperform both the generic branch-and-cut algorithm embedded in a mathematical programming solver and the original hierarchical decomposition algorithm at solving the mixed-integer linear program to

optimality. Our results also show that, even if some approximations are done in the problem modelling, the obtained deployment plans are of very good quality.

Résumé

Au cours des dernières décennies, l'industrie énergétique s'est efforcée d'améliorer l'efficacité de la production d'énergie, de réduire les émissions de gaz à effet de serre liées à la production et à la distribution d'énergie et de mieux intégrer les ressources énergétiques renouvelables. Les systèmes multi-énergies locaux (LMES) constituent une alternative intéressante pour atteindre ces objectifs ambitieux. Fondamentalement, un LMES est un système énergétique décentralisé produisant de l'énergie sous de multiples formes pour satisfaire les besoins énergétiques de clients situés dans son voisinage. Les clients correspondent à un ensemble de bâtiments appartenant, par exemple, à un campus, un complexe hospitalier ou un quartier urbain. Les LMES ont une efficacité de production plus élevée et un coût de maintenance plus faible que les systèmes énergétiques traditionnels fonctionnant au niveau d'un seul bâtiment.

Les LMES présentent donc de nombreux avantages pratiques. Toutefois, afin de fournir leurs meilleures performances potentielles, ils doivent être conçus avec soin. La conception d'un LMES consiste essentiellement à sélectionner les dispositifs de conversion et de stockage de l'énergie qui composent le système. Le LMES obtenu doit être capable de satisfaire à tout moment la demande énergétique fluctuante et de minimiser le coût total de construction et d'opération du système sur sa durée de vie, qui s'étend généralement sur plusieurs décennies.

Il existe déjà de nombreux outils numériques pour aider les gens à concevoir des LMES. Cependant, la plupart de ces outils reposent sur des hypothèses et des simplifications importantes. Par exemple, ils supposent que la capacité d'un dispositif de conversion d'énergie peut prendre des valeurs dans une gamme continue prédéfinie, alors que cette capacité devrait en fait être sélectionnée dans une liste discrète de valeurs correspondant aux modèles disponibles produits par les fabricants d'équipements. De plus, ces outils limitent généralement fortement la taille des instances considérées afin de pouvoir proposer des solutions dans un temps de calcul acceptable.

Cette thèse de doctorat se concentre sur le problème de la conception optimale d'un LMES impliquant à la fois des dispositifs de conversion et de stockage d'énergie. En termes de modélisation du problème, nous améliorons l'état de l'art actuel dans plusieurs directions. Premièrement, nous permettons de choisir la capacité des dispositifs installés dans une liste de valeurs discrètes prédéfinies. Deuxièmement, nous considérons le fait que la construction d'un LMES ne se fait pas en une seule étape mais est plutôt un processus en plusieurs étapes dans lequel les décisions d'investissement sont prises petit à petit pour ajuster le déploiement du système à l'augmentation à long terme de la demande d'énergie. Nous cherchons donc à construire un plan de déploiement stratégique en plusieurs phases pour le LMES. Troisièmement, nous incorporons dans la fonction objectif le coût d'opération du système sur sa durée de vie. Pour estimer ce coût aussi précisément que possible, nous construisons des plannings d'opération journaliers détaillés aux pas de temps horaires pour un ensemble de jours représentatifs. Ces plannings tiennent compte de plusieurs caractéristiques réalistes compliquées telles que l'efficacité à charge partielle et la charge minimale des dispositifs de conversion d'énergie.

Le problème d'optimisation qui en résulte est formulé comme un très grand programme linéaire à nombres entiers mixtes. Pour le résoudre efficacement, nous

développons deux nouveaux algorithmes de décomposition qui exploitent la structure spécifique à deux niveaux du problème. Le premier algorithme étend un algorithme de décomposition hiérarchique précédemment publié, le second est un algorithme de décomposition de Benders généralisé.

L'approche de modélisation et de résolution proposée sont appliquées à trois cas d'étude réels situés en Chine. Nos résultats numériques montrent d'abord que les algorithmes de décomposition proposés sont plus performants que l'algorithme générique de branch-and-cut intégré à un solveur de programmation mathématique et que l'algorithme original de décomposition hiérarchique pour résoudre le programme linéaire en nombres entiers à l'optimalité. Nos résultats montrent également que, même si certaines approximations sont effectuées dans la modélisation du problème, les plans de déploiement obtenus sont de très bonne qualité.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

A Local Multi-Energy System (LMES), or Decentralized Energy System, produces energy within multiple forms in order to satisfy the energy needs of customers located in its vicinity. The produced energy can be in the form of heating power, cooling power, domestic hot water or locally generated electricity The served customers may correspond to e.g. a small set of buildings [Wak+19; Meh+12], a hospital complex [AFF07] or a urban district [JFS14].

LMESs contrast with individual energy systems in which each single building is equipped with its own heating or cooling system. LMESs are usually much more efficient than individual energy systems with respect to both energy conversion and cost. Namely, individual energy systems mostly rely on small-capacity energy production devices whose conversion efficiency is much lower than the one of the medium-capacity devices that can be used in an LMES. In addition, the energy consumption pattern of a building depends on its function. For instance, the energy consumption of an office building concentrates in weekday daytime while the one of a residential building is higher at night and during the weekend. An LMES serves buildings with various functions. As a consequence, the total demand that it will have to satisfy will be somewhat less variable than the one of an individual building and is less likely to fall to zero, which will reduce the idle time of the system. Finally, the local generation of electricity may contribute in reducing the dependence of the customers from the national and global electricity markets, in which the price of electricity displays strong fluctuations.

An LMES is thus a decentralized energy system involving a variety of energy conversion devices (photo-voltaic panels, combined heat and power generators, boilers, chillers...) and in some cases, energy storage systems (hot/chilled water tanks, ice storage, batteries, ...). These devices are used to transform energy resources into multiple forms of energy to satisfy the demands of the system's customers. The energy resources may be local energy resources (solar, wind, waste heat, geothermal, ...) or conventional energy sources (electricity, gas, biofuels, ...). Figure 1.1 describes the structure of an illustrative local multi-energy system aiming at satisfying the local demand of power, gas, cold and heat by consuming and converting electricity and gas bought from the national distribution network.

Although LMESs display many practical advantages, they should be carefully designed in order to provide their best potential performance. This PhD thesis focuses on the LMES optimal design problem which can be described as follows. Given the available energy resources, the predicted total energy demand in the neighbourhood over the LMES lifetime and the available energy conversion and storage technologies, select the proper technologies to be installed in the system and

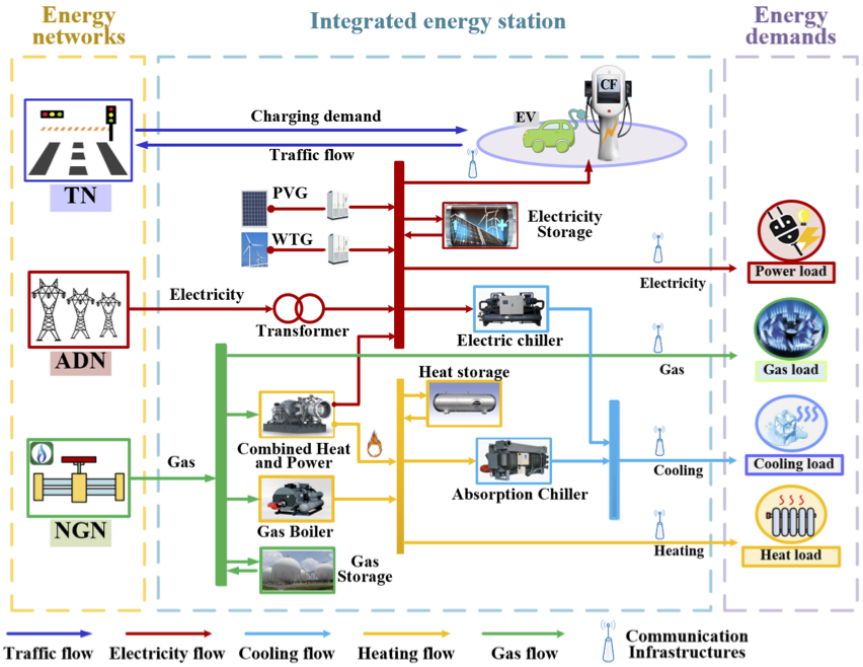


FIGURE 1.1: Example of a local multi-energy system [Xie+20]

determine the size (i.e. capacity) of the corresponding conversion and storage devices so as to minimize the total construction and operation cost of the system while satisfying the customers' demand at all time.

Indeed, design decisions will have a strong impact on the daily operational management of the LMES so that the system design should consider not only the construction cost but also the operation cost over the system lifetime which usually spans several decades. Namely, the operation cost typically represents between 20% and 60% of the total cost of LMES over its lifetime. Moreover, a given conversion device may be rather cheap in terms of purchasing and installation costs but lead to high operation costs due to e.g. a poor energy conversion efficiency.

The operation cost of an LMES, which corresponds mainly to the cost of the energy resources bought from external suppliers to power the installed conversion devices, is however challenging to compute. Namely, the energy demands are highly variable and display a daily, weekly and yearly seasonality together with random variations. The price of the energy resources to be bought is also time-varying. Moreover, due to the technical features of some of the conversion devices, the energy resource consumption of the system is not proportional to the produced energy. This implies that using an average or aggregate representation of the demand to estimate the operation cost may lead to a significant underestimation of its actual value. Thus, in order to accurately estimate it, a detailed schedule describing, on a hourly basis, the on/off status and the load allocation of each conversion device should be built for an horizon spanning a whole year.

Furthermore, an LMES is often built together with the district it will serve. This means that the forecasted yearly energy demands feature an upward trend over the years as the district is developed and new buildings progressively connect to the LMES. As a consequence, the design of an LMES should not be seen as a one-time decision but rather as a process in which investment decisions are made step by step, following the development of the district and the increase of the demand. This implies that a multi-phase strategic deployment plan should be built.

We thus need to simultaneously determine, for each investment phase of the deployment plan, the optimal system layout and the corresponding detailed operation plan ensuring that the customers' energy demands are satisfied at all time and computing an estimation of the operation cost. Therefore, the optimal design problem is also referred to as the simultaneous system design and operation optimization problem.

1.2 Context

1.2.1 Industrial context

This PhD thesis results from a collaboration between EDF China, EDF R&D France and the University Paris Saclay.

In addition to its activities related to the large-scale generation of electricity in nuclear reactors or coal-fired power plants, EDF China is involved in several projects, sometimes together with Chinese external clients and partner companies, to build and operate LMESs in China. For instance, EDF China have been operating since 2016, together with the Chinese company Datang Group, a district heating network in the city of Sanmenxia [EDF22a]. The operation of the network is based on the heat recovered from two existing Datang power plants and 2x17 km backbone pipes have been built to connect the thermal power plants to the distribution networks in three industrial and business districts of the city. The heating area represented a total of 12 millions m^2 in 2020. Similarly, EDF China is currently developing, in partnership with ChangFeng Energy, a multi-energy system in the city of Sanya [EDF22b]. This system will supply chilled water for air conditioning and will contribute to the supply of sanitary hot water to a set of buildings (hotels, shopping centers and hospital) located along the Haitang Bay, a fast-developing tourist area. It is estimated that the LMES will reduce CO2 emission by 20% as compared to using individual and isolated systems, which represents a reduction of 70,000 tons of CO2 emission each year.

These LMES projects usually start with a call for tenders launched by cities wishing to develop an LMES in one of their districts. Companies such as EDF China and its competitors may reply by presenting a tender describing potential layouts for the system, together with their costs. This tender process usually comprises two steps. The first step involves a global selection of the technologies to be used in the system but usually relies on a gross estimate of the system operation cost assuming e.g. that the operating cost of each device is strictly proportional to the amount of output power. The second step consists in providing a detailed description of the future system comprising the number and capacity of each type of conversion and storage device to be installed in this system. In this second step, an estimation, as accurate as possible, of the future operation cost of the system is required. Overall, the goal of EDF China while presenting a tender is to select and size the devices installed in the system in such a way to simultaneously ensure that all the energy demands will be satisfied and that the total design and operation cost will be as low as possible.

Currently, the detailed tender to be prepared for the second step of this process is built manually by EDF China practitioners. Basically, the cost of each considered potential system layout is estimated through a set of computations carried out in Excel files. These computations rely on strong assumptions on the operation of the system, (e.g. devices are activated to produce energy in a predefined order and and energy storing and releasing is limited to some specific time periods). The system

layout to be presented in the tender is then selected through a manual trial-and-error process but there is no systematic exploration of the set of all potential layouts.

Clearly, there is a need for a decision-aid tool based on mathematical optimization to help EDF China practitioners identify the best system layout (together with the multi-phase deployment plan) to make sure that the LMES projects developed will be profitable.

EDF R&D has a strong experience in developing mathematical-programming based decision-aid to help in the strategic and daily management of energy systems. For instance, EDF Lab Chatou developed a software called Clevery to help in the daily operation of smart micro-grids and local energy systems. This system is able to build optimal operation schedules for a large set of small-scale systems whose layout is already known. It however does not allow to make design decisions for such systems.

Thus, EDF currently has no decision-aid tool dedicated to the long-term design of LMESs. Our work in this PhD thesis aims at developing such a tool.

1.2.2 Academic context

As discussed in the previous subsection, optimally designing an LMES is a very challenging problem for practitioners which may be overwhelmed by the exponential number of design possibilities to be considered and compared. The difficulty experienced by practitioners is supported theoretically by the fact that the LMES design problem was recently proved to be NP-hard in [GCW19], even when there is a single type of conversion device, no storage device and a single time-step to represent the demand at the operational level.

[KV21] provided a recent state of the art on the modeling and solution approaches proposed to design LMES. They highlighted the fact that mixed-integer linear programming (MILP) was one of the most used optimization techniques, probably thanks to its high flexibility in terms of problem modeling and to the availability of mature off-the-shelf commercial MILP solvers. The literature review focused on MILP models and decomposition approaches for the LMES design problem, to be presented in Chapter 2, shows that the research is this field is currently very active and that much progress has been made over the last two decades.

However, there remains a significant gap between the current state of the art on the LMES design problem and the needs of our industrial partner.

Indeed, we first note that all the reviewed works consider that the system is built in a single construction phase while, in our practical problem, the LMES is to be built and extended progressively following a multi-phase deployment plan. Second, in most previously published works, the sizing of technologies is oversimplified. Namely, these works assume that, for each technology, there is a single piece of equipment to be installed and that the capacity of this device may be set to any value within a continuous range. However, in practice, we may install several devices corresponding to the same technology, the capacity of which should be selected among a set of discrete values corresponding to the models available in the catalog of the equipment manufacturer. Third, due to the high complexity of the problem, the assessment of the operation cost is usually carried out by introducing some simplifications on the functioning of the conversion devices, in particular with respect to their conversion efficiency, and by using a limited number of representative days.

In terms of solution approaches, most of the reviewed papers rely on a direct resolution of the problem by an MILP solver. This direct resolution is only possible at the expense of some strong simplifications on the actual problem. Moreover,

decomposition approaches exploiting the natural hierarchy between the design and operation decisions in the problem have been proposed. But their direct application on the problem under study here is not straightforward.

Clearly, some research is needed to extend the previously published works and to incorporate into the problem modeling the above-mentioned important practical features. Taking these features into account leads to an increase in the MILP size and consequently makes it harder to solve. Part of the work carried out in this thesis will thus be devoted to the development of efficient solution approaches for this problem.

1.3 Background on mixed-integer linear programming

In this work, we heavily rely on mathematical programming, more specifically on mixed-integer linear programming (MILP), to tackle the problem of optimally designing an LMES. We thus provide in this section a short background on MILP to ease the reading of the manuscript.

1.3.1 Definition

A mathematical programming problem is an optimization problem taking the following form [BBV04]:

$$\begin{array}{ll} \text{minimize} & f_0(\mathbf{x}) \\ \text{subject to} & f_i(\mathbf{x}) \leq b_i, \quad i = 1, \cdots, m \\ & x \in \mathbb{R}^n \end{array}$$

The vector $\mathbf{x} = (x_1, \cdots, x_n)$ represents the set of optimization or decision variables, the value of which is to be determined by solving the optimization problem. The function $f_0 : \mathbb{R}^n \to \mathbb{R}$ is the objective function and represents the performance criterion used to evaluate and compare the various solutions of the problem. There is a set of m inequalities defining the feasible space, i.e. the subset of vectors $\mathbf{x}$ which are considered as acceptable (feasible) solutions of the optimization problem. These set of inequalities is defined through constraint functions $f_i : \mathbb{R}^n \to \mathbb{R}, i = 1, \cdots, m$ and right-hand side values $b_1, \cdots, b_m$.

In case the objective function and all the constraint functions are linear functions of $\mathbf{x}$, the problem simplifies into a linear programming (LP) problem in the form:

$$\begin{array}{ll} \text{minimize} & \sum_{j=1}^{n} c_j x_j \\ \text{subject to} & \sum_{j=1}^{n} a_{i,j} x_j \leq b_i \quad i = 1, \cdots, m \\ & x \in \mathbb{R}^n \end{array}$$

Here, all input parameters c_j, $a_{i,j}$ and b_i are real numbers whose value is assumed to be known.

An LP problem can also be written in the matrix form, using the convention that the $\leq$ sign compares the two vectors located on each side element-wise:

$$\begin{array}{ll} \text{minimize} & \mathbf{c}^T \mathbf{x} \\ \text{subject to} & A\mathbf{x} \leq \mathbf{b} \\ & \mathbf{x} \in \mathbb{R}^n \end{array}$$

Linear programming is a mathematical optimization tool which is widely applied to design energy systems. This may be explained by the fact that it enables

to flexibly model a rather large range of design problems and that it leads to the formulation of mathematical programs which are relatively easy to solve. However, linear programming presents some drawbacks and limitations and may lead to recommending system designs which are sub-optimal or even not relevant in practice. In particular, the fact that the decision variables $\mathbf{x}$ may take any continuous value poses some modeling problems to represent e.g. decisions that should take integer values (such as the number of energy conversion units of a given type that should be installed in the system) or logical conditions (such as the fact that if a device is turned on, it should produce an amount of energy above its minimum production load). The optimal solution of an LP progam may involve components of $\mathbf{x}$ with a frational value and simply rounding this frational value up and down does not always provide a feasible, let alone a good, solution of the optimization problem (see e.g. [Wol20]).

We thus sometimes have to add constraints to the formulation stating that some decision variables should be binary or integer in the solution. If all the variables are restricted to be binary, we obtain a Binary Linear Programming (BLP) problem. If all the variables are restricted to be integer, the problem is an Integer Linear Programming (ILP) problem. If the formulation involves both real valued variables and integer/binary variables, the problem is called a Mixed Integer Linear Programming (MILP) problem. The variables taking resp. binary, integer or fractional values are often referred to as resp. binary, integer and continuous variables. Let p be the number of integer variables in the MILP problem with $0 < p < n$, the matrix form of an MILP problem takes the following form:

$$\begin{array}{ll} \text{minimize} & \mathbf{c}^T\mathbf{x} \\ \text{subject to} & A\mathbf{x} \leq \mathbf{b} \\ & \mathbf{x} \in \mathbb{Z}^p \times \mathbb{R}^{n-p} \end{array}$$

1.3.2 Formulation of a mixed-integer linear program

The formulation of a real-life optimization problem as an MILP problem, i.e. the transformation of an informal textual description of the problem into a mathematical model, can be done through e.g. the procedure described in Chapter 1 of [PW06]. This modeling relies on the definition of the indices, data, variables, constraints and objective function of the MILP problem.

Indices Similar objects are grouped into object classes represented by mathematical indices, allowing one to use indexed notation for data, variables, and constraints. For instance, the index y (resp. ϕ) may be used to represent a year within the set of years (resp. a phase within the set of investment phases) of the LMES lifetime. Similarly, the index $\mathbf{m}$ may be used to indicate a type of energy conversion device within the set of all considered energy conversion devices.

Data Data refer to the "known values" in the optimization problem. For instance, the installation price of an energy conversion unit is considered as an input parameter in our case, the value of which is assumed to be known before solving the optimization problem.

Variables Variables are the "unknown values" of an MILP model and represent the decisions to be made to define a solution to the optimization problem. When

formulating real-life problems, for the sake of clarity and readability, we usually do not use a single variable vector $\mathbf{x}$ to represent all the decisions to be made, but rather a set of multi-dimensional vectors referred to by different names. Thus, in our context, the integer decision variable $\mathbf{SD}_{\phi,\mathbf{m}}$ indicates the number of conversion and storage units of type $\mathbf{m}$ to be installed at the beginning of investment phase ϕ whereas $\mathbf{STO}_{\mathbf{t},\mathbf{m}}$ represents the amount of energy stored in the devices of type $\mathbf{m}$ at the beginning of the scheduling time-step $\mathbf{t}$. A solution of an optimization problem thus consists in an assignment of a (binary, integer or continuous) value to each decision variable involved in the problem modeling.

Constraints Constraints are mathematical relationships between the input data and the decision variables and take the form of inequalities or equalities. Recall that in order to obtain an MILP problem, these relationships should be linear expressions of the decision variables. A feasible solution of an MILP problem is a solution in which the values assigned to the decision variables satisfy all the constraints. A solution is said to be infeasible as soon as it violates one or several constraints. The set of all feasible solutions is called the feasible region or feasible space of the optimization problem. If the feasible region of a problem is empty, the optimization problem is said to be infeasible.

Objective function The objective function provides a way to evaluate or compare feasible solutions and to select the best or optimal solution among the feasible ones. Both its optimization direction and its mathematical expression should be clarified. The optimization direction can be either maximization or minimization: these directions are interchangeable by adding a negative sign to the objective function. As for the mathematical expression, in an MILP problem, it should be a linear expression of the decision variables. Finally, if there exists a sequence of feasible solutions in the feasible region whose objective values are unbounded, in other words if the objective function of the problem can take any arbitrarily 'good' value, the problem is referred to as an unbounded problem. In some books, the objective value of a minimization problem is defined as $+\infty$ if it is unfeasible and as $-\infty$ if it is unbounded (and vice versa for maximimization problems).

1.3.3 Branch-and-bound algorithm

One of the reasons explaining the widespread use of mixed-integer linear programming to tackle optimization problem is the successful implementation of the branch-and-bound (B&B) algorithm in mathematical solvers. The B&B algorithm basically carries out an enumeration of all feasible solutions of an MILP, but uses the information provided by the linear programming relaxation of the original problem to cut out solutions proven to be non-optimal whenever it is possible. Note that using a B&B algorithm does not guarantee to find the optimal solution of any MILP in a reasonable time. However, as compared with other resolution strategies, it provides a relatively efficient solving routine for a large class of MILPs. It is in any case a fundamental cornerstone and a good starting point for solving MILPs.

Before introducing the details of algorithm, we will first explain a trivial but important proposition for MILPs.

Proposition 1.3.1. *Let* $IP_1 := \min\{\mathbf{c}^T\mathbf{x} : \mathbf{x} \in F_1\}$ *and* $IP_2 := \min\{\mathbf{c}^T\mathbf{x} : \mathbf{x} \in F_2\}$ *be two MILP problems. We denote by* F_1 *and* F_2 *the feasible regions of* IP_1 *and* IP_2*. Let* z_1^* *and* z_2^* *denote respectively the optimal objective value of* IP_1 *and* IP_2*. If* $F_1 \subseteq F_2$*, then* $z_1^* \geq z_2^*$*.*

Proof. The proof of this proposition is straightforward. If the optimal solution of IP_1 exists and can be found at point $\mathbf{x}_1^* \in F_1$, then by inclusion, $\mathbf{x}_1^*$ is also in F_2 and the optimal objective value of P2 is at most z_1^*. Moreover, if we adopt the convention that the optimal value of an infeasible problem is $+\infty$ and the one of unbounded problem $-\infty$, this proposition still holds. If IP_2 is infeasible, then F_2 is empty, which implies that F_1 is also empty and IP_1 also infeasible. On the other hand, if IP_1 is unbounded, then there exists a sequence of $\{\mathbf{x}_n\}_n$ in F_1 whose objective value is unbounded, i.e., $\liminf_{n\to\infty} \mathbf{c}^T\mathbf{x} = -\infty$. By inclusion, this sequence is also in F_2, hence the problem IP_2 is also unbounded. □

We then consider an MILP problem of the form $IP := min\{\mathbf{c}^T\mathbf{x} : \mathbf{x} \in F\}$, where $F = P_0 \cap \mathbb{Z}^p \times \mathbb{R}^{n-p}$ and $P_0 = \{\mathbf{x} \in \mathbb{R}^n : A\mathbf{x} \leq \mathbf{b}\}$. P_0 is the convex set obtained when removing the integrality constraints on the variables of IP. Problem $LP_0 := min\{\mathbf{c}^T\mathbf{x} : \mathbf{x} \in P_0\}$ is a linear program called the linear programming (LP) relaxation of the original MILP problem.

The B&B algorithm starts with the resolution of LP_0. A solution of the LP relaxation is said to be integer feasible if it is feasible for the original MILP problem. There are three possible cases for the optimal solution of LP_0: it may be infeasible, non-integer feasible or integer feasible. First, if it is infeasible, from Proposition 1.3.1, we deduce that IP is also infeasible. Second, if it is integer feasible, we can deduce from Proposition 1.3.1 that the solution of LP_0 is the optimal solution of IP. Third, if it is non-integer feasible, i.e., if it contains non-integral values for some integer variables, by Proposition 1.3.1, the optimal solution of LP_0 provides a lower bound of the optimal solution of IP.

In the first two cases, the algorithm may stop, having either proved that the MILP is infeasible or found an optimal integer solution. Let us now focus on the third case. We select one of the integer variables having a non-integral value in the LP relaxation and denote by x_1 this integer variable and by $\hat{x}_1 \notin \mathbb{Z}$ its current value. We create two new LP problems: LP_1 (resp. LP_2) is obtained by adding constraint $x_1 \geq \lceil \hat{x}_1 \rceil$ (resp. $x_1 \leq \lfloor \hat{x}_1 \rfloor$) to the LP relaxation. The LP relaxation together with these two generated LP problems form a binary tree, which is usually referred to as the branch-and-bound search tree, and the operation of generating two new LP problems is called branching. The root node of the tree, indexed by 0, corresponds to Problem LP_0 whereas its immediate children, Node 1 and Node 2, are associated to the two LP problems LP_1 and LP_2 created by branching. Let P_1 and P_2 denote the feasible region respectively of these two new LP problems. P_1 and P_2 are two disjoint subsets of the feasible region of LP_0, i.e., $P_1 \cup P_2 \subseteq P_0$ and $P_1 \cap P_2 = \emptyset$. Moreover, the corresponding integer sets $F_1 = P_1 \cap \mathbb{Z}^p \times \mathbb{R}^{n-p}$ and $F_2 = P_2 \cap \mathbb{Z}^p \times \mathbb{R}^{n-p}$ form a partition of the feasible set F of IP, i.e., $F_1 \cup F_2 = F$ and $F_1 \cap F_2 = \emptyset$. So the optimal solution of IP, if it exists, is either in F_1 or in F_2.

We thus continue searching for the optimal solution of IP by repeating the process described above. At each iteration, we select an active node, branch from it and solve the LP problems corresponding to its newly created child nodes. For each child node, we again have three cases. If the solution found is not feasible, we prune the node and will not consider branching from it in the latter iterations. If the solution found is integer feasible, it means that it is a feasible solution of IP. We refer to the best integer feasible solution of IP found so far as the incumbent: each time a new feasible integer solution is found, its value is compared with the one of the current incumbent and if this value is better, the incumbent is updated. If the child node returns a non-integer feasible solution, we add it to the list of active nodes and start a new iteration by selecting another active node.

The B&B tree thus progressively grows with each branching operation. However, as such, the B&B algorithm is equivalent to a trivial enumeration of all integer solutions in the feasible region. In order to be more computationally efficient, we can decide to prune nodes corresponding to non-integer feasible solutions by comparing the objective value of the associated LP problem with the one of the current incumbent. If the objective value of the associated LP problem is worst (i.e. larger or smaller depending on the optimization sense) than the one of the current incumbent, we can prune this node and stop branching from it. Namely, all integer feasible solutions of IP located in the feasible sub-region of IP defined by the branching decisions made up to this node, if they exist, cannot be better than the current incumbent and thus cannot be the optimal solution. This directly relies on Proposition 1.3.1.

Proposition 1.3.2. *In the B&B search tree of a minimization (resp. maximization) problem, the objective value of a parent node is always smaller (resp. greater) than the objective value of its child nodes.*

Proof. The proof is also straightforward, since the feasible region of a child node is always a subset of the one of its parent node and from the Proposition 1.3.1, we show that this proposition is true. □

The B&B algorithm terminates when the set of active nodes is empty, and the incumbent is the optimal solution of the original IP problem.

Algorithm 1 provides a detailed description of the branch-and-bound algorithm than can be used to solve for a minimization problem.

1.3.4 Branch-and-cut algorithm

The computational efficiency of a branch-and-bound algorithm heavily depends on the quality of the bound computed at each node of the search tree. In a nutshell, the better this bound, the sooner a branch can be discarded and the less nodes have to be explored before the algorithm converges. In the context of mixed-integer linear programming, this bound is most often obtained by solving the linear programming relaxation of the problem. The main advantage of this is that linear programs can be very efficiently solved by the simplex algorithm. However, in many cases, the linear programming bound is of poor quality, which negatively impacts the performance of the branch-and-bound algorithm. In order to solve this issue, branch-and-bound algorithms can be coupled with a cutting-plane generation approach, giving rise to a branch-and-cut algorithm. Basically, a cutting-plane generation approach aims at improving the quality of the linear programming bound by adding a set of linear inequalities to the problem formulation. These inequalities are chosen so as to cut away non-integer solutions that would otherwise be solutions of the linear relaxation and to improve the value provided by the linear relaxation by restricting its feasible space. The problem of finding a cut separating a non-integer solution from the feasible space of the linear relaxation of the MILP is called the separation problem. The branch-and-cut algorithms embedded in MILP solvers use generic cuts, i.e. cuts that may apply to any MILP or at least to a wide range of MILPs, such as Gomory fractional cuts, clique cuts or mixed-integer rounding cuts.

Thanks among others to the use of branch-and-cut algorithms, MILP solvers have undergone tremendous progress over the last thirty years. For instance, [Bix12] reported that the machine-independent computational performance of CPLEX solver improved by a factor of 29000 between the 1.2 version of the software released in 1991 and the 11.0 version released in 2007. Yet, despite this progress, there still are

Algorithm 1: Branch and Bound Algorithm for a minimization problem

Data: The MILP formulation of the problem
Result: The optimal solution and the minimum objective value

Solve the LP relaxation LP_0 of the original MILP problem;
if *LP_0 is infeasible* **then**
  return the original problem is infeasible;
else
  if *the solution of LP_0 is integer feasible* **then**
    return the solution of LP_0 as the optimal solution and the objective value of LP_0 as the minimum objective value ;
  else
    Create the root node Node 0 of the search tree and add it to the list of active nodes;
  end
end
Set Z^{best} to positive infinity and $\mathbf{x}^{best}$ to be empty;
while *the list of active nodes is not empty* **do**
  Select an active node, Node i associated to problem LP_i, remove Node i from the list of active nodes;
  Select an integer variable with a fractional value in the solution $\mathbf{x}_i$: $x_j = \hat{x}_{j,i}$;
  Create two child nodes of Node i in the search tree and index them by l_i and r_i ;
  Associate to Node l_i an LP problem LP_{l_i} corresponding to problem LP_i in which the constraint $x_j \leq \lfloor \hat{x}_{j,i} \rfloor$ has been added to the formulation, and associate to Node r_i an LP problem LP_{r_i} corresponding to problem LP_i in which the constraint $x_j \geq \lceil \hat{x}_{j,i} \rceil$ has been added to the formulation;
  for *each newly created node $n \in \{l_i, r_i\}$* **do**
    Solve the corresponding problem LP_n;
    if *LP_n is infeasible* **then**
      Prune Node n;
    else
      Find the optimal solution $\mathbf{x}_n^*$ and the optimal objective value Z_n;
      if *$\mathbf{x}_n^*$ is not integer feasible* **then**
        if $Z_n \geq Z^{best}$ **then**
          Prune Node n;
        else
          Add Node n to the list of active nodes;
        end
      else
        if $Z_n < Z^{best}$ **then**
          Update the incumbent and set $\mathbf{x}^{best} \leftarrow \mathbf{x}_n^*$ and $Z^{best} \leftarrow Z_n$;
        end
      end
    end
  end
end
return $\mathbf{x}^{best}$ and Z^{best} ;

many MILPs whose direct resolution by a mathematical programming solver leads to prohibitive computation times. In particular, this arises when the MILP to be solved involves a large number of integer variables and constraints, as this is the case in the LMES optimal design problem.

1.4 Research objectives and main contributions

In this thesis, we aim to create a decision-aid tool based on mathematical optimization to help EDF practitioners at optimally designing an LMES. This tool should take as input data a list of available energy storage and conversion technologies, to be potentially installed in the LMES, together with a set of time series describing the predicted evolution of energy resources' price and availability and of the clients' energy demands during the expected lifetime of the system. The output should be a multi-phase deployment plan indicating the technologies to be used in the LMES together with the number and size of the conversion and storage devices of each technology to be installed in each construction phase. The obtained plan should ensure that the system will be able to satisfy the clients' energy demands at all time and that the total cost of the system, which includes its design and operation cost, is minimized.

The numerical tool to be developed should be flexible enough to design a broad class of LMESs and thus should take into account a wide range of energy resources, energy demands and energy conversion and storage technologies. Moreover, it should make it possible to size the selected technology in a practically relevant manner, i.e. by selecting the conversion and storage devices to be installed in the LMES among the discrete list of models available in the catalog of the equipment manufacturer rather than by simply fixing the capacity within a continuous range. Finally, as the operation cost makes up a large part of the total cost of an LMES, the tool should be able to accurately estimate this operation cost. A large number of typical and extreme days should thus be used in the mathematical optimization model to represent, as best as possible, the various conditions under which the system will be operated. Moreover, it will be necessary to take into account some practical constraints related to the conversion devices, in particular a non-zero minimum working load and non-linear performance curves. Finally, the tool should be able to provide deployment plans with a reasonable computational effort. Namely, even if the design of an LMES is a long-term strategic decision, the tool might not be adopted by its end-users if the computation time needed to find a deployment plan exceeds a few hours.

As can be seen from the description of the state of the art to be provided in Chapter 2, none of the available off-the-shelf decision-aid tools is able to meet all these needs. Moreover, to the best of our knowledge, no previously published academic work investigated a mathematical optimization-based approach for the LMES design problem while simultaneously considering all the above-mentioned practical features. Our contributions are thus threefold.

The first contribution pertains to the modeling of the LMES design problem as a mixed-integer linear program. First, we explicitly consider the fact that building an LMES is most often not accomplished in a single step but that it is rather a process involving multiple construction phases, each of which expanding over one or several years, to adjust to the long-term increase of the energy demand. To the best of our knowledge, this is the first time the LMES design problem is modeled as a multi-phase investment problem. Second, our model allows to choose the capacity

of the installed devices within a discrete list of available models. We thus not only decide whether a technology will be used or not in the LMES but also determine the number and capacity of the devices belonging to this technology that should be installed. Finally, to estimate the operation cost of the system as accurately as possible, we build detailed daily operation schedules using hourly time steps. Each of these operation schedules consider several realistic complicating features such as the partial load efficiency and the minimum working load of energy conversion devices. Clearly, taking into account all these aspects of the problem leads to the formulation of a huge mixed-integer linear program which as such is computationally intractable. We thus propose a modeling approach aiming at reducing the size of the obtained mathematical program and at removing the non-linearities. This approach relies on the use of a limited number of investment phases to make design decisions and on the careful selection of a rather large number of disjoint representative days to estimate the operation costs. Moreover, in order to obtain a mixed-integer linear program, we use a piece-wise linear approximation of the energy conversion performance curves in case these ones are non-linear. Finally, we propose to exploit the convexity of these performance curves to build aggregate operation schedules, i.e. schedules describing the number of active devices, the energy consumption and the energy production of each set of identical devices, rather than detailed operation schedules describing the on/off status, the energy consumption and the energy production of each individual device. This enables us to reduce to a significant extent the number of decision variables needed in the MILP formulation to build these operation schedules.

The second contribution consists in the application and extension of two decomposition approaches to efficiently solve the large-scale MILP problem. Both approaches exploit the natural hierarchy between the design decision variables (used to build the deployment plan) and the operation decision variables (used to build the operation schedules). This hierarchy translates into a special structure of the mathematical problem. Namely, once the design decisions determining the system layout at each investment phase are fixed, the problem decomposes into a set of independent operation sub-problems, one for each considered representative day, aiming at building an optimal operation schedule for this day under the current system layout. This special structure is exploited by the hierarchical decomposition method previously published in [Yok+15] but our numerical results show that a direct application of this method on our multi-phase problem is not computationally efficient. We thus extend the algorithm of [Yok+15] to improve its efficiency at solving our multi-phase problem. This is achieved in particular through the addition of single-phase no-good cuts into the upper-level problem to forbid deployment plans using a system layout found to be infeasible for a certain investment phase, and through the recording of the solution value of previously solved operation sub-problems to avoid repetitive calculations. Furthermore, we develop a new decomposition approach for the problem which can be seen as a generalized Benders' decomposition algorithm. Note that the classical Benders' decomposition algorithm requires that the second-stage sub-problems display a strong duality property, which is not the case here since the operation scheduling sub-problems include discrete variables. We thus customize the framework presented in [BR21] to extend the classical Benders' decomposition algorithm to a larger class of optimization problems. The proposed generalized Benders' decomposition algorithm relies on a new set of feasibility and optimality cuts. These cuts exploit the specific structure of the constraints coupling the first-stage and the second-stage decisions in the operation scheduling sub-problems and enable us to obtain an algorithm with a theoretically guaranteed finite and optimal

convergence.

The third contribution deals with the evaluation of the proposed modeling and solution approaches on instances based on three case studies corresponding to three real-life LMESs currently under construction by EDF in China. Two of these LMESs are district cooling systems (DCS) aiming at meeting a demand for cold. The third one is a district cooling and heating system aiming at meeting a demand for cold and heat: for this system, the installation of a trigeneration system, i.e. of a device capable of converting gas into electricity, cold and/or heat, is considered. For each of these three projects, the deployment plan includes multiple construction phases. Moreover, each system potentially involves conversion devices with non-linear performance curves and a non-zero minimum load rate. Our numerical results showed that the two proposed decomposition approaches significantly outperform, in terms of solution quality and computation time, both an MILP solver directly solving the MILP formulation and the previously published hierarchical decomposition method at solving the problem. In particular, the proposed generalized Benders' decomposition algorithm was able to solve all the considered instances to optimality within two hours of computation. However, solving the MILP problem to optimality does not guarantee that the obtained deployment plan is optimal with respect to the initial optimization problem as some approximations were made during the problem modeling. We thus carried out a post-optimization simulation study to evaluate the quality of the deployment plans obtained for the two DCS projects. Overall, this study indicates that, provided the number of representative days used in the model is large enough, the design decisions do not depend on the subset of selected representative days and the operation cost is estimated with a good level of accuracy by the optimization model so that the obtained deployment plans may be confidently recommended for a practical implementation.

This remainder of this manuscript is organized as follows. In Chapter 2, we review the academic works closely related to ours, focusing in particular on the ones using an MILP model and/or a decomposition-based solution approach. In Chapter 3, we provide a detailed description of the optimization problem. We start by introducing the main components of an LMES: the commodities and the energy conversion and storage technologies. We then explain a number of additional aspects to be considered when designing an LMES and formally state the resulting optimization problem in terms of input data, decisions, constraints and objective function. Chapter 4 is devoted to the problem modeling and to its mathematical formulation as an MILP. We present among others the modeling choices and assumptions we made to obtain an MILP of tractable size. We then provide a detailed description of the three case studies used to create instances of the problem. We finally present the numerical results obtained when attempting to directly solve these instances with CPLEX 12.8 solver using the proposed MILP formulation. In Chapter 5, we extend the hierarchical decomposition approach previously published in [Yok+15] to make it more efficient at solving the large-scale MILP model formulated in Chapter 4. This approach is applied on the three case studies. Our computational results show a clear improvement of the computational efficiency on the instances related to the two DCS projects. However, out-of-memory issues were still encountered while trying to solve the instances related to the trigeneration project. In Chapter 6, a new generalized Benders' decomposition algorithm is proposed. This new generalized Benders' decomposition algorithm relies on the specific structure of the constraints coupling the first-stage and the second-stage decisions in the operation scheduling sub-problems and extends the scope of the traditional Benders' decomposition to the case where sub-problems are MILP problems. The results of our computational

experiments, carried out on instances based on our three case studies, are reported. They show that the generalized Benders' decomposition algorithm clearly outperforms all the other considered algorithms at solving the problem. Moreover, the outcome of a post-optimization simulation study carried out to evaluate the quality of the deployment plans obtained for the two DCS projects is discussed. Finally, in Chapter 7, we provide a general conclusion and discuss several potential research perspectives.

Chapter 2

State-of-the-art on the LMES design problem

2.1 Introduction

We provide in this chapter an overview of the works closely related to ours. We first discuss in Section 2.2 papers proposing a mathematical-programming based model to optimize the design of an LMES. We also present several already existing numerical MILP-based tools that may assist in managing and designing an LMES. We then focus in Section 2.3 on the literature dealing with solution algorithms relying on a decomposition of the main problem into a strategic design master problem and one or several operation scheduling sub-problems.

2.2 Mathematical programming models

Several modeling and optimization techniques have been investigated to tackle the problem of optimally designing an LMES: see e.g. [KV21] for a recent review on existing approaches. Among all these techniques, mixed-integer linear programming appears to be one of the most widely studied. This may be explained by the fact that MILP offers a high flexibility for modeling the problem and that several off-the-shelf mature commercial solvers (e.g. CPLEX [IBM21], Gurobi [Gur21] and Xpress [FIC21]) capable of solving large size MILPs to optimality with a reasonable computational effort are now available.

This section thus aims at presenting the current state of the art on MILP models for the LMES design problem. Our work initially focused on the optimal design of district cooling systems (DCS) supplying a single type of energy (cold) and was later generalized to a broader class of local multi-energy systems supplying energy under different forms (cold, heat, electricity...). In what follows, we thus first review the works related to the design of a DCS before broadening the discussion to papers investigating the design of more general LMESs.

2.2.1 Optimal design of district cooling systems

DCSs have been the subject of many research works. We refer the reader to the two recent literature reviews provided by [EA19] and [Gan+16] for a general introduction on this domain and concentrate on papers dealing with a mathematical-programming based approach to optimally design a central chiller plant.

As mentioned among others by [EA19], most previously published works focus on optimizing the DCS distribution network configuration. Thus, we could find only a limited number of papers discussing the design of the central chiller plant. [Sö07]

studies the optimal design of a district cooling network in an urban region. His problem consists in simultaneously locating and sizing a set of electric-powered compression chillers and of cold water storage sites and in selecting cooling pipeline routes to distribute the chilled water to the customers' buildings. The optimization problem seeks to minimize the total cost of building the system and operating it, i.e. of running the installed chillers and pumping the water in the pipes. A set of 8 twelve-hour periods are used to represent the seasonal and daily (night/day) variations of the cooling demand. [KH15] considers a similar problem and uses an oriented graph to represent the potential DCS. This graph comprises a single supply node representing the central chiller plant whose location is assumed to be known, a set of demand nodes representing the customers' buildings and a set of connection nodes corresponding to pipe junctions. Their design problem involves choosing the chiller plant capacity and the storage tank capacity among a discrete set of options, in selecting the arcs of the networks where distribution pipes should be built and in sizing these pipes. [ANKH19] extends the work of [KH15] to the case where chillers may be installed at different potential supply nodes in the graph. Both works assume that the cooling demand of each connected building is one-day periodic and stationary and use a single typical day divided into 4 to 24 scheduling periods to evaluate the operation cost of the system. [Alg+20] addresses the problem of optimally designing a DCS based on an absorption chiller using the hot water obtained from a set of solar collectors to produce chilled water. The design decisions involve determining the type and area of the solar collectors, sizing the absorption chiller and the auxiliary boiler and choosing the capacity of the cold and hot thermal storage tanks. The operation cost of the system is estimated by considering 12 typical days (one per each month in the year) divided into 24 scheduling periods. In all the above-mentioned works, the obtained optimization problem is formulated as a mixed-integer linear program and directly solved by a mathematical programming solver.

2.2.2 Optimal design of local multi-energy systems

Unlike DCSs which supply energy under a single form (cold), an LMES involves multiple forms of energy and a wide variety of conversion and storage devices, which significantly complicates the design problem. Several articles ([Yok+15; WHY21; Zho+13b]) point out that directly solving to optimality the LMES design problem with an MILP solver might not be possible in practice. These numerical findings are backed by the recent theoretical results presented in [GCW19]. The optimal design problem, which is referred to as the synthesis problem in this paper, is proven to be NP-hard even if there is a single type of conversion device, no storage device and a single period to represent the demand at the operational level.

In what follows, we review papers dealing with the design of the central power plant of an LMES. Papers dealing with the structure of the pipeline networks, such as [CEA+15], [Meh+13] or [WMF07], are out of the scope of this study. Tables 2.1 and 2.2 provide a summarized description of a list of recent papers dealing with the optimal design problem of an LMES. Each work is described according to five main features. The first three features, which are related to the problem modeling, are presented in Table 2.1. We thus consider the structure of the LMES (i.e., the available energy resources, the form of the energy demands to be met, the available conversion and storage technologies), the temporal granularity (length of a scheduling time-step and number of representative days) used to estimate the operation cost and the representation of the conversion efficiency of the devices (constant or

partial-load efficiency). The last two features, which are related to the mathematical formulation and the solution approach, are presented in Table 2.2. We thus report for each paper the type (continuous, integer or binary) of the design and operation variables, the characteristics of the mathematical formulation (mixed-integer linear or non-linear programming) and the type of solution approach (direct resolution by a solver, evolutionary algorithm, Benders' decomposition...). Notice that in [Zho+13b] and [Ste+15], the proposed mathematical model can be used to design a wide range of LMESs and does not assume a specific structure of the system: this is indicated by noting "All" for the set of energy resources, energy demands and conversion technologies in Table 2.1.

Tables 2.1 and 2.2 show among others the wide variety in terms of structure of the LMESs investigated in the literature. In what follows, we review the papers while focusing mainly on the way the structure of the LMES is determined in the mathematical model and on three aspects which have a strong impact on the formulation of the operation sub-problems: the representation of the performance curves, the incorporation of an imposed minimum load for conversion devices and the presence of storage devices. The solution approaches will be compared and discussed in the next section.

We first compare the papers with respect to the way design decisions about the type and capacity of the installed devices are made: see column 'Design variables' in Table 2.2. We note that most papers use a combination of binary and continuous ('B+C') variables. The binary variables are used to decide whether a specific technology (heat pump, electric heater, gas-fired boiler, photo-voltaic panels...) or the device corresponding to this technology is selected or not. Continuous variables are used to determine, for each selected technology, the capacity of the installed device. Note that some papers [Zho+13b; UDT19; Ste+15; AFF07] use only continuous variables for the system design. However, sizing a technology, i.e. determining the capacity of the corresponding device installed in the system, using continuous variables is not relevant for many industrial case studies. Namely, in practice, the capacity of the installed devices cannot be set to any arbitrary value but rather has to be chosen within a set of discrete values. This set is defined by the catalog of available models provided by the equipment manufacturer. Moreover, using binary variables to decide whether a technology will be used or not in the LMES relies on the assumption that there is a single type of device available for this technology whereas there may be several types of devices, each one corresponding to a given capacity, for a given technology. This is why integer variables indicating the number of devices of each type (i.e. of each technology and each possible value of the capacity) to be installed should be used to determine the structure of an LMES. We found only four papers [Zho+13a; Ren+10; CSL11; Yok+15] using integer variables to determine the number of conversion devices to be installed in the system. Moreover, in [OHO16], a single integer variable is used to determine the number of solar thermal collector panels to be built, while the size of the electric heater is chosen within a continuous range.

The second important aspect when modeling the LMES design problem by MILP is the representation of the performance curves of the conversion devices: see column 'Conversion efficiency' in Table 2.1. Recall that these curves provide the amount of energy consumed by a device as a function of the energy produced. The conversion efficiency of a device may depend on many factors such as the ambient temperature (see e.g. the solar thermal collectors in [OHO16]) or the capacity of the device (see e.g. the gas turbine of the combined heat and power unit in [WMF07]). As long as these factors do not depend on the scheduling decisions, the conversion efficiency can be considered as constant, which results in linear performance curves.

However, for some types of device (e.g. electric chillers), the conversion efficiency depends on the load rate of the device, which is a scheduling decision, and most often, its partial-load efficiency is smaller than its full-load or nominal efficiency. This translates into non-linear performance curves, which are difficult to handle in the MILP formulation. Note that the vast majority of the reviewed papers assume a constant conversion efficiency and use linear performance curves. We could find only three exceptions. In [Zho+13a], the non-linear performance curves of the gas boiler, the gas engine and the absorption chiller are linearized using a piece-wise linear approximation approach. The fuel consumption of the combined heat and power (CHP) component in [Els+17] depends on both the load rate and the extraction valve opening rate: the authors use a triangle method [DLM10] to approximate this function by a two-dimensional piece-wise linear function. In [UDT19], the partial-load efficiency of a combined heat and power unit is explicitly considered in the model, leading to the formulation of an MINLP.

There is a third feature strongly impacting the problem modeling: the minimum load rate of the conversion devices. Namely, in some cases, the output power of a device cannot take any arbitrary value between zero and its maximum capacity. Instead, a device, when active, has to produce a minimum amount of power. As can be seen from the column 'Operation variables' of Table 2.2, most papers use binary variables to represent the on/off status in each scheduling time-step of the single piece of equipment installed for each technology. However, models using a binary variable for selecting a technology and a continuous variable for sizing it neglect the fact that there will in fact be multiple devices of this technology in the system and can therefore not optimize the detailed operation of these multiple devices. The simulation study presented in [Che+20] shows that the actual energy consumption of a DCS designed while explicitly building optimal schedules for each of the chillers involved in the system can be around 20% that the one of a DCS designed while using a simplified control strategy not taking into account the presence of multiple chillers in the system. Among the three papers using integer design variables to determine the number of devices of each type to be installed, we may distinguish two cases. [Zho+13a] builds a detailed operation schedule for each individual device thanks to the use of binary operation variables whereas [Ren+10] builds an aggregate operation schedule for the set of similar devices thanks to the use of integer operation variables. As for [CSL11], the authors' focus is mostly on the environmental impact of the system design and operation details, such as the minimum load rate of the conversion devices, are not considered in their model.

Finally, the last aspect to be discussed is the presence of energy storage devices in the LMES: see column 'Storage technology' in Table 2.1. When there is no storage device in the system, the only way to satisfy the energy demand in a given time-step is to produce it through a conversion device during this time-step. Therefore, once the system design is determined, each operation sub-problem consists in ensuring the demand satisfaction in a single time-step and is thus of limited size. In contrast, the presence of an energy storage device results in a coupling between the scheduling time-steps as the amount of energy stored in the device at the end of a time-step defines the amount of energy available in this device at the beginning of the following time-step. These means that the single time-step operation sub-problems mentioned above are linked together by inventory balance equations, leading to larger and more difficult to solve operation sub-problems. As shown in Table 2.1, about one third of the reviewed articles do not involve energy storage in their system. Among the articles considering energy storage, the most common method to assess the operation cost of the LMES consists in using representative days, each one typically covering

24 one-hour time-steps. These representative days are chosen to represent as best as possible the energy demand profile along the year and are assigned a weight corresponding to the number of actual days in the year they represent. [MEC16] and [Wak+19] both use a set of disjoint representative days, i.e. a set of representative days not linked to one another by inventory balance equations, and build a separate schedule for each of these days. Note that they consider that the entering energy inventory at the beginning of a representative day should be equal to the leaving energy inventory of the same day. This amounts to assuming that each representative day will be cyclically repeated during the year a number of times equal to its weight. [UDT19] consider inventory balance equations for all time-steps of the year and apply the rolling-horizon approach described in [Bis+19]. This approach solves a sequence of operation sub-problems spanning two consecutive days. At iteration k, an operation schedule is buit for two consecutive days, denoted by Day k and Day $k+1$ and the inventory level at the end of Day k is recorded. This level is used during iteration $k+1$ to define the inventory level at the beginning of the Day $k+1$ and build an operation schedule for Days $k+1$ and $k+2$. [Gab+18] also build an operation schedule spanning a whole year and taking into account all the inventory balance equations. However, they use an approximation enabling them to significantly reduce the number of binary scheduling variables and to obtain a computationally tractable problem. Namely, the authors propose to first build clusters of days. Then, when building the operation schedule for the whole year, they impose that the binary scheduling variables describing the on/off status of each device in each time-step should take the same value for all the days belonging to the same cluster. They however allow the continuous variables describing e.g. the output power of each device to vary from one day to the next in each cluster.

We end this section by a short discussion on the available decision-aid tools that may be used to optimize the cost of energy systems ([VBGS15; Man14; Lop+18]). Noe of them seems to be versatile enough to model and optimize all the aspects of an LMES that we wish to take into account. For example, the tools Balmorel [Bal21; Wie+18] and EnergyPLAN [Ene21]) can only optimize the operational cost. The tool DER-CAM ([DC21]) models an LMES by an MILP and can find an optimal strategy for the planning and operation of the system. It however does not allow to make design decisions and cannot take into account the fact that some devices may be operated under multiple modes (such as a dual-mode electric chiller producing either cold water or ice). Finally, eTransport ([BSW07]), another tool based on MILP, focuses on the topology structure of the system and the timing of the investments to be made. But its modeling of the operational functioning is not detailed enough to be used in our case as it considers neither the partial load efficiency, nor the minimum load rate, nor the multiple operation modes of energy conversion devices.

2.3 Decomposition-based solution approaches

Modeling the optimal design of local or district energy systems as a combinatorial optimization problem leads to the formulation of large-size mathematical programs which are very challenging to solve. Consequently, a wide variety of operation research-based solution approaches have been investigated for these problems. We focus here on the subset of approaches exploiting the natural hierarchy between the design and the operation decisions to decompose the optimization problem. More precisely, in the considered works, the optimization problem is split into a master problem determining the general structure of the energy system and one or several

	Energy resource	Energy supply	Conversion technology	Storage technology	Timestep & scale	Conversion effciency
[Els+17]	Fuel	E, H	CHP boiler	H	1 hour 3 rep. weeks	partial-load
[MEC16]	S, E, G, H	E, H, ES	CHP, PV, ST boiler	H	1 hour 12 rep. days	constant
[Zho+13b]	All	All	All	H, I, B	1 hour 4 rep. days	constant
[UDT19]	Fuel	E,H	CHP, boiler	H	1 hour 1 year (rolling horizon)	partial-load
[OHO16]	S, E	H	ST, EH	H	1 hour 1 year	constant
[Ste+15]	All	All	All	H, I, B	1 hour 24 rep. days	constant
[Zho+13a]	E, G	E, H, C	CCHP, boiler	H	1 hour 3 rep. days	partial-load
[Ren+10]	E, G, S, W	E, H, C, HW	CCHP, PV, boiler, FC, HP, WT	B	1 hour 2 rep. days	constant
[AB16]	E, S, Fuel	E, ES, H, C	CCHP, PV, boiler, EC	/	1 hour 2 rep. days	constant
[CSL11]	G, E	H, C	CCHP, EC, boiler	/	1 hour 24 rep. days	constant
[Gab+18]	G, E, S	E, H	PV, FC, boiler, EL	B, H, H_2	1 hour 3-72 rep. days	constant
[Wak+19]	G, E	E, H	CHP, HP	H	1 hour 12 rep. days	constant
[AFF07]	G, E	E, H, C, ES, HW	CHP, HP, boiler	/	1 hour 12 rep. days	constant
[YZX15]	G, E	E, H, C	CCHP, EC, boiler	/	2 hour 3 rep. days	constant
[CEA+15]	G, E	H, C	CCHP, Airco, boiler	H	1 hour 3 rep. days	constant
[Meh+13]	G, S	H, E	CHP, PV, boiler	H	1 hour 3 rep. days	constant
[WMF07]	G, E	H, ES, E,HW	CHP, HP, boiler	/	12 hour 1 rep. days	constant
[WHY21]	G, H, E	H, ES, E, C,HW	CCHP, GFAC, boiler	H	1 hour 3 rep. days	constant
[Yok+15]	G, E	E, H, C	CCHP, EC, boiler	/	1 hour 3 rep. days	constant
[Sch+18]	G, E, S	H, E, ES	ST, EH, PV, HP, CHP, boiler	H, B	1 hour 12 rep. days	constant
[Liu+20b]	G, S, W	E, H, C	CCHP, FC, EC, PV, HP, boiler	H_2, B, H, C	1 hour 4 rep. days	constant

Energy Type: S = solar radiation, E = electricity, G = natural gas, W = wind power, H = heating power, C = cooling power, ES = electricity sold to the grid, HW = hot water. Conversion Technology: CHP = combined heating and power, ST = solar thermal collector, EH = electric heater, CCHP = Combined cooling, heating and power, FC = fuel cell, HP = heat pump, WT = wind turbine, EC = electric chiller, EL = electrolyzer, Airco = air conditioning, GFAC = gas-fired absorption chiller. Storage Technology: H = heat storage, C = cooling storage, B = battery, H_2 = hydrogen.

TABLE 2.1: Recent articles on the optimal design problem of LMES

	Design variables	Operation variables	Formulation	Solution approach
[Els+17]	B+C	B+C	MINLP	EA+MILP
[MEC16]	B+C	B+C	MILP	Direct
[Zho+13b]	C	B+C	MINLP	EA+MILP
[UDT19]	C	B+C	MINLP	EA+MILP
[OHO16]	I+C	C	MILP	Iterative procedure
[Ste+15]	C	C	LP	DER-CAM
[Zho+13a]	I	B+C	MILP	Direct
[Ren+10]	I	I+C	MILP	Direct
[AB16]	B+C	B+C	MILP	Direct
[CSL11]	I	B+C	MILP	Direct
[Gab+18]	B+C	B+C	MILP	Direct
[Wak+19]	B+C	B+C	MILP	AIS+MILP
[AFF07]	C	B+C	MINLP	Direct
[YZX15]	I+C	I+C	MILP	Direct
[CEA+15]	B+C	C	MILP	Direct
[Meh+13]	B+C	C	MILP	Direct
[WMF07]	B+C	C	MINLP	EA+LP
[WHY21]	B+C	B+C	MILP	BD+DW (sub-optimal)
[Yok+15]	I	B+C	MILP	Hierarchical decomposition
[Sch+18]	B+C	B+C	MILP	DW
[Liu+20b]	B+C	B+C	MILP	Direct

Variable type: I = integer, B = binary, C = continuous. Solution approach: EA = evolutionary algorithm, AIS = artificial immune system, BD = Benders decomposition, DW = Danzig-Wolfe decomposition, DER-CAM = a numerical tool.

TABLE 2.2: Recent articles on the optimal design problem of LMES

sub-problems seeking to optimize its daily operation management under various demand and price conditions.

A first set of approaches is based on evolutionary algorithms: see among others [WMF07], [Els+17], [Zho+13b] and [UDT19] in Table 2.2, as well as [FBM14] which is not in the table. These works consider a master problem in which all the operational decision variables and constraints have been removed and explore the set of feasible design decisions through a scatter search algorithm [UDT19; Zho+13b; Els+17] or a multi-objective evolutionary algorithm [WMF07; FBM14]. The operation cost of each potential design solution considered during this exploration is evaluated by solving a set of scheduling sub-problems using a mixed-integer linear programming solver.

Benders' decomposition algorithms such as the ones investigated e.g. by [FS16] and [LB15] also consider master problems comprising only design decisions. The impact of these decisions on the operation scheduling sub-problems is taken into account in the master problem through a set of feasibility and optimality cuts. Note however that the generation of these cuts rely on the assumption that the scheduling problems are linear or convex programs involving neither binary nor integer variables. This implies that these algorithms may not straightforwardly extended to incorporate features such as a minimum working load or multiple operating modes for the conversion devices. [WHY21] also proposes to apply Benders' decomposition to a design problem in which sub-problems contains binary variables. The feasibility cuts are added in the form of no-good cuts but the optimality cuts are generated by using the dual solution of the linear relaxation of the scheduling sub-problems. This means that the operation cost is under-estimated in the master problem and that the obtained design solution is sub-optimal.

A decomposition approach based on the selection of representative days is proposed in [Tso+20]. The system design is determined by solving a MILP problem with a small number of representative days, and then, its corresponding operation cost over the full time horizon is computed. If the gap between the total cost computed with representative days and the total cost over the full time horizon is larger than the threshold, the algorithm increases the number of representative days and repeats the whole process.

Finally, several works consider master problems in which some or all of the operational decision variables and constraints are kept within the master problem but only (relaxed) continuous operation variables are considered. In this way, [IG98] proposes a bi-level decomposition method. At the upper level, they formulate a master problem corresponding to the initial optimization problem in which all the binary operation variables have been removed but the continuous ones are kept. This master problem selects the conversion devices to be installed. Each time a potential system design is found, a set of lower level sub-problems is solved, each one taking as input the current system design and seeking to build optimal single-period schedules. The authors introduce the concept of design cuts to tighten the gap between the solutions obtained at both levels. [Yok+15] also presents a bi-level decomposition method. The upper level problem is a relaxed version of the initial optimization problem, in which all operation integer variables are kept but relaxed to be continuous. This problem is solved by a Branch & Cut algorithm. Each time a potential incumbent solution is found during the tree search, a set of single-period independent scheduling sub-problems is solved to check the feasibility and accurately compute the operation cost of the current system design.

2.4 Conclusion

We provided in this section an overview of the literature closely related to our problem. This overview enabled us to identify a significant gap between the current state of the art on the LMES design problem and our industrial problem.

Namely, we first note that all reviewed works consider a single construction phase while, in our practical problem, the LMES is to be built and extended progressively following a multi-phase deployment plan. Second, in most previously published works, the sizing of technologies is oversimplified. Namely, these works assume that, for each technology, there is a single piece of equipment to be installed and that the capacity of this device may be set to any value within a continuous range. As mentioned above, in practice, we may install several devices corresponding to the same technology, the capacity of which should be selected among a set of discrete values corresponding to the models available in the catalog of the equipment manufacturer. Third, due to the high complexity of the problem, the assessment of the operation cost is usually carried out by introducing some simplifications on the functioning of the conversion devices, in particular with respect to their conversion efficiency, and by using a limited number of representative days. In terms of solution approach, most of the reviewed papers solve the MILP problem directly by a mathematical solver. Some articles propose decomposition algorithms exploiting the hierarchical structure of the LMES but only a handful of these algorithms has a guaranteed finite and optimal convergence.

This is why, in this thesis, we aim to extend the current state of the art in order to meet the needs of our industrial partner. We thus seek to develop an MILP model of the LMES design problem which can simultaneous take into account (i) multiple construction phases, (ii) a wide range of energy resources, energy demand, energy conversion and storage devices, (iii) a realistic choice of these devices among the models available at the manufacturer, (iv) the non-linearity of the performance curves of some conversion devices and (v) a number of representative days sufficiently large to obtain a good estimation of the actual operation cost of the system. In order to solve the resulting large-size combinatorial optimization problem to optimality within an acceptable computation time, we will investigate and extend decomposition algorithms exploiting the bi-level nature of the problem.

Chapter 3

Problem description

3.1 Introduction

This chapter provides a detailed description of the optimization problem under study and introduces the various elements to be taken into account when modeling it as a mathematical program.

In Section 3.2, we first introduce the key components of a generic LMES and provide a short technical description for the main energy conversion and storage technologies encountered in LMESs. We then further explain in Section 3.3 a number of additional aspects to be considered when designing an LMES. In Section 3.4, we formally state the optimization problem by defining the problem input data, the decisions to be made, the main constraints to be respected and the performance criterion to be optimized. Section 3.5 finally describes three application case studies corresponding to LMESs EDF is currently developing in China.

3.2 Components of an LMES

An LMES usually comprises one or several local energy plants, producing energy under multiple forms, and a distribution network to distribute this energy to the buildings connected to the system. In the present work, we will consider the case where the LMES involves a single local energy plant and will focus on choosing and sizing the conversion and storage devices in this energy plant. The problem of optimally designing the distribution network is thus outside of the scope of this PhD thesis. This corresponds to situations, often encountered in practice, in which the LMES system to be designed will use local pipeline networks which either already exist or will be built by a partner company. Such situations were considered e.g. in [AFF07; GCW19; Sch+18].

The central energy plant of an LMES involves a large number of various components and the complexity of its structure makes the design problem difficult to solve. However, not all the components have the same importance when making long-term decisions on the system layout. Our focus will be on choosing and sizing the energy conversion and storage devices so that the structure of the central energy plant can be significantly simplified without a big loss of accuracy. Thus, auxiliary components (such as pipes, valves, pumps and etc.) will not be considered in our problem modeling. Namely, the cost of these auxiliary elements is usually either fixed or proportional to the cost of the selected conversion and storage devices. It can thus be incorporated in the construction cost of these devices without major consequences on the long-term design decisions.

Therefore, we will consider two main classes of components of the LMES: commodities and technologies. Commodities refer to the various forms of energy involved in the system: electricity, cooling power, heat... Technologies are the processes which transform one or several types of commodities into one or several other types of commodities.

In what follows, we discuss in more detail the commodities and technologies to be considered.

3.2.1 Commodities

We first classify the available commodities into three sub-types according their role in the LMES: resource commodities, supply commodities and intermediate commodities.

A resource commodity refers to a form of energy consumed by the system (e.g electricity from the national grid, natural gas and solar radiation). It is associated to a buying price, and/or to a limit on the available instantaneous power.

A supply commodity refers to a form of energy produced by the LMES. It can be used to satisfy the customers' demand (e.g. cooling power, heating power, electricity), in which case it is associated to a time-varying demand that should be satisfied at all time. In some cases, a supply commodity can also be sold (e.g. generated electricity sold to the national grid) and it is associated with a selling price.

Finally, an intermediate commodity refers to a form of energy produced and consumed within the system (e.g. hot water for storage, locally generated electricity, ice, exhausted gas). Intermediate commodities, as they exist only within the system, should be balanced between production and consumption.

Besides this classification based on their role in the LMES, commodities can also be described according to their nature. We introduce in what follows several types of commodities commonly used in LMESs.

Electricity An LMES generally consumes electricity coming either from the national grid or from a local generation. According to our classification, the electricity bought from the national grid is a resource commodity for which a time-varying price per unit of consumed electricity is charged. In some cases, there is also a limit (determined e.g. by a contract with the electricity supplier) on the electric instantaneous power that the LMES is allowed to consume. Electricity can also be generated locally by Combined Heat and Power (CHP) units or photo-voltaic (PV) panels. This locally generated electricity can be seen as an intermediate commodity consumed within the LMES by some conversion devices and/or as a supply commodity sold to the national grid or used to satisfy the customers' demand for electricity.

Natural gas Natural gas is a resource commodity for an LMES. The unit buying price of natural gas is usually much more stable than the one of electricity. The flame temperature of natural gas is around $1980^{\circ}C$, which means that its combustion generates a lot of heat. Natural gas can thus be used to supply an internal combustion engine generating electricity and/or to provide direct heating. When it is used to generate electric power, the exhaust heat can be collected by heat exchangers or absorption chillers to produce heating or cooling power. This technique is widely used to improve the energy efficiency of natural gas combustion.

Chilled water The thermal energy produced in the central plant of an LMES is transferred to the customers via a pipeline network, in which the circulating medium is usually water. Chilled water has a temperature lower than the one of the ambient environment and is used among others to provide air conditioning to the customers in summer. Chilled water is obtained either by cooling down water through devices such as chillers and heat pumps or by melting ice held in a thermal storage tank. Chilled water is transferred to the customers' buildings and is thus considered as a supply commodity with a given time-varying demand to be satisfied.

Ice Similar to chilled water, ice is produced to provide cooling power to the customers. Ice is obtained by cooling down water below its melting point through specific devices such as dual-mode chillers. Once produced, ice is usually stored for a short period of time (typically a few hours) in a thermal storage tank. When necessary, the LMES operator can melt the stored ice to obtain chilled water to be transferred to the customers' buildings. Ice is thus a way of storing thermal energy and is not used directly to satisfy the customers' demand. In an LMES, it is considered as an intermediate commodity.

Hot water In winter, hot water is circulated in the pipeline network to satisfy the customers' heating demand. The heating energy carried by hot water can be obtained from the combustion of gas (e.g. in a CHP unit or a boiler) or from an electricity-driven thermodynamic cycle (e.g. in an air source heat pump). The hot water may be directly circulated to the customers' side to provide heating energy or be stored for a short period of time in a thermal storage tank. Hot water is thus both an intermediate and a supply commodity in an LMES.

Solar radiation Solar radiation is seen as a clean and sustainable source of energy but it has some practical drawbacks. First, it is of course only available at daytime. Second, its availability strongly depends on the weather conditions and on the season. In practice, researchers use local historical solar radiation records to forecast the future availability of solar energy: see e.g. [Alg+20; Tso+20]. Solar radiation is thus considered as a resource commodity which is free (the unit buying price is zero) but has a strongly time-varying limited availability. Solar radiation can be converted to electricity by photo-voltaic panels and/or to hot water by solar collectors.

3.2.2 Technologies

The energy conversion and storage technologies that may be used in an LMES differ among others with respect to their technical functioning and to the consumed and produced commodities. Due to the high variety of available technologies, it is not possible to provide here an exhaustive description. We thus only briefly present some technologies commonly used in LMES, focusing in particular on the ones encountered in our application case studies. For each considered technology, we present among others the consumed commodities, the produced commodities, the working load range, the conversion efficiency and the sizing pattern.

Electric chillers An electric chiller is a machine converting electricity into cooling power. This cooling power can be in the form of chilled water, which circulates through a pipeline network to absorb the heat in the buildings served by the LMES, or in the form of ice, which is stored in a thermal storage tank for a few hours and

melted into chilled water afterwards. Not all chillers produce ice. We thus refer to chillers producing only chilled water as single-mode electric chillers (SMEC) and to chillers producing both chilled water and ice as dual-mode electric chillers (DMEC). An electric chiller has a working range defined by a maximum load, i.e. a maximum amount of cooling power it can produce per hour, and a minimum load, i.e. a minimum amount of cooling power it produces per hour when turned on. In practice, the minimum load under which it cannot be operated is equal to 10% to 20% of its maximum load. The efficiency of a chiller is usually characterized by its coefficient of performance (COP), defined as: $COP = \frac{removed\ heat}{input\ electricity}$. The value of this COP is not constant: it varies with the chiller load, i.e. with the amount of produced cooling power, and with the ambient temperature. For example, at an ambient temperature of $30°C$, the COP of a single-mode chiller is about 6 when it is operated at full load, but goes down to 4 (resp. to 3) when the chiller is operated at a 30% (resp. 20%) load rate. Taking into account this varying part-load efficiency is especially important when the cooling demand is satisfied by multiple chillers running in parallel: the load rate of each active chiller should be carefully chosen to maximize the overall conversion efficiency. When designing an LMES in which some cooling capacity is required, we have to select the types of chillers to install by choosing among the available types of chillers provided in a given predefined list. Each type of chiller corresponds to a given category (either SMEC or DMEC), to a given working range (i.e. a minimum and maximum load) and to a given set of performance curves. These performance curves give the amount of consumed electricity as a function of the produced cooling power for some discrete values of the ambient temperature. The design decisions thus correspond to determining the number of chillers of each type to be installed in the system.

CHP unit A Combined Heat and Power (CHP) unit, also called cogeneration system, converts natural gas to electricity and heating power. It contains two main components: a gas engine and a heat exchanger. A gas engine is an internal combustion engine which consumes gaseous fuel (usually natural gas in an LMES) to generate electric power. The exhaust heat is collected by the heat exchanger to obtain heating power. As mentioned above, the locally generated electricity is an intermediate commodity consumed by other devices in the LMES and/or a supply commodity distributed to the customers and sold to the national grid. The heating power produced is either an intermediate commodity stored in a thermal storage tank or a supply commodity used to satisfy the heating demand of the customers. The gas engine has a minimum working load which is usually set to more than half of its maximum working load or capacity [Bis+14]: this means that a CHP unit should be operated within a minimum and maximum working load. Moreover, since the thermal energy is collected from the waste heat of power generation, heating power and electric power are always produced simultaneously in a CHP unit. A CHP unit is thus associated with both a heat efficiency coefficient and an electricity efficiency coefficient measuring the conversion efficiency of gas into electricity and heat. These two coefficients are usually constant, i.e. their value depends neither on the ambient temperature nor on the CHP load. When installing a CHP unit in an LMES, we have to select the type of CHP unit among a discrete list of available types given in a catalog, each type of CHP unit corresponding to a given gas engine and a given heat exchanger. The design decisions thus correspond to determining the number of CHP units of each type to be installed in the system.

CCHP unit A Combined Cooling, Heat and Power (CCHP) generation unit, also called trigeneration system, is a CHP unit linked with an absorption chiller. The absorption chiller uses some of the heat produced by the CHP unit to produce chilled water for air conditioning. A CCHP unit thus converts natural gas into electricity, heating power and cooling power. A CCHP unit cannot simultaneously produce heating and cooling power. It should be either on heating mode (i.e. producing heating power and electricity) or cooling mode (i.e. producing cooling power and electricity). Similar to a CHP unit, a CCHP unit is associated with a heat efficiency coefficient, a cooling efficiency coefficient and an electricity efficiency coefficient. All these three coefficients are constant as they vary neither with the ambient temperature nor with the load rate. Moreover, a CCHP unit must be operated within a minimum and maximum working load. When installing a CCHP unit in an LMES, we need to select the type of CCHP unit in a list of available types. Each type refers to a specific type of gas engine, a given heat exchanger and an a given type of absorption chiller, which are linked as a whole. The design decisions thus consist in determining the number of CCHP units for each type to be installed in the LMES.

Boiler A boiler converts natural gas to heating power. Basically, it is a pressure vessel in which water is heated through the combustion of natural gas. This hot water then circulates through the pipeline network to transfer heating power to the customers' buildings. Boilers may be operated at any value between 0 and their maximum working load or capacity, i.e. the minimum working load is equal to 0. Their conversion efficiency is constant and does not vary with the load or the ambient temperature. Similar to chillers and CHP units, installing a boiler in an LMES requires to select a type of boiler from a list of available models in a given catalog. The design decisions thus correspond to determining the number of boilers of each type to be installed in the system.

Heat pump A heat pump converts electricity to heating or cooling power. The thermal energy is transferred from the customers' buildings to the environment thanks to the circulation of a refrigerant which is driven by electricity. Heat pumps differ with respect to their environmental thermal source: there are e.g. air source heat pumps, geothermal heat pumps and water source heat pumps. In contrast to electric chillers which provide only cooling power, a heat pump can work either in heating mode or in cooling mode by reversing the circulation direction of the refrigerant. The cooling power produced by a heat pump cannot be stored and must be directly consumed by customers, while the heating power can be stored in the form of hot water. A heat pump may be operated at any value between 0 and its maximum working load or capacity. For a given (heating or cooling) mode, its conversion efficiency, which is defined as the ratio between the thermal energy production and the electricity consumption, varies only with the ambient temperature and does not depend on the load rate. When designing an LMES, we are given a discrete list of available heat pump sizes and we should choose the number of heat pumps of each size to install.

Thermal storage Thermal storage in an LMES usually refers to a container of heat or cooling material such as hot water or ice. There are many reasons to integrate thermal storage in an LMES. First, thermal storage may be useful to handle the short-term (i.e. intra-day) variations of the energy demand. Thus, the system can produce

more thermal energy than what is requested by the customers in low-demand periods, store the excess in the thermal storage container for a few hours and use this excess later in peak periods in which the customers' demand is high. Thus, a thermal storage may enable us to avoid installing energy conversion devices with a very large capacity to be used only in a small number of peak periods. Second, thermal storage may also be useful in an LMES involving co-generation devices such as a CHP unit. Namely, recall that with this technology, heat power and electric power are generated simultaneously when the gas engine is turned on. But the customers' thermal demand and electric demand do not necessarily vary in the same way. In this case, the surplus heat energy produced by the CHP unit can be stored in the thermal storage and used later instead of being rejected in the environment [Bis+14]. A thermal storage may be used at any value between 0 and its maximum storage capacity. When designing an LMES, this storage capacity can usually be flexibly chosen to any value within a predefined continuous range. At the operational level, two main constraints should be complied with when operating a thermal storage: the amount of stored energy should not exceed the chosen thermal storage capacity and the evolution of the stored quantity should respect energy inventory balance equations stating that the energy flows into and out of the container together with the energy stored within the container respect the law of energy conservation.

3.2.3 RES diagram and superstructure

The design of an LMES starts from the high-level description of the potential energy system through a tool commonly used in the field of energy systems management called a Reference Energy System (RES) diagram. This diagram describes the commodities, the energy conversion and storage technologies, the energy flows between these technologies and the energy inputs and outputs. The commodities are represented by vertical segments and the technologies by rectangles. Arrows are used to indicate the commodity flows within the system and the connection relationships between commodities and technologies.

The upper part of Figure 3.1 displays the RES diagram of a small illustrative example corresponding to a simple co-generation system. The system consumes natural gas to produce heating power and electricity. The heating power is provided to the buildings connected to the LMES and the generated electricity is sold to the national grid. In this example, the resource commodity is the natural gas, which is noted as GAS in the figure. The supply resources are the heating power provided to the customers and the generated electricity, which are noted respectively as HEAT2 and ELEC. The hot water stored in the thermal storage tank, noted as HEAT1, is an intermediate commodity. The available technologies are CHP units, boilers and thermal storage. The CHP units consume natural gas to simultaneously produce heat and electricity. The produced heat can be either stored as hot water in the storage tank or transferred directly to customers' end. The electricity generated is sold to the national grid. The hot water produced by boilers is used in the same way as the one produced by CHP components.

However, a RES diagram only displays the technologies that may be used in the system but does not provide the detailed catalog describing the types of device available for each technology. It is thus possible to introduce a more detailed description of the design problem through the use of a diagram representing the superstructure of an LMES. As defined by [YI06], the superstructure of an LMES is composed of all the units of equipment considered as candidates for selection. The superstructure thus describes the set of available conversion and storage technologies with their

corresponding input and output commodities together with a list of the available equipment types for each technology (described e.g. by their size or capacity, their energy conversion performance and their price).

The bottom part of Figure 3.1 displays the superstructure of our illustrative example. Let us assume that, for the design of this co-generation system, two types of CHP units with different efficiency coefficients and capacities are available in the catalog and that we can install at most 5 units for each type in the system. There are also two types of boilers with distinct efficiencies and capacities and we are allowed to install no more than 10 units of each type of boiler. Finally, there is a single thermal storage tank and its storage capacity should not exceed 50 MWh. In the superstructure provided in Figure 3.1, the input parameters associated to the maximum number of each type of devices or to the maximum storage capacity allowed are displayed in grey.

Designing an LMES will thus consist in determining the number of units of each type of devices (among the ones belonging to the superstructure) to be installed in the actual system and in fixing the capacity of the energy storage devices to be installed.

In our simple example, the design decisions would consist in fixing the number of CHP units and boilers of each available type to install (if any) and in determining the size of the thermal storage tank to be built (if any). These decisions are denoted by blue question marks in the superstructure provided in Figure 3.1.

3.3 Design considerations

Besides the description of the superstructure, some other aspects need to be taken into account when designing an LMES.

3.3.1 Price and availability of the resource commodities

As mentioned in Subsection 3.2.1, the resource commodities are often bought from an external supplier. The corresponding unit buying price is likely to vary over the lifetime of the LMES. In fact, depending on the case, this price may vary over the day, the week and the season and may also display long-term trends over the years. When designing an LMES, it is important to take into account these variations and thus to follow as precisely as possible the time at which the resource commodities are used. This will namely directly impact our ability at accurately estimating the operation cost of the LMES. Moreover, the daily variations of the price will have a strong impact on the sizing of the thermal storage. In case there are strong variations in the buying price within the day, it may be interesting to install a rather large thermal storage capacity. This will enable us to produce energy in periods where the price of the resource commodities is low, store it for a few hours and use it afterwards to meet the energy demand in periods where the price of the resource commodities is higher. On the contrary, a thermal storage might not be as useful in case the price is relatively stable throughout the day.

Furthermore, we should also take into account the limited availability of these resource commodities. For instance, the electricity, when considered as a resource commodity, is bought from an external supplier through the national electric grid. There is usually a limit on the electric instantaneous power that may be consumed by the system. The value of this limit is negotiated each year with the supplier and gives rise to a contract. The corresponding cost is proportional to the value of the

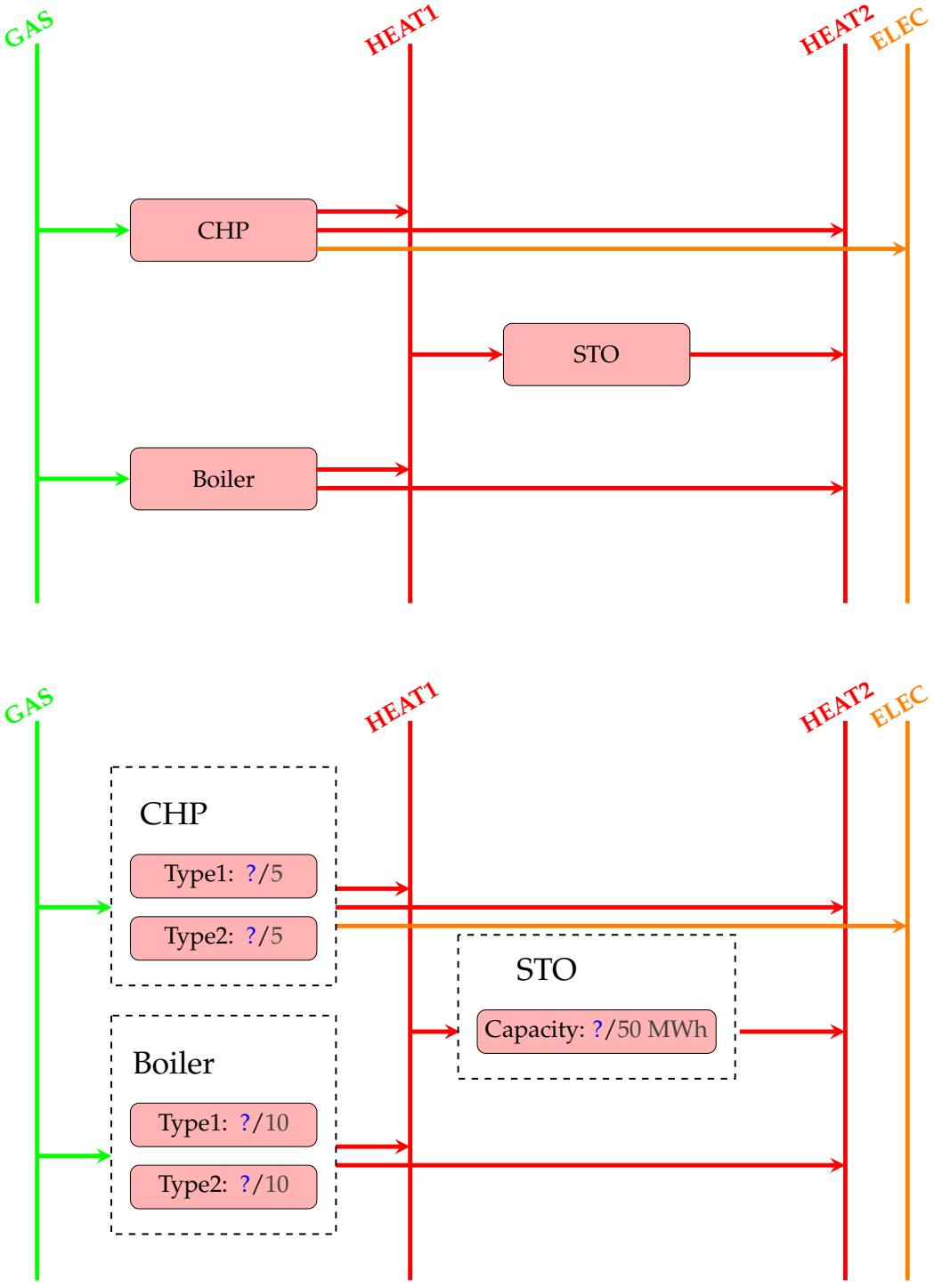


FIGURE 3.1: RES diagram (above) and superstructure (bottom) of a simple cogeneration system

contracted maximum electric power. As for renewable energy sources such as solar radiation, as mentioned above, their availability depends on weather conditions and is also strongly limited.

3.3.2 Demand for the supply commodities

The energy demand to be satisfied by an LMES depends on many factors, among which is the ambient temperature, the number of consumers located in the district and their daily and weekly schedules. As a consequence, it displays a strong temporal variability.

First, the energy demand strongly varies throughout the day. For instance, the demand for cooling power is usually much higher at daytime than at night. Second, the demand pattern of a weekday often differs from the one observed during the week-end due to the fact that consumers do not have the same routines in working days as in the week-end. Third, the cooling and heating demands are highly dependent on the ambient temperature and thus feature strong seasonal variations: the total cooling (resp. heating) demand of a winter day is thus significantly smaller (resp. higher) than the one of a summer day. These demand variations display a seasonal and periodic pattern and are thus to some extent predictable. Nevertheless, extreme weather conditions which are more difficult to accurately anticipate may happen. These extreme conditions lead to the presence of extreme days, i.e days in which the energy demand is exceptionally high or low as compared to its average value. A major requirement is that the LMES should be able to meet this demand at all time, whatever the hour of the day, the day in the week and the time in the year. In particular, it should be able to satisfy all the demand, even during the extreme days.

Furthermore, an LMES is a long-term investment with an expected lifetime spanning several decades and is often built together with the district it will serve. This means that the forecasted yearly energy demands feature an upward trend over the years as the district is developed and new buildings progressively connect to the LMES. As a consequence, the design of an LMES should not be seen as a one-time decision but rather as a process in which investment decisions are made step by step, following the development of the district and the increase of the demand. This implies that we not only have to determine the final optimal layout of the system but also a multi-phase strategic deployment plan describing how the various conversion and storage devices should be progressively added to the system over the years.

3.3.3 Ambient temperature

The ambient temperature has an impact on the efficiency of some technologies. For example, the COP of electric chillers varies with the ambient temperature and this impact should be considered when estimating the operation cost of the system.

3.3.4 Maintenance cost

Each device installed in an LMES needs to regularly undergo maintenance to stay in a good working condition. The total maintenance cost of a device over its lifetime can be straightforwardly computed once we know when it will be installed in the system. Depending on the project, the maintenance cost of a device is either fixed or varies with the number of years the device functions in the LMES. In both cases, this maintenance cost can be incorporated in the fixed installation cost, resulting either in a fixed or a time-varying total installation and maintenance cost.

3.3.5 Discount rate

Since the construction of an LMES is a long-term investment project lasting several decades, we need to take into account the time value of money. Basically, investing money in the project now is more costly than investing money in the future. This is because an amount of capital, if not invested in the project now, will grow each year thanks to e.g. bank interests.

Let β denote the discount rate, i.e. the percentage by which an amount of money will grow each year if not invested in the project. Investing an amount R of money n years from now is considered equivalent to investing the amount of money $\frac{R}{(1+\beta)^n}$ now. Consequently, installing a device in the LMES in the future costs less than installing it now.

Our objective in this work will thus be to minimize the net present cost of the LMES, i.e. the net present value (NPV) of the total cost of designing and operating the system over its whole lifetime.

3.4 Problem statement

In Sections 3.2 and 3.3, we described the various elements to be considered when optimally designing an LMES. This optimization problem can be formally stated as follows. Given the available energy resources, the predicted total demand for supply commodities in the neighbourhood over the LMES lifetime and the predefined superstructure, select the technologies to be installed in the system and determine the number and size (i.e. capacity) of the corresponding conversion and storage devices so as to minimize the total construction and operation cost of the system while satisfying the customers' demand at all time.

We now describe in more details the problem input data, the decisions to be made, the main constraints to be complied with and the objective to be reached.

3.4.1 Input data

The problem input data first consist in a description of the superstructure of the system. As explained in Subsection 3.2.3, the superstructure gives a list of all the conversion and storage devices considered as candidates for selection. Each candidate device corresponds to a given technology (with its input and output commodities), to a size (or maximum capacity), to an energy conversion performance (if relevant) and to an installation and maintenance price. We also know the unit cost per kilowatt (KW) of the maximum electric power to be contracted with the electricity supplier.

Second, we are given a set of time series describing the forecasted time variations (usually on a hourly basis) of the buying price and availability of the resource commodities, of the demand for supply commodities and of the ambient temperature over the LMES lifetime.

Finally, the value of the rate to be used to discount the value of future costs is also assumed to be known.

3.4.2 Decisions

Design decisions The design decisions to be made consist in selecting, among the list of technologies and devices provided in the superstructure, the ones to be installed in the system and in determining the value of the maximum electric power to be contracted with the electricity supplier. Note that, regarding the sizing of the

installed devices, two different situations occur. For some technologies such as thermal storage, the capacity of the device can be fixed to any continuous value in a predefined range and we only have to determine this value. In contrast, for other technologies such as chillers, CHP units or heat pumps, the capacity of the devices is to be chosen within a discrete list of values corresponding to off-the-shelf models available at the equipment manufacturers. In this case, we should determine the discrete number of devices of each available type to be installed in the system.

Moreover, as mentioned above, as the energy demand features long-term trends over the years, these decisions should not be made once and for all at the beginning of the system lifetime, but rather progressively, year after year, to adjust to the long-term variations of the energy demand. In other words, we should be able to provide, not only a final layout of the system, but also a multi-phase deployment plan covering the system lifetime and describing how the devices should be installed year after year.

Operation decisions Our design decisions will have a strong impact on the daily operational management of the system, in particular on our ability at satisfying the demand for supply commodities. Furthermore, the operation cost, which corresponds mainly to the cost of buying the resource commodities, makes up for a large part of the total cost of the system (typical value is between 20% and 60% of the total cost) and can therefore not be neglected when computing the total cost of an LMES.

The main challenge here is that it is not possible to use average values of the demand for supply commodities and/or average values of the buying price of resource commodities to estimate the impact of our design decisions on the operational management of the system. Namely, both the demands and prices display strong time variations. Moreover, the technical features of the conversion devices, in particular the fact that they have a non-zero minimum working load and a maximum working load, make it impossible to ensure that an LMES able to satisfy the average hourly demand will be able to actually satisfy the time-varying demand at all time. Furthermore, some devices such as chillers have a load-dependent and temperature-dependent COP, which means that the consumption of resource commodities is not proportional to the production of supply commodities. Finally, as the buying price of resource commodities is time varying, it is not enough to know how much resource commodities we need: we also have to determine when these resource commodities will be bought to accurately compute the corresponding cost. Hence, using an average or aggregated representation of the demand to estimate the operation cost may lead to a significant underestimation of its actual value.

As a consequence, in order to accurately estimate the impact of our design decisions on the operational management of the LMES, we should build a detailed operational schedule describing, on a hourly basis, the on/off status and the load allocation of each conversion device, the amount of energy stored in each storage device and the amount of supply commodities bought from an external supplier. Ideally, we should build such a schedule for each hour, each day and each year of the LMES lifetime.

The design and operation scheduling decisions to be made for the illustrative simple cogeneration system are partially represented in Figure 3.2. The upper part of the figure shows the design decisions which have been made for a given year of the system lifetime: note how the blue question marks of the superstructure of Figure 3.1 are replaced by blue numbers. Thus, at the beginning of the considered year, the LMES comprises 2 CHP unites of type 2, 8 boilers of type 1, 2 boilers of type

2 and a thermal storage capacity of 30 MWh. For each hour of the corresponding year, we should build a schedule describing among others the on/off status of each devices. The bottom part of Figure 3.2 partially describes (see the red numbers) this schedule by displaying the number of units of each type active during this hour together with the amount of thermal energy stored in the tank at the end of this hour.

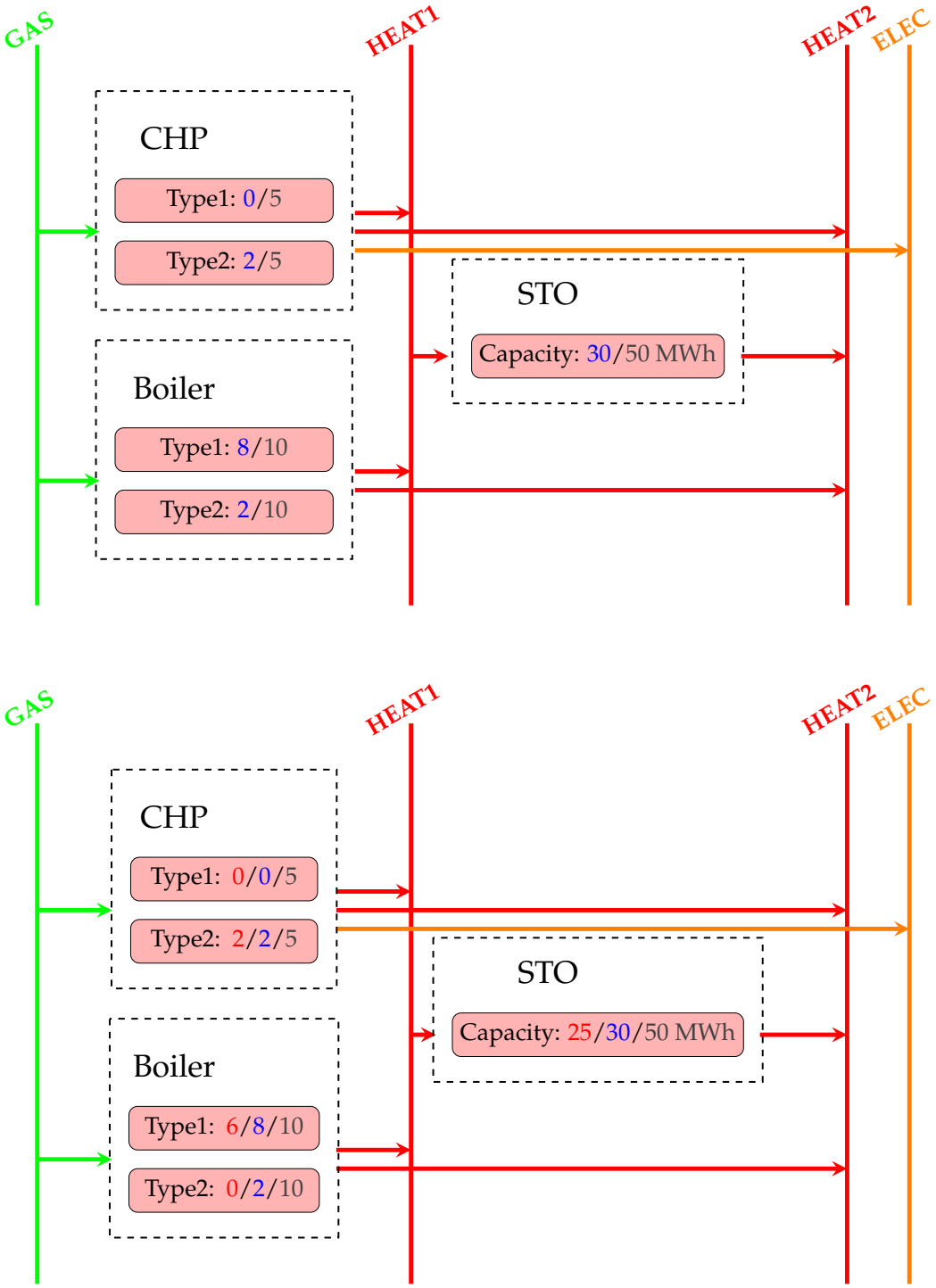


FIGURE 3.2: System layout for a given year and operation schedule for a given hour for a simple cogeneration system

3.4.3 Constraints

The designed LMES should comply with some restrictions. For instance, we have an upper bound on the number of devices of a given type that may be installed. For technologies such as thermal storage for which the capacity can be customized, we also have an upper bound on the capacity of the tank to be installed.

We also have a large set of constraints to ensure that the operational schedule built for each hour is relevant in practice. These constraints guarantee among others that the demand for supply commodities is satisfied, that the energy balance in thermal storage devices is respected, that energy conversion is carried out in each device following its actual COP... and this at all time, i.e. at each hour of the LMES lifetime.

3.4.4 Objective

As mentioned above, our objective is to minimize the net present total cost of the LMES, i.e. the net present value (NPV) of the total cost of designing and operating the system over its whole lifetime.

3.5 Case studies

We now introduce three case studies which we will use in our numerical experiments to evaluate our models and algorithms. These case studies correspond to LMESs that EDF is currently developing in China.

3.5.1 District cooling systems

A district cooling system (DCS) is a local energy system which distributes cooling capacity in the form of chilled water to a set of buildings located nearby. The customers purchase the delivered chilled water from the DCS operator and use it for air conditioning and dehumidifying or for industrial process cooling.

In a DCS, we have a single supply commodity, the chilled water, whose demand displays strong daily, weekly and seasonal variations. In our numerical experiments, we consider two DCSs located in two different cities. City A is in a tropical region in the south of China so that customers need cooling power during most of the year. City B is in a region with a temperate climate: hence, the cooling demand is mainly concentrated around the summer. In both cases, the expected lifetime of the DCS is 30 years. The population of the district, and as a consequence the demand for chilled water, is anticipated to increase during the first years and to stay stable afterwards.

In the considered DCSs, there is a single resource commodity corresponding the electricity bought from the national grid. The price for electricity is daily periodic but is otherwise stationary and does not change from one day to the next. Its availability is limited by a contract with the electricity supplier that may be renegotiated each year.

In terms of technologies, the superstructure comprises a set of single-mode electric chillers, dual-mode electric chillers and a thermal storage. The capacity of the thermal storage tank may take any continuous value within a predefined range. As for the chillers, we have a discrete set of available models that may be bought from equipment manufacturers and we should determine the number of chillers of each type to be installed in the DCS.

Figure 3.3 displays the RES diagram of the DCSs considered in our two case studies. The resource commodity is denoted by ELEC, the intermediate commodity by ICE and the supply commodity by COLD. Single-mode chillers consume the ELEC commodity to produce COLD. As for the dual-mode chillers consume, they also consume the ELEC commodity but produce both ICE to be stored in the storage tank and COLD to be directly delivered to the customers' buildings.

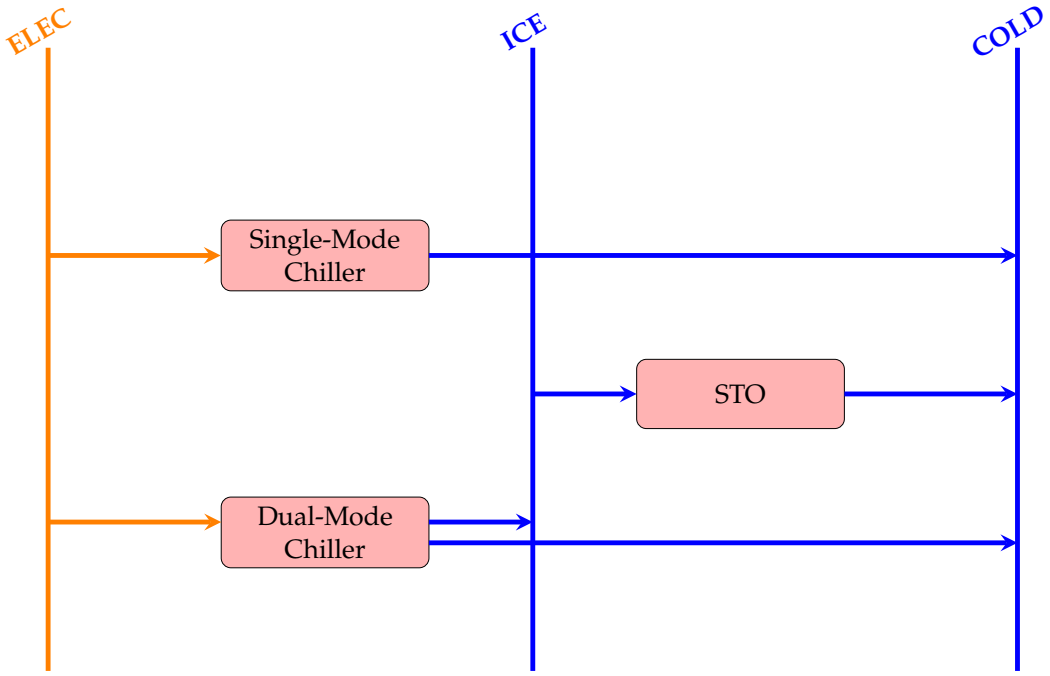


FIGURE 3.3: RES of the studied district cooling systems

3.5.2 Trigeneration system

The third case study corresponds to a trigeneration system located in a city where both heating and cooling power are demanded. The system serves various types of customers in the surrounding area, such as residential buildings, commercial centers, office buildings and hotels. Its expected lifetime is 30 years. The served district is currently under development: the population is thus expected to increase for the first four years and to stay stable afterwards.

The system should satisfy both a cooling demand and a heating demand. These demands have a seasonal feature. Cooling power is needed during 6 months whereas heating power is required during 4 months. The cooling supply period and heating supply period have no overlap. We assume that the energy demand grows with the same trend as the population in the district. The ambient temperature of the system is considered to be yearly periodic.

The available resource commodities are natural gas and electricity. The unit buying price natural gas is constant in time. The price for electricity is daily periodic but is otherwise stationary and does not change from one day to the next.

The technologies available in the superstructure include CCHP units, boilers, air source heat pumps, single-mode electric chillers, dual-mode electric chillers, hot water thermal storage and ice thermal storage. Recall that for thermal storage, the capacity can be set to any value in a predefined continuous range. For the other

technologies, the installed devices should be selected among the ones available in the manufacturer's catalog so that there is only a finite and discrete set of potential capacities.

Figure 3.4 displays the RES diagram of the trigeneration system. On this figure, ELEC1 and GAS denote the resource commodities, ELEC2, HEAT1 and ICE the intermediate commodities and HEAT2 and COLD the supply commodities. The CCHP unit converts the resource commodity GAS into electricity ELEC2 (to be consumed within the LMES), heating power HEAT1 (to be stored in the thermal storage STO_HEAT), HEAT2 and COLD (to be distributed directly to the customers' buildings). As for the boiler, it also converts GAS into HEAT1 or HEAT2 heating power. We then have three technologies powered by electricity (either bought from the supplier ELEC1 or generated locally by the CCHP unit ELEC2). The heat pump produces HEAT1 or HEAT2 heating power and COLD, i.e. cooling power to be directly distributed to the customers in the form of chilled water. Single-mode chillers only produce COLD while dual-mode chillers produce both COLD and ICE, i.e. cooling power to be stored in the thermal storage STO_COLD before being distributed.

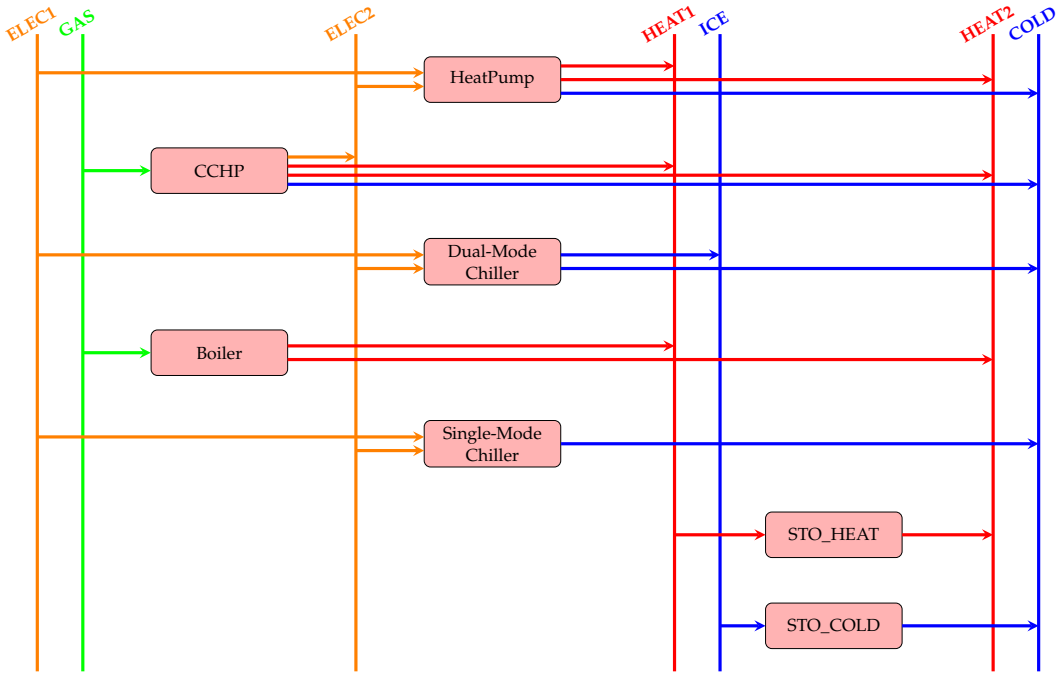


FIGURE 3.4: RES of the trigeneration system

3.6 Conclusion

This chapter provided a detailed description of the optimization problem under study in this PhD work. We first introduced the main components of an LMES by focusing on the ones which have a major impact on the total cost of the system. These components are roughly classified into two classes: commodities and technologies. This classification allowed us to represent the layout of an LMES using either an RES diagram or a more detailed superstructure diagram. Second, we presented several additional key aspects of the problem which also have a strong influence on the LMES final layout. The overall optimal design problem was then formally stated as

a classic combinatorial optimization problem. We underlined the necessary input data, the decision variables, the main constraints to be respected and the objective function. Finally, we briefly introduced the three case studies that we will use in our numerical experiments. They correspond to two types of energy systems: district cooling systems and trigeneration systems.

Chapter 4

Problem modeling and mathematical formulation

4.1 Introduction

The combinatorial optimization problem described in Chapter 3 leads to the formulation of a huge-size mixed-integer non-linear program which is computationally intractable. This huge size mainly comes from the need to simultaneously make strategic design decisions over a long-term planning horizon and evaluate the impact of these strategic decisions on the operation cost of the system through the building of detailed schedules using short (typically hourly) time-steps. Moreover, the presence of energy conversion devices whose efficiency depends on the load (e.g. electric chillers) brings some non-linearities in the mathematical programming formulation.

In this chapter, we first present in Section 4.2 the modeling choices and assumptions we use to reduce to some extent the size of the obtained mathematical program and to eliminate the aforementioned non-linearities. We then provide in Section 4.3 the resulting MILP formulation. Finally, Section 4.4 presents the computational results obtained when directly solving this MILP formulation with a mathematical programming solver on instances based on the three case studies described in Section 3.5.

4.2 Problem modeling

4.2.1 Investment phases

The lifetime of an LMES usually spans several decades: let us denote by Y the corresponding number of years. As explained earlier in Section 3.4, the energy demand to be satisfied by the LMES is likely to display a long-term upward trend, especially when the LMES is built in a new district currently under development. This means that it may not be cost-efficient to build the whole system in one step. We should rather only install part of the energy conversion and storage devices at the beginning of the system lifetime and progressively add, year after year, the other devices as they become necessary to adjust to the long-term variations of the demand.

In what follows, we propose to divide the system lifetime into a set of Φ investment periods termed phases. A phase typically spans one or several years. Let $\underline{y}_\phi$ (resp. $\overline{y}_\phi$) be the index of the first (resp. the last) year belonging to phase ϕ.

We make two modeling assumptions here. First, design decisions, such as the installation of new devices or the building of additional storage capacity, can only be made at the beginning of a phase. Second, the hourly evolution over the year

of the energy demands, of the price and availability of resource commodities are described by the same time series for all the years belonging to the same phase. Note that these two modeling assumptions are based on the ones currently used by EDF practitioners when studying the design of an LMES.

4.2.2 Representative days

Ideally, the operation cost during a given investment phase should be estimated by building a detailed operation schedule describing the on/off status and the load allocation of each device together with the stored energy for each hour, each day and each year of this phase. This clearly leads to a huge intractable optimization problem.

A first simplification can be made here by exploiting the second assumption on investment phases mentioned in Subsection 4.2.1. Basically, this one amounts to assuming that all years belonging to the same phase are strictly identical in terms of energy demand, commodity prices and availability. Thus, a single operation schedule covering 365 days may be built for all these years and the total operation cost over the investment phase can be computed as the sum, over all years belonging to this phase, of the actualized value of the cost of this schedule.

However, building an operation schedule covering the 365 days of a full year for each investment phase is still out of reach in terms of computational effort. This is why, in order to further decrease the size of the problem, we propose to build an operation schedule only for a subset of disjoint typical and extreme days. These days should be carefully selected in order to represent, as best as possible, the various conditions under which the LMES will be operated and to obtain an estimation, as accurate as possible, of the operation cost.

We propose to select a subset of typical days based on an analysis of the time series representing the hourly evolution of the input parameters over a full year. More precisely, we seek to identify the 'best' typical days by solving a clustering problem which corresponds to an extension of the k-medoid problem investigated by [Zat+19]. This problem consists in partitioning the set of days belonging to the available time series into a predefined number of groups or clusters and in selecting one member (i.e. one day) in each cluster to represent all the days belonging to the cluster. The objective is to minimize the sum of the Euclidean distance between each day in the time series and its representative.

Moreover, we propose to choose 4 extreme days for each supply commodity and for each resource commodity with a limited availability. For each supply commodity, we first select 2 days, the day with the highest total daily demand and the one with the highest hourly demand, in order to ensure that the capacity of the installed devices will be large enough to satisfy the demand even if it is extremely high. Moreover, in case the demand is very low but not equal to zero, we may encounter difficulties to satisfy it due to the constraints on the minimum production power of some devices. In order to avoid it, we also select as extreme days the day with the lowest non-zero total demand and the one with the lowest non-zero hourly demand. As for the resource commodities with a limited availability (e.g. solar radiation), we also select 4 extreme days: the day with the highest hourly availability, the one with the highest total daily availability, the one with the smallest hourly availability and the one with the smallest total daily availability. Note that, the extreme days are assumed to happen once in the year so that the corresponding cost has a small impact on the total annual operation cost. However, by imposing that the system is able to satisfy the energy demands even under extreme conditions these extreme days have

a huge impact on the design of the system. Moreover, for the time being, we do not select extreme days with respect to the price of resource commodities because in our case studies, these prices are either constant or follow the exact same pattern every day.

Figure 4.1 shows the hourly evolution of the cooling demand for 6 representative days which may be selected with respect to the supply commodity COLD to estimate the operation cost during a given investment phase for the district cooling system located in City B (see Subsection 3.5.1). Note that the hourly demand is normalized, i.e., is expressed as a proportion of the maximum hourly cooling demand. We choose 2 typical days and 4 extreme days to represent the cooling demand variation of the year. Each representative day is allocated a weight, which corresponds to the number of days of the year it stands for in the clustering. Thus, Day 136 (resp. Day 263) is a typical day which acts as the representative of a cluster comprising 66 (resp. 95) days and will be assigned a weight equal to 66 (resp. 95). Each extreme day is assumed to occur only once a year and is thus assigned a weight equal to 1. Note that the sum of these weights is not equal to 365 because in the considered DCS located in a region with a temperate climate, there is no cooling demand in more than half of the year.

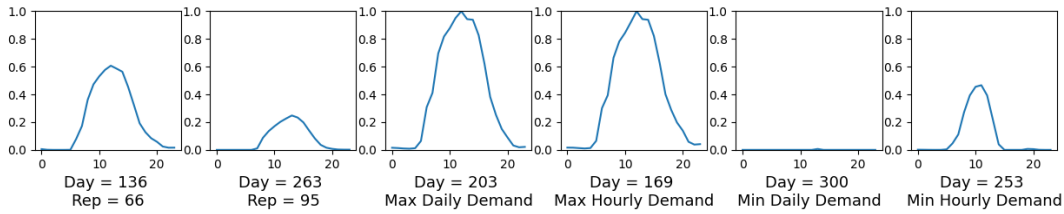


FIGURE 4.1: Representative days (2 typical days and 4 extreme days) for the cooling demand during Investment Phase 1 for the DCS located in City B

4.2.3 Piece-wise linear approximation of non-linear performance curves

As explained in Chapter 3, for some technologies, the conversion efficiency coefficient is not constant but rather depends on the ambient temperature and/or on the load. In this case, the manufacturer of each conversion device provides us with a set of non-linear performance curves giving the energy consumption as a function of its load rate under a discrete set of values of the ambient temperature. A typical example are the electric chillers. The conversion efficiency of a single-mode chiller is described by a set of performance curves: each curve depicts the electric power consumed by the chiller as a function of the produced cooling power under a certain ambient temperature. Similarly, the conversion efficiency of a dual-mode chiller is described by two sets of performance curves: one set for its cooling mode and one set for its ice-production mode.

The non-linearity of these performance curves poses a major difficulty for the efficient resolution of the optimization problem as this one would be formulated as a mixed-integer non-linear program.

To overcome this difficulty, we propose, for each type of conversion device featuring a part-load efficiency, to approximate each of its non-linear performance curves through a piece-wise linear function. This function is built using the following procedure. We first choose a number of breakpoints to be used in the piece-wise linear approximation. We then determine the coordinates of these breakpoints so

as to minimize the distance between the approximate performance curve and the actual one. This is done by heuristically solving a small non-linear unconstrained optimization problem through the Powell's conjugate direction method [Pow64]. In Figure 4.2, we show as an example the piece-wise linear approximation of a the performance curve of a single-mode electric chiller (SMEC). Four breakpoints are used to approximate the non-linear curve so that the piece-wise linear approximation comprises three continuous segments.

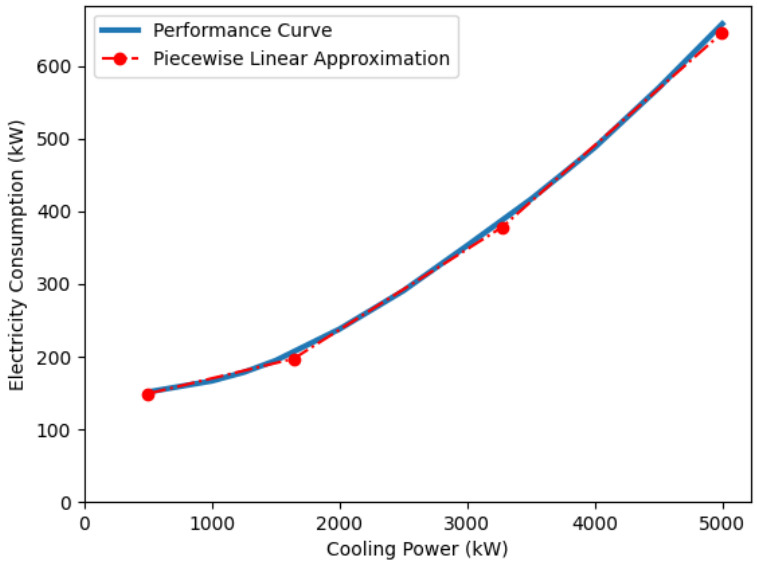


FIGURE 4.2: Piece-wise linear approximation of the performance curve at 30°C of a single-mode electric chiller

We then consider historical data about the average ambient temperature at each time-step of each representative day and use this information to identify which approximate performance curve (among the ones built in the previous step) should be used to compute the energy consumption of each type of device during this time-step.

4.2.4 Leveraging the convexity of the performance curves

When applicable, we propose to exploit the convexity of the performance curves of the conversion devices to further reduce the size of the optimization problem. When the performance curves of a given type of energy conversion devices are convex, we have the following property.

Lemma 4.2.1. *For a set of identical active conversion devices producing the same commodity, the optimal load allocation consists in equally distributing the total output power between the active devices.*

Proof. The proof is done by contradiction. Suppose we have two identical active devices, producing a total output power of P with device 1 producing P^1 and device 2 producing $P^2 > P^1$. Let $\pi : P \mapsto Q = \pi(P)$ be the convex performance curve of these two devices. The total amount of input commodity consumed by the two devices producing P is $Q = \pi(P^1) + \pi(P^2)$.

We show that this load allocation is not optimal, i.e. that it is possible to reduce the total amount of consumed commodity. Namely, let δP be a small variation in the output. By decreasing the output of device 2 by δP, we can obtain a decrease in the input commodity consumed by this device of $\pi'(P^2)\delta P$, where π' is the derivative function of π. In order to still be able to provide a total output of P, we increase the output of device 1 by δP, which leads to an increase in its input commodity consumption of $\pi'(P^1)\delta P$.

By convexity of function π, $\pi'(P^1) < \pi'(P^2)$. Hence the total amount of input commodity consumed with the load allocation $(P^1 + \delta P, P^2 - \delta P)$ is smaller than the one consumed with the load allocation (P^1, P^2). The result follows.

□

Let us now focus on the case in which the performance curve π is convex and piece-wise linear. When several identical active devices are simultaneously producing the same output commodity, the relation providing the total amount of consumed input commodity as a function of the total amount of output power can be plotted as an aggregate performance curve. We have the following property:

Lemma 4.2.2. *Let π be the piece-wise linear and convex performance curve of a given type of active device producing a given commodity. π involves B breakpoints. Let (a_b, o_b) be the abscissa and ordinate of breakpoint b.*

The aggregate performance curve Π^S of S identical active devices of this type simultaneously producing the same commodity is also piece-wise linear and convex. It involves B breakpoints whose coordinates are given by (Sa_b, So_b).

Proof. Let us consider the case where the total output of the S active devices is $P \in [SP^{min}; SP^{max}]$ where $[P^{min}, P^{max}]$ is the production range of a single device.

By Lemma 4.2.1, the optimal load allocation consists in requiring that each active device $\sigma = 1...S$ produces the same output $P^\sigma = \frac{P}{S}$. Let b be the index of the breakpoint of function π such that $P^\sigma \in [a_b, a_{b+1}]$. The energy consumed by each device σ is thus given by: $Q^\sigma = s_b P^\sigma + c_b$ where s_b and c_b are the slope and constant value of the b^{th} line segment of π. The total energy consumed by the S devices is thus equal to $Q = s_b P + c_b S$. This equality holds for any value of P such that $\frac{P}{S} \in [a_b, a_{b+1}]$, i.e. any value of $P \in [Sa_b, Sa_{b+1}]$. This means that Π^S is linear over the segment $[Sa_b, Sa_{b+1}]$, with a slope equal to s_b and a constant value of $c_b S$.

By generalizing this result to all possible values of the total output P, we have that Π^S is a piece-wise linear function involving B breakpoints of coordinates (Sa_b, So_b). Moreover the slope of Π^S on its b^{th} segment is s_b. As π is convex, we have $s_b \leq s_{b+1}, b = 1...B$. Π^S is thus convex. □

We first note that, when the performance curves are convex, the optimal load allocation between a set of identical active devices producing the same commodity consists in equally distributing the total output power between the active devices: see Lemma 4.2.1. For instance, a set of identical single-mode electric chillers simultaneously producing cooling power should be operated at the same working load to minimize the electricity consumption.

Moreover, let us now focus on the case where the performance curve of these devices is piece-wise linear and convex. In this case, when several identical active devices are simultaneously producing the same output commodity, the relation providing the total amount of consumed input commodity as a function of the total amount of output power can be plotted as an aggregate performance curve taking the form of a piece-wise linear and convex function: see Lemma 4.2.2. More precisely, assume that the performance curve of an individual active device producing

a given commodity involves B breakpoints with (a_b, o_b) the abscissa and ordinate of breakpoint b. The aggregate performance curve of S identical active devices of this type simultaneously producing the same commodity is also piece-wise linear and convex and involves B breakpoints whose coordinates are given by (Sa_b, So_b). Figure 4.3 illustrates this for a given type of single-mode electric chillers. The blue curve describes the performance of a single active chiller of this type as a piece-wise linear function. The orange (resp. green) curve describes the aggregate performance of two (resp. three) chillers of this type simultaneously producing cooling power at the same load rate. Thus, each point of these aggregate performance curves gives (in ordinate) the minimum electricity power needed to produce the demanded cooling power (in abscissa) for the corresponding number of identical active chillers.

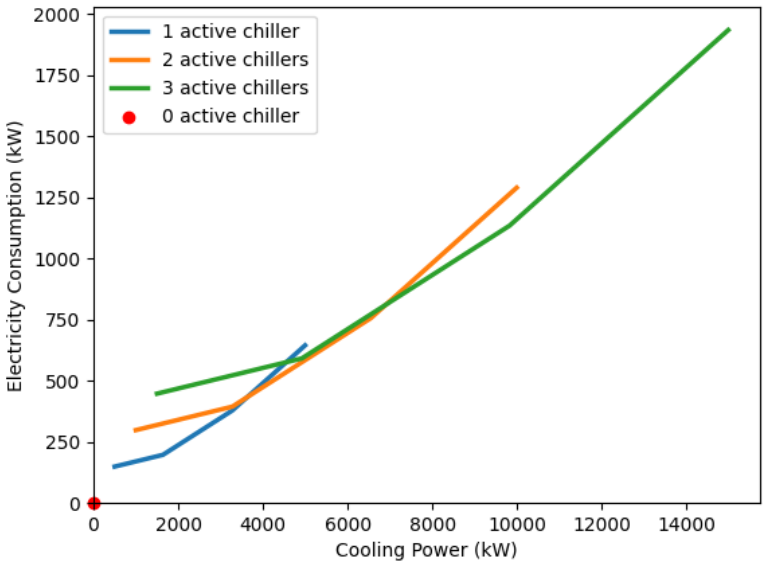


FIGURE 4.3: Aggregate performance curves of single-mode electric chillers

Note that in case the performance curves are linear, i.e. in case the energy conversion coefficient is constant, any load allocation of the total output power of a set of identical active devices leads to the same total consumption of the input commodity. It is thus also possible to build aggregate linear performance curves in this case.

The use of these (convex or linear) aggregate performance curves enables us to simplify to some extent the building of the operation schedules for each representative day. Namely, it is possible to build these schedules by using aggregate scheduling variables describing the number of on/off devices of each type, their aggregate production of output commodity and their aggregate consumption of input commodity (rather than detailed scheduling variables describing the on/off status, output and input power of each individual conversion device). This leads to a significant decrease in the number of variables and constraints introduced in the formulation. It also avoids symmetry problems coming from the fact that there exist many alternative optimal solutions that differ from one another only by the way each detailed operation schedule is assigned to one of the identical conversion devices.

Moreover, thanks to the fact that we will seek to minimize the operation cost, we do not have to introduce extra binary variables to handle these piece-wise linear and convex functions. We only have to add a set of linear constraints to ensure that the

point representing the aggregate amounts of output and input commodity of a set of S identical active devices will be in the epigraph of each corresponding aggregate curve. In any optimal solution, this point will be on the aggregate performance curve. This leads to the formulation of the union of a sequence of disjoint convex sets depending on three aspects: the aggregate cooling power, the aggregate electricity consumption power and the number of active identical chillers. Such formulation problem is usually called a disjunctive programming problem and we recommend readers to refer to [Vie15] for more details.

4.3 Mixed-integer linear programming formulation

Thanks to the modeling choices and assumptions introduced in the previous section, the problem of optimally designing an LMES can be formulated as an MILP. We provide in this section a detailed description of the corresponding formulation.

4.3.1 Indices and input parameters

Time representation and management

We use a three-level time discretization in our problem modeling. We first divide the system lifetime involving Y years into a set of Φ investment periods termed phases. Let $\underline{y}_\phi$ (resp. $\overline{y}_\phi$) be the index of the first (resp. the last) year belonging to phase ϕ. The index d represents the time unit used to build schedules at the operational level. In practice, the length of this time unit is chosen from a half day to one week. In this thesis, we use a time unit of one day to build operation schedules. For each considered phase $\phi \in \{1, ..., \Phi\}$, a set $\mathcal{D}_\phi$ of typical and extreme days selected as explained in Subsection 4.2.2 are used to represent the various daily patterns of the energy demand, commodity price and availability. Each representative day d of phase ϕ, denoted by (ϕ, d), is assigned a weight $w_{\phi,d}$. If d is a typical day identified through the clustering approach discussed in Subsection 4.2.2, $w_{\phi,d}$ corresponds to the number of days it represents, i.e. to the number of days in the original data assigned to the same cluster as day (ϕ, d). If d is an extreme day, $w_{\phi,d}$ is set to 1. Finally, in order to depict the intra-day variation of the input parameters, each representative day is further divided into H time-steps indexed from 0 to $H-1$, each of which has a duration of Δt hours. For the sake of readability, in the remainder of this thesis, we use the bold letter $\mathbf{t}$ to represent the time period (ϕ, d, h), which corresponds to phase ϕ, selected day d and time step h.

The net present value of the future cost is calculated using an annual discount rate denoted by γ. We use two distinct actualization rates in our problem formulation. The first one, called the design actualization rate, is used to compute the net present value of the system design cost at the beginning of each phase. It is equal to $\alpha_\phi = \frac{1}{(1+\gamma)^{\underline{y}_\phi}}$ for phase ϕ. The second one, called the operation actualization rate, is used to calculate the net present value of the operation cost during phase ϕ. It is defined by $\beta_\phi = \sum_{y=\underline{y}_\phi}^{\overline{y}_\phi} \frac{1}{(1+\gamma)^y}$ for phase ϕ. Recall that all years belonging to a given phase ϕ are assumed to be described through the same input data so that the total annual operation cost is constant for all years belonging to ϕ. The net present value of the operation cost during ϕ can thus be obtained by multiplying the total annual operation cost during phase ϕ by β_ϕ.

Commodities

Let $\mathcal{N} = \{GAS, ELEC, HEAT, COLD, ICE, ...\}$ denote the various forms of energy in the considered LMES. As can be seen from the examples and case studies described in Chapter 3, in an LMES, a given form of energy may correspond to different commodities (e.g. electricity bought from the national grid or generated by a gas engine). In order to distinguish between them, we denote each commodity by a pair (n, s): index $n \in \mathcal{N}$ represents the form of energy and index $s \in \{1, \cdots, S_n\}$ is used to number commodities with the same type n. The set of all commodities is thus $\mathcal{C} = \{(n, s) : n \in \mathcal{N}, s \in \{1, \cdots, S_n\}\}$. $\mathcal{C}$ may be partitioned into the set of resource commodities $\mathcal{C}^R$, the set of intermediate commodities $\mathcal{C}^I$ and the set of supply commodity $\mathcal{C}^S$.

For example, the set of commodities involved in the trigeneration system described by Figure 3.4 is given by:

$$\mathcal{C} = \{(ELEC, 1), (ELEC, 2), (GAS, 1), (HEAT, 1), (HEAT, 2), (ICE, 1), (COLD, 1)\}$$

with $\mathcal{C}^R = \{(ELEC, 1), (GAS, 1)\}$, $\mathcal{C}^I = \{(ELEC, 2), (HEAT, 1), (ICE, 1)\}$ and $\mathcal{C}^S = \{(HEAT, 2), (COLD, 1)\}$.

We use $EP_{\mathbf{t},\mathbf{c}}$ to denote the energy price of a commodity $\mathbf{c}$ at timestep $\mathbf{t}$. This parameter corresponds to the price for purchasing one unit of energy if $\mathbf{c} \in \mathcal{C}^R$ is a resource commodity. Moreover, let $\mathcal{C}^{S,Sell} \subset \mathcal{C}^S$ be the subset of supply commodities which may be sold to an outsider buyer in addition (or instead of) being sold to the buildings connected to the LMES (e.g. electricity generated by the LMES sold to the national grid). For $\mathbf{c} \in \mathcal{C}^{S,Sell}$, $EP_{\mathbf{t},\mathbf{c}}$ denotes the price for selling one unit of energy to the outsider buyer.

The demand of a supply commodity $\mathbf{c}$ at time-step $\mathbf{t}$, i.e. the amount of this commodity to be produced by the LMES, is denoted by $Dem_{\mathbf{t},\mathbf{c}}$. In a typical cooling system, the difference between the temperature of the circulating chilled water and the ambient temperature is less than $20°C$. Therefore, we may assume that the thermal loss during distribution in a cooling system is negligible and directly use the actual demand of the customers for cooling power to set the value of $Dem_{\mathbf{t},\mathbf{c}}$. In contrast, in a heating system, the difference between the temperature of the circulating hot water and the ambient temperature is above $60°C$. We should thus consider heat losses during distribution. The heat loss rate, denoted by $\epsilon_{\mathbf{t},HEAT}$, is the ratio of heat dissipation when the hot water is transferred from the LMES to customers at time-step $\mathbf{t}$. To take this into account, $Dem_{\mathbf{t},\mathbf{c}}$ should be set to the value of the actual demand of the customers for heating power divided by $(1 - \epsilon_{\mathbf{t},HEAT})$.

Finally, the availability of some resource commodities is limited. Let $\mathcal{C}^{R,Ph}$ be the set of resource commodities for which this limited availability comes from physical reasons (e.g. renewable energy resources such as solar radiation whose availability depends on weather conditions). In this case, the amount of commodity $\mathbf{c}$ available at time $\mathbf{t}$ is an input parameter denoted by $Conso^{max}_{\mathbf{t},\mathbf{c}}$. Moreover, the availability of some resource commodities (e.g. electricity bought from the national grid) may also be limited due to a contract signed with the outside supplier. Let $\mathcal{C}^{R,Co}$ be the set of resource commodities for which this limited availability comes from a contract negotiated with an outside provider. Let $MaxC_{\mathbf{c}}$ be the highest allowed value for this contracted maximum instantaneous power. In theory, the contracted maximum power may take any value between 0 and $MaxC_{\mathbf{c}}$. However, in practice, this one is most often fixed at a value corresponding to an integer multiple of a predefined discretization unit denoted by $C^{step}_{\mathbf{c}}$. The largest number of discrete units of power

that may be contracted for commodity $\mathbf{c}$ is thus $MaxUnits_{\mathbf{c}} = \lfloor \frac{MaxC_{\mathbf{c}}}{C_{\mathbf{c}}^{step}} \rfloor$. We denote by $SC_{\phi,\mathbf{c}}$ the cost of contracting one discrete unit of size $C_{\mathbf{c}}^{step}$ of maximum nstantaneous power.

Technologies

Let $\mathcal{P} = \{CCHP, BOILER, SMEC, ..., STO\_HEAT, ...\}$ denote the set of available energy conversion and storage technologies. Each technology is described among other by $\mathcal{N}_p^{in}$, the set of energy forms it consumes, and $\mathcal{N}_p^{out}$, the set of energy forms it produces. In what follows, for the sake of simplicity, we will assume that $|\mathcal{N}_p^{in}| = 1$ and denote by n_p^{in} the single form of energy consumed by technology p.

By using the high-level description of the LMES structure provided by the RES diagram, we may define, for each technology $p \in \mathcal{P}$, the set $\mathcal{C}_{n,p}^{in}$ (resp. $\mathcal{C}_{n,p}^{out}$) of all consumed (resp. produced) commodities belonging to the energy form $n \in \mathcal{N}$. The other way round, we may define, for each commodity $\mathbf{c} \in \mathcal{C}$, the set $\mathcal{P}_{\mathbf{c}}^{outflow}$ (resp. $\mathcal{P}_{\mathbf{c}}^{inflow}$) of technologies that consume (resp. produce) commodity $\mathbf{c}$.

As explained in Chapter 3, the set of technologies $\mathcal{P}$ can be partitioned according to the nature of the design decisions to be made. Thus, for the set $\mathcal{P}^{DIS} \subset \mathcal{P}$ of discrete technologies, the design decisions consist in selecting from a discrete list of equipment types available in the manufacturers' catalogs the number of devices of each type to install. In this case, let $l \in \{1, \cdots, L_p\}$ denote the various types of device available for technology p. The pair $\mathbf{m} = (p, l)$ thus represents a given type of energy conversion device belonging to technology p. The maximum number of devices of type $\mathbf{m}$ that may be installed is denoted by $MaxNum_{\mathbf{m}}$.

In contrast, for the set $\mathcal{P}^{CONT} \subset \mathcal{P}$ of continuous technologies, the design decisions consist in selecting the capacity of the single piece of equipment to be installed within 0 and a maximum size. In this second case, for the sake of readability, we will assume $L_p = 1$ and denote by $(p, 1)$ the single available type of equipment, the capacity of which should be determined during the design of the system. The maximum size of the installed piece of equipment is denoted by $MaxSize_{\mathbf{m}}$. Although in theory, the size of the installed device may take any value between 0 and $MaxSize_{\mathbf{m}}$, in practice, this one is most often fixed at a value corresponding to an integer multiple of a predefined discretization unit denoted by $Size_{\mathbf{m}}^{step}$. The maximum number of discrete units of capacity that may be installed for the device of type $\mathbf{m}$ is thus $MaxNum_{\mathbf{m}} = \lfloor \frac{MaxSize_{\mathbf{m}}}{Size_{\mathbf{m}}^{Step}} \rfloor$.

Let $\mathcal{M} = \{(p, l) | p \in \mathcal{P}, l \in \{1, \cdots, L_p\}\}$ be the set of all types of devices and $\mathcal{M}_{p_1} = \{(p, l) \in \mathcal{M} | p = p_1\}$ be the subset of all devices belonging to technology $p_1 \in \mathcal{P}$.

Each type of device $\mathbf{m} \in \mathcal{M}$ is associated with an installation cost $FC_{\mathbf{m}}$ and an accumulated maintenance cost $MC_{\phi,\mathbf{m}}$. The installation cost includes the purchase and construction costs, to be paid only once at the beginning of the phase in which the device is installed. Note that $FC_{\mathbf{m}}$ corresponds to the cost of installing one device $\mathbf{m} = (p, l)$ if $\mathbf{m}$ belongs to a discrete technology $p \in \mathcal{P}^{DIS}$ and to the cost of installing one discrete capacity unit of size $Size_{\mathbf{m}}^{step}$ if $\mathbf{m}$ belongs to a continuous technology $p \in \mathcal{P}^{CONT}$. Once this device (or discrete capacity unit) is installed, we may compute its actualized total maintenance cost over the interval between its installation time (beginning of phase ϕ) and the end of the system' lifetime. The corresponding cost, whose value depends on the installation phase, is defined as the accumulated maintenance cost $MC_{\phi,\mathbf{m}}$. Therefore, when installing a device of type $\mathbf{m}$ in phase ϕ, the net present value (NPV) computed at phase ϕ of the total cost is $FC_{\mathbf{m}} + MC_{\phi,\mathbf{m}}$

and the NPV computed at the beginning of the project can be obtained by multiplying $FC_{\mathbf{m}} + MC_{\phi,\mathbf{m}}$ by the design discount rate α_ϕ.

In what follows, we will also distinguish when needed between the set of conversion technologies $\mathcal{P}^{CONV} \subset \mathcal{P}$ and the set of storage technologies $\mathcal{P}^{STO} \subset \mathcal{P}$. We denote by $\mathcal{M}^{CONV}$ the set of types of energy conversion devices and $\mathcal{M}^{STO}$ the set of types of energy storage devices.

Regarding the energy conversion devices, we have a set of additional input parameters to describe the working range and energy conversion efficiency.

We thus define, for each type of conversion device $\mathbf{m} = (p, l)$ with $p \in \mathcal{P}^{CONV}$ and each form of energy $n \in \mathcal{N}_p^{out}$ produced by technology p, $P_{n,\mathbf{m}}^{min}$ (resp. $P_{n,\mathbf{m}}^{max}$) the minimum (resp. maximum) amount of energy of type n that an active device of type $\mathbf{m}$ can produce within one scheduling time-step.

As for the conversion efficiency, two main situations may occur:

- The efficiency does not depend on the load. In this case, we use the input data describing the forecasted evolution of the ambient temperature over the year to determine the value of the energy conversion efficiency to be applied to each scheduling time-step. We thus introduce a coefficient $\eta_{\mathbf{t},n,\mathbf{m}}$ giving the amount of energy of form $n \in \mathcal{C}_{n,p}^{out}$ produced per unit of consumed energy of type n_p^{in} by an active device of type $\mathbf{m}$ at scheduling time-step $\mathbf{t}$. Note that, when the energy conversion efficiency depends neither on the load nor on the ambient temperature, the value of coefficient $\eta_{\mathbf{t},n,\mathbf{m}}$ is the same for all time-steps $\mathbf{t}$.

- The efficiency depends on the load. In this case, we use the procedure described in Subsection 4.2.3 to build a piece-wise linear approximation of the performance curves provided by the manufacturer and to determine which approximation should be used at each time-step $\mathbf{t}$. Let $B_{t,n,\mathbf{m}}$ be the number of breakpoints used in the approximation of the performance curve providing the amount of energy of form $n \in \mathcal{N}_p^{out}$ produced per unit of consumed energy of type n_p^{in} by an active device of type $\mathbf{m}$ at scheduling time-step $\mathbf{t}$. Each of these breakpoints has coordinates given by $(a_{\mathbf{t},n,\mathbf{m},b}, o_{\mathbf{t},n,\mathbf{m},b})$.

4.3.2 Design variables and constraints

In order to formulate the problem as an MILP, we first introduce the decision variables relative to the system layout at each phase ϕ. Note that, to avoid ambiguity in the notation, the names of all decision variables are printed in bold letters.

For each discrete technology $p \in \mathcal{P}^{DIS}$, we introduce $\mathbf{SD}_{\phi,\mathbf{m}}$, the integer decision variable representing the number of devices of type $\mathbf{m} = (p, l) \in \mathcal{M}_p$ added to the system at the beginning of phase ϕ.

Similarly, for each continuous technologies $p \in \mathcal{P}^{CONT}$, we introduce $\mathbf{SD}_{\phi,\mathbf{m}}$, the integer decision variable indicating the number of discrete capacity units of type $\mathbf{m} = (p, l) \in \mathcal{M}_p$ added the system at the beginning of phase ϕ.

This leads to the following design constraints, stating that, for each $\mathbf{m} \in \mathcal{M}$, the total number of devices /discrete capacity units present in the system during the last investment phase Φ should be less that the maximum allowed number.

$$\sum_{\varphi=1}^{\Phi} \mathbf{SD}_{\varphi,\mathbf{m}} \leq MaxNum_{\mathbf{m}} \qquad \forall \mathbf{m} \in \mathcal{M} \tag{4.1}$$

Furthermore, we have a second set of design decisions. These decisions are relative to the resource commodities $\mathbf{c} \in \mathcal{C}^{R,Co}$ bought from an outside supplier under a

contract specifying a maximum value for the instantaneous power of this commodity that may be bought. This maximum value may be changed from one phase to the next. We thus introduce $\mathbf{Cdis}_{\phi,\mathbf{c}}$ the integer decision variable indicating the number of discrete units of power contracted at phase ϕ with the supplier of commodity $\mathbf{c}$.

This leads to the following design constraints, ensuring that, for each commodity $\mathbf{c} \in \mathcal{C}^{R,Co}$ and each phase ϕ, the contract signed with the provider of $\mathbf{c}$ stays below the maximum allowed value.

$$\mathbf{Cdis}_{\phi,\mathbf{c}} \leq MaxUnits_{\mathbf{c}} \qquad \forall \mathbf{c} \in \mathcal{C}^{R,Co}, \forall \phi \tag{4.2}$$

In what follows, we detail the variables and constraints introduced in the formulation in order to build a feasible operation schedule for each day $d \in \mathcal{D}$. We first describe the variables and constraints relative to the commodities, then the ones relative to the technologies and finally the constraints linking commodities and technologies.

4.3.3 Operation variables and constraints for commodities

Variables for the commodities

In the operational schedule relative to time-step $\mathbf{t} = (\phi, d, h)$, the total amount of commodity $\mathbf{c}$ produced (resp. consumed) by all the devices corresponding to technology $p \in \mathcal{P}$ is indicated respectively by the continuous variable $\mathbf{Inflow}_{\mathbf{t},\mathbf{c},p}$ (resp. $\mathbf{Outflow}_{\mathbf{t},\mathbf{c},p}$). Note that, thanks to this notation, each arrow in a RES diagram is associated with a flow variable: if the arrow points towards the vertical line representing the commodity, it will be associated to an **Inflow** variable and if it starts from this vertical line, it will be associated to an **Outflow** variable.

Figure 4.4 illustrates this definition on a small RES diagram in which $\mathbf{c}_1 = (n_1, s_1)$ is an intermediate commodity. The arrows in the diagram indicates that technologies p_1 and p_2 produce commodity $\mathbf{c}_1$ and technology p_3 consumes commodity $\mathbf{c}_1$. Therefore, at time-step $\mathbf{t}$, the production of commodity $\mathbf{c}_1$ by technologies p_1 and p_2 is denoted as $\mathbf{Inflow}_{\mathbf{t},\mathbf{c}_1,p_1}$ and $\mathbf{Inflow}_{\mathbf{t},\mathbf{c}_1,p_2}$, and the consumption of commodity $\mathbf{c}_1$ by p_3 is denoted as $\mathbf{Outflow}_{\mathbf{t},\mathbf{c}_1,p_3}$.

Moreover, we introduce two additional set of decision variables for each time-step: $\mathbf{Conso}_{\mathbf{t},\mathbf{c}}$, the total amount of resource commodity $\mathbf{c} \in \mathcal{C}^R$ consumed by the system within $\mathbf{t}$, and $\mathbf{Sold}_{\mathbf{t},\mathbf{c}}$, the amount of supply commodity $\mathbf{c} \in \mathcal{C}^{S,Sell}$ sold within $\mathbf{t}$.

Table 4.1 recalls the sets of indices, data and variables relative to commodities.

We now describe the constraints relative to commodities to be added to the MILP formulation. These constraints are grouped according to the type (resource, intermediate or supply) of the considered commodity.

Constraints for the resource commodities

The total amount of resource commodity $\mathbf{c} \in \mathcal{C}^R$ consumed by the system during time-step $\mathbf{t}$ is equal to the sum of the amount consumed by all the technologies in $\mathcal{P}^{outflow}_{\mathbf{c}}$.

$$\mathbf{Conso}_{\mathbf{t},\mathbf{c}} = \sum_{p \in \mathcal{P}^{outflow}_{\mathbf{c}}} \mathbf{Outflow}_{\mathbf{t},\mathbf{c},p} \qquad \forall \mathbf{c} \in \mathcal{C}^R, \forall \mathbf{t} \tag{4.3}$$

Set of indices	
$\mathcal{N}$	Set of energy forms
$\mathcal{C}$	Set of commodities
$\mathcal{C}^{R}$	Set of resource commodities
$\mathcal{C}^{R,Ph}$	Set of resource commodities with a physically limited availability
$\mathcal{C}^{R,Co}$	Set of resource commodities with a contractually limited availability
$\mathcal{C}^{I}$	Set of intermediate commodities
$\mathcal{C}^{S}$	Set of supply commodities
$\mathcal{C}^{S,Sell}$	Set of supply commodities sold to an outside buyer
$\mathcal{P}_{c}^{outflow}$	Set of technologies consuming commodity c.
$\mathcal{P}_{c}^{inflow}$	Set of technologies producing commodity c.
Data	
$EP_{\mathbf{t},\mathbf{c}}$	Buying (resp. selling) price for commodity $c \in \mathcal{C}^{R}$ (resp. $c \in \mathcal{C}^{S,Sell}$) in $\mathbf{t}$
$\epsilon_{\mathbf{t},HEAT}$	Ratio of heat dissipation in $\mathbf{t}$
$Dem_{\mathbf{t},\mathbf{c}}$	Demand for commodity $\mathbf{c} \in \mathcal{C}^{S}$ in $\mathbf{t}$
$Conso_{\mathbf{t},\mathbf{c}}^{max}$	Available amount of commodity $\mathbf{c} \in \mathcal{C}^{R,Ph}$ in $\mathbf{t}$
$C_{\mathbf{c}}^{step}$	Size of the discrete units used to define the value of the maximum instantaneous power contracted with the supplier of commodity $\mathbf{c} \in \mathcal{C}^{R,Co}$
$SC_{\phi,\mathbf{c}}$	Cost of contracting one discrete unit of size $C_{\mathbf{c}}^{step}$ of commodity $\mathbf{c} \in \mathcal{C}^{R,Co}$
$MaxUnits_{\mathbf{c}}$	Maximum number of discrete units of power that may contracted with the supplier of commodity $\mathbf{c} \in \mathcal{C}^{R,Co}$
Variables	
$\mathbf{Cdis}_{\phi,\mathbf{c}}$	Number of discrete units of power contracted at phase ϕ with the supplier of commodity $\mathbf{c} \in \mathcal{C}^{R,Co}$ (Integer)
$\mathbf{Outflow}_{\mathbf{t},\mathbf{c},p}$	Amount of commodity $\mathbf{c} \in \mathcal{C}$ consumed by technology p within $\mathbf{t}$ (Continuous)
$\mathbf{Inflow}_{\mathbf{t},\mathbf{c},p}$	Amount of commodity $\mathbf{c} \in \mathcal{C}$ produced by technology p within $\mathbf{t}$ (Continuous)
$\mathbf{Conso}_{\mathbf{t},\mathbf{c}}$	Total amount of resource commodity $\mathbf{c} \in \mathcal{C}^{R}$ consumed by the system within $\mathbf{t}$ (Continuous)
$\mathbf{Sold}_{\mathbf{t},\mathbf{c}}$	Amount of supply commodity $\mathbf{c} \in \mathcal{C}^{S,Sell}$ sold within $\mathbf{t}$ (Continuous)

TABLE 4.1: Notations for commodities

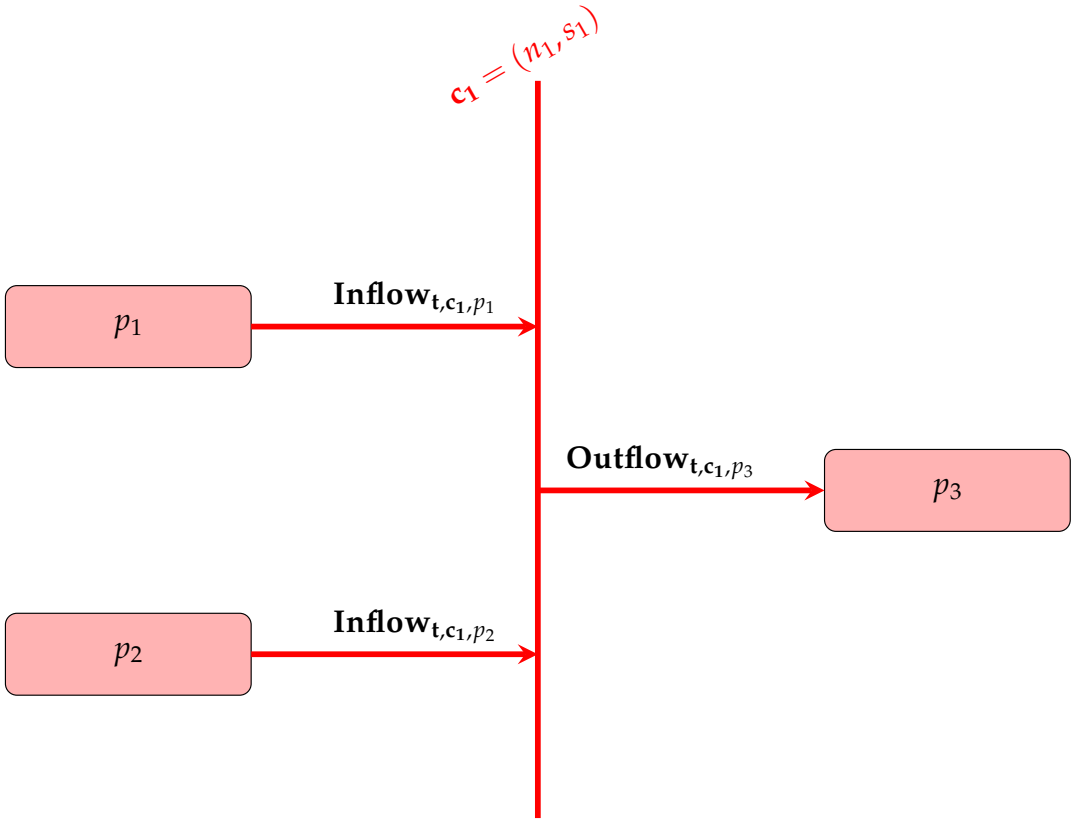


FIGURE 4.4: Inflow and outflow variables related to intermediate commodity $\mathbf{c_1} = (n_1, s_1)$

For each resource commodity $\mathbf{c} \in \mathcal{C}^{R,Ph}$, the total amount consumed by the system in each time-step $\mathbf{t}$ should stay below the maximum available amount determined by some physical limitations.

$$\mathbf{Conso_{t,c}} \leq Conso^{max}_{\mathbf{t,c}} \qquad \forall \mathbf{c} \in \mathcal{C}^{R,Ph}, \forall \mathbf{t} \tag{4.4}$$

For each resource commodity $\mathbf{c} \in \mathcal{C}^{R,Co}$, the total amount consumed by the system in each time-step $\mathbf{t}$ should stay below the maximum available amount determined by the contract signed with the outside supplier. This limit is equal to the maximum instantaneous power contracted for phase ϕ, $C^{step}_{\mathbf{c}}\mathbf{Cdis}_{\phi,\mathbf{c}}$, multiplied by the length of the time-step Δt.

$$\mathbf{Conso_{t,c}} \leq C^{step}_{\mathbf{c}}\mathbf{Cdis}_{\phi,\mathbf{c}}\Delta t \qquad \forall \mathbf{c} \in \mathcal{C}^{R,Co}, \forall \mathbf{t} \tag{4.5}$$

Constraints for the supply commodities

The LMES should be able to satisfy the customers' demand at all time. Thus, for each supply commodity $\mathbf{c} \in \mathcal{C}^{S}$, the total amount of energy produced by the technologies in $\mathcal{P}^{inflow}_{\mathbf{c}}$ in time-step $\mathbf{t}$ should be equal to the sum of the forecast energy demand coming from the buildings connected to the LMES and of the amount of energy sold to an outside buyer.

$$\sum_{p \in \mathcal{P}^{inflow}_{\mathbf{c}}} \mathbf{Inflow_{t,c,}}_{p} = Dem_{\mathbf{t,c}} + \mathbf{Sold_{t,c}} \qquad \forall \mathbf{c} \in \mathcal{C}^{S}, \forall \mathbf{t} \tag{4.6}$$

The amount of energy sold to an outside buyer is equal to 0 in time-step $\mathbf{t}$ for all supply commodities $\mathbf{c} \in \mathcal{C}^S$ not belonging to $\mathcal{C}^{S,Sell}$.

$$\mathbf{Sold}_{\mathbf{t},\mathbf{c}} = 0 \qquad \forall \mathbf{c} \in \mathcal{C}^S \setminus \mathcal{C}^{S,Sell}, \forall \mathbf{t} \tag{4.7}$$

Constraints for the intermediate commodities

An intermediate commodity $c \in \mathcal{C}^I$ is produced and consumed within the system. The total amount produced by the technologies in $\mathcal{P}^{inflow}_{\mathbf{c}}$ of production should thus be equal to the total amount consumed by the technologies in $\mathcal{P}^{outflow}_{\mathbf{c}}$ at all time.

$$\sum_{p \in \mathcal{P}^{inflow}_{\mathbf{c}}} \mathbf{Inflow}_{\mathbf{t},\mathbf{c},p} = \sum_{p \in \mathcal{P}^{outflow}_{\mathbf{c}}} \mathbf{Outflow}_{\mathbf{t},\mathbf{c},p} \qquad \forall \mathbf{c} \in \mathcal{C}^I, \forall \mathbf{t} \tag{4.8}$$

4.3.4 Operation variables and constraints for the conversion technologies

We now focus on modeling the energy conversion technologies $\mathcal{P}^{CONV}$ at the operation level. Recall that $\mathcal{M}^{CONV}$ denotes the set of all types of conversion devices: $\mathcal{M}^{CONV} = \{(p,l) : p \in \mathcal{P}^{CONV}, l = 1, \dots, L_p\}$.

For the sake of simplicity, we will use two assumptions in what follows. First, we suppose that $\mathcal{P}^{CONV} \subset \mathcal{P}^{DIS}$, i.e. that all conversion technologies are of discrete nature. This means in particular that we assume that there is a set of L_p types of devices available for each technology p and that that the total number of devices of type $\mathbf{m} = (p,l)$ available at the beginning of phase ϕ can be computed using the discrete design variables as $\sum_{\varphi=1}^{\phi} SD_{\varphi,\mathbf{m}}$.

Second, we will assume that the performance curves of all available types of conversion devices are either linear (which corresponds to a load-independent conversion efficiency coefficient) or piece-wise linear and convex. As explained in Subsection 4.2.4, this allows us to use aggregate scheduling variables indicating, for a given type of device and a time-step, the number of on/off devices and the aggregate production and consumption of energy and to avoid building a detailed schedule for each individual device.

Variables for the conversion technologies

We first introduce, for each type of energy conversion device $\mathbf{m} \in \mathcal{M}^{CONV}$, two sets of continuous variables: $\mathbf{P}^{in}_{\mathbf{t},n,\mathbf{m}}$ (resp. $\mathbf{P}^{out}_{\mathbf{t},n,\mathbf{m}}$) gives the aggregate amount of energy of form $n \in \mathcal{N}^{in}_p$ (resp. $n \in \mathcal{N}^{out}_p$) consumed (resp. produced) by the devices of type $\mathbf{m}$ in $\mathbf{t}$.

Second, the integer decision variable $\mathbf{S}_{\mathbf{t},\mathbf{m}}$ represents the number of active devices of type $\mathbf{m}$ in $\mathbf{t}$. These variables are needed among others for all devices belonging to technologies for which the minimum output power of an operating device is strictly positive.

Finally, recall that for some technologies, the devices may produce energy under various operating modes: see e.g. CCHP units which may produce either cold or hot water (in addition to electricity) or dual-mode electric chillers (DMEC) which may produce either cold water or ice. These operating modes are mutually exclusive, i.e., in a time-step $\mathbf{t}$, a given device is either off or producing using a single operating mode. Each operating mode will be identified by the distinguishing form of energy it produces. We thus introduce $\mathbf{MODE}_{\mathbf{t},n,\mathbf{m}}$ the integer variable representing the number of devices of type $\mathbf{m}$ operating in mode $n \in \mathcal{N}^{out}_p$ in $\mathbf{t}$. Note that these

variables are introduced in the formulation only for the types of devices for which several operating modes exist.

Table 4.2 recalls all the notation (set of indices, input parameters and decision variables) related to commodities.

In what follows, for the sake of simplicity, we do not provide operation constraints for all potential technologies but only for the subset of technologies mentioned in the case studies reported in Chapter 3. Note that these constraints are basically of three main types. They ensure that the schedule is feasible with respect to the limited capacity defined by design decisions, that each device is operated within its working range and that the energy consumed by an active device complies with its energy conversion efficiency.

Constraints for Combined Cooling, Heating and Power (CCHP) units

At time-step $\mathbf{t}$, the number of active CCHP units should be less than the total number of CCHP units present in the system, which is computed using the design variables $\mathbf{SD}_{\phi,\mathbf{m}}$.

$$\mathbf{S}_{\mathbf{t},\mathbf{m}} \leq \sum_{\varphi=1}^{\phi} \mathbf{SD}_{\varphi,\mathbf{m}} \qquad \forall \mathbf{m} \in \mathcal{M}_{CCHP}, \forall \mathbf{t} \tag{4.9}$$

A CCHP unit converts natural gas either into electricity and heating power or into electricity and cooling power. It thus has two operating modes that may be referred to as heating mode ($n = HEAT$) and cooling mode ($n = COLD$). At time-step $\mathbf{t}$, a CCHP unit operates using a single mode. The total number of active CCHP units in $\mathbf{t}$ is thus equal to the sum, over the two operating modes, of the number of CCHP units operating in each mode.

$$\mathbf{Mode}_{\mathbf{t},HEAT,\mathbf{m}} + \mathbf{Mode}_{\mathbf{t},COLD,\mathbf{m}} = \mathbf{S}_{\mathbf{t},\mathbf{m}} \qquad \forall \mathbf{m} \in \mathcal{M}_{CCHP}, \forall \mathbf{t} \tag{4.10}$$

Each CCHP component should be operated within its working range. The aggregate amount of energy of form $n \in \{ELEC, HEAT, COLD\}$ produced by the set of CCHP units of type $\mathbf{m}$ active at $\mathbf{t}$ should thus stay within the aggregate working ranges defined as follows.

$$P^{min}_{ELEC,\mathbf{m}} \mathbf{S}_{\mathbf{t},\mathbf{m}} \leq \mathbf{P}^{out}_{\mathbf{t},ELEC,\mathbf{m}} \leq P^{max}_{ELEC,\mathbf{m}} \mathbf{S}_{\mathbf{t},\mathbf{m}} \qquad \forall \mathbf{m} \in \mathcal{M}_{CCHP}, \forall \mathbf{t} \tag{4.11}$$

$$P^{min}_{n,\mathbf{m}} \mathbf{Mode}_{\mathbf{t},n,\mathbf{m}} \leq \mathbf{P}^{out}_{\mathbf{t},n,\mathbf{m}} \leq P^{max}_{n,\mathbf{m}} \mathbf{Mode}_{\mathbf{t},n,\mathbf{m}} \qquad \forall \mathbf{m} \in \mathcal{M}_{CCHP}, \forall \mathbf{t}, \forall n \in \{HEAT, COLD\} \tag{4.12}$$

A CCHP unit consumes a single type of energy form ($n^{in}_{CCHP} = GAS$) and its conversion efficiency is load-independent. For each type of devices $\mathbf{m} \in \mathcal{M}_{CCHP}$, this efficiency is thus characterized by three coefficients: an electricity efficiency coefficient $\eta_{\mathbf{t},ELEC,\mathbf{m}}$, an heating efficiency coefficient $\eta_{\mathbf{t},HEAT,\mathbf{m}}$ and a cooling efficiency coefficient $\eta_{\mathbf{t},COLD,\mathbf{m}}$.

The relation between the amount of consumed gas and the amount of produced electricity at time-step $\mathbf{t}$ is thus given by the following relation.

$$\mathbf{P}^{in}_{\mathbf{t},GAS,\mathbf{m}} = \frac{\mathbf{P}^{out}_{\mathbf{t},ELEC,\mathbf{m}}}{\eta_{\mathbf{t},ELEC,\mathbf{m}}} \qquad \forall \mathbf{m} \in \mathcal{M}_{CCHP}, \forall \mathbf{t} \tag{4.13}$$

Moreover, the relation between the amount of consumed gas and the amounts of heating energy (produced by the CCHP units in heating mode) and cooling energy

Set of indices	
$\mathcal{P}$	Set of technologies
$\mathcal{N}_p^{in}$	Set of energy forms consumed by technology $p \in \mathcal{P}$
$\mathcal{N}_p^{out}$	Set of energy forms produced by technology $p \in \mathcal{P}$
$\mathcal{C}_{n,p}^{in}$	Set of commodities with energy form n consumed by technology p
$\mathcal{C}_{n,p}^{out}$	Set of commodities with energy form n produced by technology p
$\mathcal{P}^{DIS}$	Set of discrete technologies
$\mathcal{P}^{CONT}$	Set of continuous technologies
$\mathcal{P}^{CONV}$	Set of energy conversion technologies
$\mathcal{P}^{STO}$	Set of energy storage technologies
$\mathcal{M}$	Set of device types
$\mathcal{M}^{CONV}$	Set of energy conversion device types
$\mathcal{M}^{STO}$	Set of energy storage device types
$\mathcal{M}_p$	Set of device types belonging to technology p
Data	
$FC_{\mathbf{m}}$	Total cost for installing one device / one discrete unit capacity of type $\mathbf{m}$
$MC_{\phi,\mathbf{m}}$	Accumulated maintenance cost of one device / one discrete unit capacity of type $\mathbf{m}$ from phase ϕ to the end of system lifetime
$MaxSize_{\mathbf{m}}$	Maximum size for the device of type $\mathbf{m} = (p, l)$ belonging to technology $p \in \mathcal{P}^{CONT}$
$Size_{\mathbf{m}}^{step}$	Size of a discrete capacity unit for the device of type $\mathbf{m} = (p, l)$ belonging to technology $p \in \mathcal{P}^{CONT}$
$MaxNum_{\mathbf{m}}$	Maximum number of devices / discrete capacity units of type $\mathbf{m}$ that may be installed
$P_{n,\mathbf{m}}^{min}$	Minimum amount of energy of type n that a device of type $\mathbf{m}$ can produce within one time-step
$P_{n,\mathbf{m}}^{max}$	Maximum amount of energy of type n that a device of type $\mathbf{m}$ can produce within one time-step
$\eta_{\mathbf{t},n,\mathbf{m}}$	Load-independent conversion efficiency of a device of type $\mathbf{m} \in \mathcal{M}^{CONV}$ producing commodity n at time-step $\mathbf{t}$
$B_{\mathbf{t},n,\mathbf{m}}$	Number of breakpoints in the piece-wise linear approximation of the performance curve for a device of type $\mathbf{m} \in \mathcal{M}^{CONV}$ producing commodity n with a load-dependent conversion efficiency at time-step $\mathbf{t}$
$a_{\mathbf{t},n,\mathbf{m},b}$	Abscissa of breakpoint b of this piece-wise linear approximation
$o_{\mathbf{t},n,\mathbf{m},b}$	Ordinate of breakpoint b of this piece-wise linear approximation
Variables	
$\mathbf{SD}_{\phi,\mathbf{m}}$	Number of devices / number of discrete units of type $\mathbf{m} \in \mathcal{M}$ newly added in the system at the beginning of phase ϕ (Integer)
$\mathbf{P}_{\mathbf{t},n,\mathbf{m}}^{in}$	Total amount of energy of type $n \in \mathcal{N}_p^{in}$ consumed by the devices of type $\mathbf{m} \in \mathcal{M}$ in $\mathbf{t}$ (Continuous)
$\mathbf{P}_{\mathbf{t},n,\mathbf{m}}^{out}$	Total amount of energy of type $n \in \mathcal{N}_p^{out}$ produced by the devices of type $\mathbf{m} \in \mathcal{M}$ in $\mathbf{t}$ (Continuous)
$\mathbf{S}_{\mathbf{t},\mathbf{m}}$	Number of devices of type $\mathbf{m} \in \mathcal{M}^{CONV}$ active in $\mathbf{t}$ (Integer)
$\mathbf{Mode}_{\mathbf{t},n,\mathbf{m}}$	Number of active devices of type $\mathbf{m} \in \mathcal{M}^{CONV}$ operating under mode $n \in \mathcal{N}_p^{out}$ in $\mathbf{t}$ (Integer)
$\mathbf{STO}_{\mathbf{t},\mathbf{m}}$	Amount of energy stored in the devices of type $\mathbf{m} \in \mathcal{M}^{STO}$ at the beginning of $\mathbf{t}$ (Continuous)

TABLE 4.2: Notations for technologies

(produced by the CCHP units in cooling mode) at time-step $\mathbf{t}$ is given by the following relation.

$$\mathbf{P}^{in}_{\mathbf{t},GAS,\mathbf{m}} \geq \frac{\mathbf{P}^{out}_{\mathbf{t},HEAT,\mathbf{m}}}{\eta_{\mathbf{t},HEAT,\mathbf{m}}} + \frac{\mathbf{P}^{out}_{\mathbf{t},COLD,\mathbf{m}}}{\eta_{\mathbf{t},COLD,\mathbf{m}}} \qquad \forall \mathbf{m} \in \mathcal{M}_{CCHP}, \forall \mathbf{t} \tag{4.14}$$

Note that in a CCHP unit, all the waste heat coming from the exhausted gas produced by the internal combustion engine does not have to be transformed into heating or cooling energy. This is why Constraints (4.14) are expressed as inequalities rather than equalities.

Constraints for air source heat pumps (ASHP)

The number of ASHPs active at time-step $\mathbf{t}$ should be less than the total number of ASHPs currently installed in the system.

$$\mathbf{S}_{\mathbf{t},\mathbf{m}} \leq \sum_{\varphi=1}^{\phi} \mathbf{SD}_{\varphi,\mathbf{m}} \qquad \forall \mathbf{m} \in \mathcal{M}_{ASHP}, \forall \mathbf{t} \tag{4.15}$$

An ASHP converts electricity into either heating power or cooling power. It thus has two operating modes which may be referred to as heating mode $n = HEAT$ or cooling mode $n = COLD$. The total number of active ASHPs in $\mathbf{t}$ is thus equal to the sum, over the two operating modes, of the number of ASHPs operating in each mode.

$$\mathbf{Mode}_{\mathbf{t},HEAT,\mathbf{m}} + \mathbf{Mode}_{\mathbf{t},COLD,\mathbf{m}} = \mathbf{S}_{\mathbf{t},\mathbf{m}} \qquad \forall \mathbf{m} \in \mathcal{M}_{ASHP}, \forall \mathbf{t} \tag{4.16}$$

Heat pumps should be operated within their working range. Recall that a heat pump does not have a minimum load rate, i.e. $P^{min}_{n,\mathbf{m}} = 0$ for $n \in \{HEAT, COLD\}$ and $\mathbf{m} \in \mathcal{M}_{ASHP}$, and can thus be operated at any value between 0 and its maximal capacity. The aggregate amount of energy of form $n \in \{HEAT, COLD\}$ produced by the ASHPs operating in mode $n \in \{HEAT, COLD\}$ in $\mathbf{t}$ should thus comply with the following inequality.

$$\mathbf{P}^{out}_{\mathbf{t},n,\mathbf{m}} \leq P^{max}_{n,\mathbf{m}} \mathbf{Mode}_{\mathbf{t},n,\mathbf{m}} \qquad \begin{array}{l} \forall \mathbf{m} \in \mathcal{M}_{ASHP}, \forall \mathbf{t}, \\ \forall n \in \{HEAT, COLD\} \end{array} \tag{4.17}$$

The conversion efficiency of an ASHP does not depend on its load. The relation between the amount of consumed electricity and the amount of produced cooling/heating energy is thus given by the following equality.

$$\mathbf{P}^{in}_{\mathbf{t},ELEC,\mathbf{m}} = \frac{\mathbf{P}^{out}_{\mathbf{t},HEAT,\mathbf{m}}}{\eta_{\mathbf{t},HEAT,\mathbf{m}}} + \frac{\mathbf{P}^{out}_{\mathbf{t},COLD,\mathbf{m}}}{\eta_{\mathbf{t},COLD,\mathbf{m}}} \qquad \forall \mathbf{m} \in \mathcal{M}_{ASHP}, \forall \mathbf{t}, \tag{4.18}$$

Constraints for boilers

The number of boilers of type $\mathbf{m}$ active at time-step $\mathbf{t}$ should be less than the total number of boilers currently installed in the system:

$$\mathbf{S}_{\mathbf{t},\mathbf{m}} \leq \sum_{\varphi=1}^{\phi} \mathbf{SD}_{\varphi,\mathbf{m}} \qquad \forall \mathbf{m} \in \mathcal{M}_{BOILER}, \forall \mathbf{t} \tag{4.19}$$

The heating power produced by a boiler can take any value between zero and its maximum capacity, i.e. $P^{min}_{HEAT,\mathbf{m}} = 0$ for any $\mathbf{m} \in \mathcal{M}_{BOILER}$. At any time-step $\mathbf{t}$,

the aggregate production of a set of active boilers of type **m** should thus lie between zero and the capacity of one boiler multiplied by the number of active boilers.

$$\mathbf{P}^{out}_{\mathbf{t},HEAT,\mathbf{m}} \leq P^{max}_{HEAT,\mathbf{m}} \mathbf{S}_{\mathbf{t},\mathbf{m}} \qquad \forall \mathbf{m} \in \mathcal{M}_{BOILER}, \forall \mathbf{t} \tag{4.20}$$

Note as boilers have a zero minimum working range, it is possible to remove variables $\mathbf{S}_{\mathbf{t},\mathbf{m}}$ from the formulation and to replace each variable $\mathbf{S}_{\mathbf{t},\mathbf{m}}$ by the expression $\sum_{\varphi=1}^{\phi} \mathbf{SD}_{\varphi,\mathbf{m}}$ in Constraints (4.20).
A boiler of type **m** converts gas into heating power with a load-independent conversion efficiency coefficient denoted by $\eta_{\mathbf{t},HEAT,\mathbf{m}}$. The aggregate amount of gas consumed by all the active boilers of type **m** active at time-step **t** is thus given by the following equality.

$$\mathbf{P}^{in}_{\mathbf{t},GAS,\mathbf{m}} = \frac{\mathbf{P}^{out}_{\mathbf{t},HEAT,\mathbf{m}}}{\eta_{\mathbf{t},HEAT,\mathbf{m}}} \qquad \forall \mathbf{m} \in \mathcal{M}_{BOILER}, \forall \mathbf{t} \tag{4.21}$$

Constraints for single-mode electric chillers (SMEC)

The number of SMECs of type **m** active at time-step **t** should be less than the total number of SMECs currently installed in the system.

$$\mathbf{S}_{\mathbf{t},\mathbf{m}} \leq \sum_{\varphi=1}^{\phi} \mathbf{SD}_{\varphi,\mathbf{m}} \qquad \forall \mathbf{m} \in \mathcal{M}_{SMEC}, \forall \mathbf{t} \tag{4.22}$$

A SMEC of type **m** should be operated within its working range defined by $[P^{min}_{COLD,\mathbf{m}}; P^{max}_{COLD,\mathbf{m}}]$, with $P^{min}_{COLD,\mathbf{m}} > 0$. The aggregate amount of cooling power ($n = COLD$) produced by the set of SMECs of type **m** active at **t** should thus lie within the aggregate working range defined as follows.

$$P^{min}_{COLD,\mathbf{m}} \mathbf{S}_{\mathbf{t},\mathbf{m}} \leq \mathbf{P}^{out}_{\mathbf{t},COLD,\mathbf{m}} \leq P^{max}_{COLD,\mathbf{m}} \mathbf{S}_{\mathbf{t},\mathbf{m}} \qquad \forall \mathbf{m} \in \mathcal{M}_{SMEC}, \forall \mathbf{t} \tag{4.23}$$

The conversion efficiency coefficient of a SMEC depends on its load and on the ambient temperature. We will therefore use the approach described in Subsection 4.2.3 to handle the set of performance curves provided by the manufacturer of each considered type of SMEC. Thus, the conversion efficiency of a SMEC of type **m** at **t** is described by a piece-wise linear function comprising $B_{\mathbf{t},COLD,\mathbf{m}}$ breakpoints with $a_{\mathbf{t},COLD,\mathbf{m},b}$ and $b_{\mathbf{t},COLD,\mathbf{m},b}$ the abscissa and ordinate of breakpoint b. Moreover, thanks to the fact that this piece-wise linear function is convex, we may use aggregate performance curves (see the discussion provided in Subsection 4.2.4 and Lemma 4.2.2) to compute the aggregate amount of input electricity $\mathbf{P}^{in}_{\mathbf{t},ELEC,\mathbf{m}}$ consumed by $\mathbf{S}_{\mathbf{t},\mathbf{m}}$ active SMECs to produce the aggregate amount of $\mathbf{P}^{out}_{\mathbf{t},COLD,\mathbf{m}}$.

$$\begin{aligned} \mathbf{P}^{in}_{\mathbf{t},ELEC,\mathbf{m}} \geq s_{\mathbf{t},COLD,\mathbf{m},b} \mathbf{P}^{out}_{\mathbf{t},COLD,\mathbf{m}} + cst_{\mathbf{t},COLD,\mathbf{m},b} \mathbf{S}_{\mathbf{t},\mathbf{m}} \quad & \forall \mathbf{m} \in \mathcal{M}_{SMEC}, \forall \mathbf{t} \\ & \forall b = 1, \ldots, B_{\mathbf{t}-1,COLD,\mathbf{m}} \end{aligned} \tag{4.24}$$

Here, $s_{\mathbf{t},COLD,\mathbf{m},b} = \frac{o_{\mathbf{t},COLD,\mathbf{m},b+1} - o_{\mathbf{t},COLD,\mathbf{m},b}}{a_{\mathbf{t},,COLD,\mathbf{m}b+1} - a_{\mathbf{t},COLD,\mathbf{m},b}}$ is the slope and $cst_{\mathbf{t},COLD,\mathbf{m},b} = o_{\mathbf{t},COLD,\mathbf{m},b} - s_{\mathbf{t},COLD,\mathbf{m},b} a_{\mathbf{t},\mathbf{m},COLD,b}$ the constant value of the line segment between breakpoints b and $b+1$ in the performance curve of as a single active device of type **m**.

Constraints for dual-mode electric chillers (DMEC)

The number of DMECs of type $\mathbf{m}$ active at time-step $\mathbf{t}$ should be less than the total number of DMECs currently installed in the system.

$$\mathbf{S}_{\mathbf{t},\mathbf{m}} \leq \sum_{\varphi=1}^{\phi} \mathbf{SD}_{\varphi,\mathbf{m}} \qquad \forall \mathbf{m} \in \mathcal{M}_{DMEC}, \forall \mathbf{t} \tag{4.25}$$

An DMEC converts electricity into either cooling power or ice. It thus has two operating modes which may be referred to as cooling mode $n = COLD$ or ice-producing mode $n = ICE$. The total number of active DMECs in $\mathbf{t}$ is thus equal to the sum, over the two operating modes, of the number of DMECs operating in each mode.

$$\mathbf{Mode}_{\mathbf{t},COLD,\mathbf{m}} + \mathbf{Mode}_{\mathbf{t},ICE,\mathbf{m}} = \mathbf{S}_{\mathbf{t},\mathbf{m}} \qquad \forall \mathbf{m} \in \mathcal{M}_{DMEC}, \forall \mathbf{t} \tag{4.26}$$

A DMEC should be operated within its working range, the bounds of which depends on its mode. Therefore, the aggregate amount of output energy $n \in \{COLD, ICE\}$ produced by the set of DMECs operating in mode n at $\mathbf{t}$ should lie within the aggregate working range defined as follows.

$$P^{min}_{n,\mathbf{m}} \mathbf{Mode}_{\mathbf{t},n,\mathbf{m}} \leq \mathbf{P}^{out}_{\mathbf{t},n,\mathbf{m}} \leq P^{max}_{n,\mathbf{m}} \mathbf{Mode}_{\mathbf{t},n,\mathbf{m}} \qquad \begin{array}{l} \forall \mathbf{m} \in \mathcal{M}_{DMEC}, \forall \mathbf{t} \\ \forall n \in \{COLD, ICE\} \end{array} \tag{4.27}$$

Similar to SMECs, the conversion efficiency coefficient of a DMEC depends on its load. It is described by its manufacturer by two set of performance curves: one set giving the amount of electricity consumed as a function of the produced cooling energy (in cooling mode) at various ambient temperatures and one set giving the amount of electricity consumed as a function of the amount of produced ice (in ice-producing mode) at various ambient temperatures. We will therefore use the approach described in Subsection 4.2.3 to handle these two sets of performance curves. Thus, the conversion efficiency of a DMEC of type $\mathbf{m}$ at $\mathbf{t}$ operating in mode $n \in \{COLD, ICE\}$ is described by a piece-wise linear function comprising $B_{\mathbf{t},n,\mathbf{m}}$ breakpoints with $a_{\mathbf{t},n,\mathbf{m},b}$ and $b_{\mathbf{t},n,\mathbf{m},b}$ the abscissa and ordinate of breakpoint b. Moreover, thanks to the fact that this piece-wise linear function is convex, we may use aggregate performance curves (see the discussion provided in Subsection 4.2.4 and Lemma 4.2.2) to compute the aggregate amount of input electricity consumed to produce the aggregate amount of $\mathbf{P}^{out}_{\mathbf{t},n,\mathbf{m}}$. This leads to the following constraints.

$$\mathbf{P}^{in}_{\mathbf{t},ELEC,\mathbf{m}} \geq s_{\mathbf{t},n,\mathbf{m},b} \mathbf{P}^{out}_{\mathbf{t},n,\mathbf{m}} + cst_{\mathbf{t},n,\mathbf{m},b} \mathbf{Mode}_{\mathbf{t},n,\mathbf{m}} \qquad \begin{array}{l} \forall \mathbf{m} \in \mathcal{M}_{SMEC}, \forall \mathbf{t} \\ \forall n \in \{COLD, ICE\} \\ \forall b = 1, \ldots, B_{\mathbf{t}-1,n,\mathbf{m}} \end{array} \tag{4.28}$$

Here, $s_{\mathbf{t},n,\mathbf{m},b} = \frac{o_{\mathbf{t},n,\mathbf{m},b+1} - o_{\mathbf{t},n,\mathbf{m},b}}{a_{\mathbf{t},n,\mathbf{m},b+1} - a_{\mathbf{t},n,\mathbf{m},b}}$ is the slope and $cst_{\mathbf{t},n,\mathbf{m},b} = o_{\mathbf{t},n,\mathbf{m},b} - s_{\mathbf{t},n,\mathbf{m},b} a_{\mathbf{t},n,\mathbf{m},b}$ the constant value of the line segment between breakpoints b and $b+1$ of the performance curve of a single DMEC operating in mode n at $\mathbf{t}$.
Note how Constraints (4.28) are expressed using variables $\mathbf{Mode}_{\mathbf{t},n,\mathbf{m}}$ instead of variables $\mathbf{S}_{\mathbf{t},\mathbf{m}}$.

4.3.5 Operation variables and constraints for the storage technologies

Let us now focus on modeling the energy storage technologies $\mathcal{P}^{STO}$ at the operation level. Recall that $\mathcal{M}^{STO}$ denotes the set of all types of storage devices: $\mathcal{M}^{STO} = \{(p,l) : p \in \mathcal{P}^{STO}, l = 1,\dots,L_p\}$.

For the sake of simplicity, we will use two assumptions in what follows. First, we suppose that $\mathcal{P}^{STO} \subset \mathcal{P}^{CONT}$, i.e. that all storage technologies are of continuous nature. This means in particular that we assume that there is at most one storage device of type $\mathbf{m} = (p,1)$ installed in the system and that the corresponding capacity available at the beginning of phase ϕ can be computed using the discrete design variables as $Size^{Step}_{\mathbf{m}} \sum_{\varphi=1}^{\phi} SD_{\varphi,\mathbf{m}}$.

Second, we assume that $|\mathcal{N}^{in}_p| = |\mathcal{N}^{out}_p| = 1$, i.e. that there is a single form of energy put into and released from the storage device. Let us denote by n^{in}_p (resp. n^{out}_p) the form of the energy put into (resp. released from) the storage device. Note that the energy is stored within the device under a form $n = n^{in}_p$. Thus, for instance, for a cold storage tank ($p = STO\_COLD$), the energy is put into and stored in the tank under the form $n^{in}_p = ICE$ and released from the tank under the form $n^{out}_p = COLD$. As for a heat storage tank ($p = STO\_HEAT$), the energy is put into, stored in and released from the tank under the form $n^{in}_p = n^{out}_p = HEAT$.

Variables for energy storage commodities

We first introduce two set of continuous variables: $\mathbf{P}^{in}_{\mathbf{t},n^{in}_p,\mathbf{m}}$ (resp. $\mathbf{P}^{out}_{\mathbf{t},n^{out}_p,\mathbf{m}}$) gives the total amount of energy of form n^{in}_p (resp. n^{out}_p) put into (resp. released from) by the storage device of type $\mathbf{m}$ in $\mathbf{t}$.

Moreover, $\mathbf{STO}_{\mathbf{t},\mathbf{m}}$ represents the amount of energy in the form n^{in}_p stored in the storage device at the beginning of time-step $\mathbf{t}$.

Constraints for energy storage commodities

For any type storage device of type $\mathbf{m}$, the amount of energy stored in the device at the beginning of time-step $\mathbf{t}$ should stay below the current installed storage capacity.

$$\mathbf{STO}_{\mathbf{t},\mathbf{m}} \leq Size^{Step}_{\mathbf{m}} \sum_{\varphi=1}^{\phi} \mathbf{SD}_{\varphi,\mathbf{m}} \qquad \forall \mathbf{m} \in \mathcal{M}^{STO}, \forall \mathbf{t} \tag{4.29}$$

Moreover, the amount of energy released from the storage device during $\mathbf{t}$ should not exceed the amount stored in the device at the beginning of $\mathbf{t}$.

$$\mathbf{P}^{out}_{\mathbf{t},n^{out}_p,\mathbf{m}} \leq \mathbf{STO}_{\mathbf{t},\mathbf{m}} \qquad \forall \mathbf{m} \in \mathcal{M}^{STO}, \forall \mathbf{t} \tag{4.30}$$

The time evolution of the energy stored in the device is described through inventory balance equations. We thus have:

$$\mathbf{STO}_{\phi,d,h,\mathbf{m}} + \mathbf{P}^{in}_{\phi,d,h,n^{in}_p,\mathbf{m}} - \mathbf{P}^{out}_{\phi,d,h,n^{out}_p,\mathbf{m}} = \mathbf{STO}_{\phi,d,h+1,\mathbf{m}} \quad \forall \mathbf{m} \in \mathcal{M}^{STO}, \forall (\phi,d) \in \mathcal{D}, \; \forall h = 0,\dots,H-1 \tag{4.31}$$

Constraints (4.31) state that the amount of energy stored at the beginning of hour $h+1$, $\mathbf{STO}_{\phi,d,h+1,\mathbf{m}}$, is equal to the amount of energy already stored at the beginning of time-step h, $\mathbf{STO}_{\phi,d,h,\mathbf{m}}$, plus the total amount of energy put into the device during time-step h minus the energy released during h.

Finally, for each representative day $(\phi, d) \in \mathcal{D}$, we have:

$$\mathbf{STO}_{\phi,d,0,\mathbf{m}} = \mathbf{STO}_{\phi,d,H,\mathbf{m}} \qquad \forall \mathbf{m} \in \mathcal{M}^{STO}, \forall (\phi, d) \in \mathcal{D} \tag{4.32}$$

In practice, the entering inventory of a given day in the scheduling horizon is imposed by the leaving inventory of the previous day. In our case, we do not consider each individual day of the scheduling horizon but rather a number of representative days which will not necessarily occur successively in practice. We thus impose that the entering inventory of a representative day $(\phi, d) \in \mathcal{D}$, $\mathbf{STO}_{\phi,d,0,\mathbf{m}}$, should be equal to the leaving inventory of the same selected day $\mathbf{STO}_{\phi,d,H,\mathbf{m}}$. This might be understood as the fact that the representative day d will be cyclically repeated $w_{\phi,d}$ times in the simplified scheduling horizon used in our optimization problem for phase ϕ.

4.3.6 Operation constraints linking commodities and technologies

On one hand, Subsection 4.3.3 presented the operation constraints relative to the commodities using variables $\mathbf{Inflow}_{\mathbf{t},\mathbf{c},p}$ or $\mathbf{Outflow}_{\mathbf{t},\mathbf{c},p}$. Each of these variables represents the aggregate amount of commodity $\mathbf{c}$ produced/consumed in $\mathbf{t}$ by all the devices belonging to technology p. Note that these **Inflow** or **Outflow** variables do not allow to follow the exact production/consumption of $\mathbf{c}$ by each type $l = 1, \ldots, L_p$ of devices belonging to technology p.

On the other hand, Subsections 4.3.4 and 4.3.5 presented the operation constraints relative to the technologies using variables $\mathbf{P}^{in}_{\mathbf{t},n,\mathbf{m}}$ and $\mathbf{P}^{out}_{\mathbf{t},n,\mathbf{m}}$. Each of these variables represents the amount of energy of form n consumed/produced in $\mathbf{t}$ by the devices of type $\mathbf{m}$. Note that these $\mathbf{P}^{in}$ and $\mathbf{P}^{out}$ variables do not allow to follow the exact consumption/production by the devices of type $\mathbf{m}$ of the various commodities $\mathbf{c} = (n, s), s = 1, \ldots, S_n$, corresponding to the energy form n.

In order to link the variables relative to commodities to the ones relative to technologies in our formulation, we assume that a given technology p has a single point of entry (resp. a single point of exit) for each form of energy $n \in \mathcal{N}^{in}_p$ (resp. $n \in \mathcal{N}^{out}_p$). The point of entry (resp. of exit) corresponding to the energy form n is connected to each commodity belonging to $\mathcal{C}^{in}_{n,p}$ (resp. $\mathcal{C}^{out}_{n,p}$). It is also connected to each set of devices $\mathbf{m} = (p, l), l = 1, \ldots, L_p$. Moreover, we assume that the flows of the various commodities entering (resp. exiting) technology p at a given point are merged and shared amongst the various types of device belonging to $\mathcal{M}_p$.

Under these assumptions, the constraints linking commodities and technologies at the operation level are flow balance equations expressed at each entry/exit point of each technology.

$$\sum_{\mathbf{c} \in \mathcal{C}^{in}_{n,p}} \mathbf{Outflow}_{\mathbf{t},\mathbf{c},p} = \sum_{\mathbf{m} \in \mathcal{M}_p} \mathbf{P}^{in}_{\mathbf{t},n,\mathbf{m}} \qquad \forall p \in \mathcal{P}, \forall n \in \mathcal{N}^{in}_p, \forall \mathbf{t} \tag{4.33}$$

$$\sum_{\mathbf{c} \in \mathcal{C}^{out}_{n,p}} \mathbf{Inflow}_{\mathbf{t},\mathbf{c},p} = \sum_{\mathbf{m} \in \mathcal{M}^p} \mathbf{P}^{out}_{\mathbf{t},n,\mathbf{m}} \qquad \forall p \in \mathcal{P}, \forall n \in \mathcal{N}^{out}_p, \forall \mathbf{t} \tag{4.34}$$

Figure 4.5 illustrates these constraints for a technology denoted by p_1. p_1 consumes energy of forms $\mathcal{N}^{in}_{p_1} = \{n_1, n_2\}$ and produces energy in the form $\mathcal{N}^{out}_p = \{n_3\}$. Two types of devices denoted by l_1 and l_2 are available for this technology.

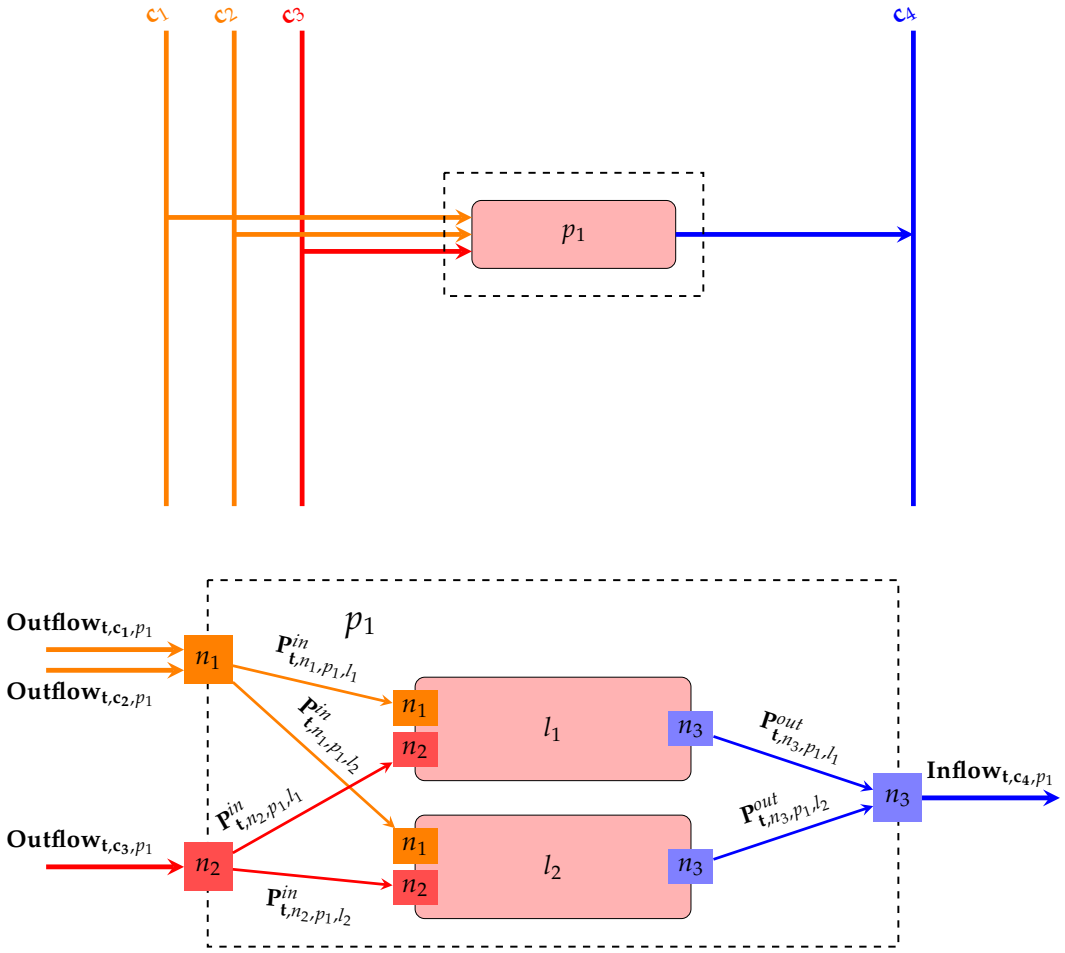


FIGURE 4.5: Example of inputs and outputs variables

In the RES diagram provided in the upper part of the figure, technology p_1 consumes three commodities $\mathbf{c}_1 = (n_1, 1)$, $\mathbf{c}_2 = (n_1, 2)$ and $\mathbf{c}_3 = (n_2, 1)$, and produces commodity $\mathbf{c}_4 = (n_3, 1)$. We thus have $\mathcal{C}^{in}_{n_1,p_1} = \{\mathbf{c}_1, \mathbf{c}_2\}$, $\mathcal{C}^{in}_{n_2,p_1} = \{\mathbf{c}_3\}$ and $\mathcal{C}^{out}_{n_3,p_1} = \{\mathbf{c}_4\}$. The lower part of the figure displays the flows of energy, together with the corresponding operation variables, at each entry/exit point of technology p_1. Observe how the entering flows of commodities corresponding to the same form n are merged at the corresponding entry point and how the resulting total amount of energy of form n is allocated between the various types of device. Similarly, the energy of form n produced by all types of devices is merged at the corresponding exit point before being allocated between the exiting flows of commodities of this form.

4.3.7 Objective Function

Finally, while optimizing the design of an LMES, we seek to minimize the net present cost of the system which comprises the actualized design and operation costs. The

objective function is thus given by:

$$\min \sum_{\phi=1}^{\Phi} \left[\alpha_\phi \left(\sum_{\mathbf{m} \in \mathcal{M}} (FC_{\mathbf{m}} + MC_{\phi,\mathbf{m}}) \mathbf{SD}_{\phi,\mathbf{m}} + \sum_{\mathbf{c} \in \mathcal{C}^{R,Co}} SC_{\phi,\mathbf{c}} \mathbf{Cdis}_{\mathbf{c},\phi} \right) \right.$$
$$\left. + \beta_\phi \sum_{d \in \mathcal{D}_\phi} w_{\phi,d} \sum_{h=0}^{H-1} \left(\sum_{c \in \mathcal{C}^R} EP_{\phi,d,h,\mathbf{c}} \mathbf{Conso}_{\phi,d,h,\mathbf{c}} - \sum_{c \in \mathcal{C}^{S,Sell}} EP_{\phi,d,h,\mathbf{c}} \mathbf{Sold}_{\phi,d,h,\mathbf{c}} \right) \right] \quad (4.35)$$

The cost computed for each phase ϕ involves two main parts. The first part corresponds to the direct cost of the design decisions, i.e. to the cost of installing and maintaining the newly added conversion and storage devices, plus the cost of the contract negotiated with the supplier of each resource commodity in $\mathcal{C}^{R,Co}$. The second part can be seen as an estimation of the cost that will have to be paid at phase ϕ to operate the designed system so as to satisfy the energy demand at all time. This operation cost is estimated by computing the weighed sum, over all representative days in $\mathcal{D}_\phi$, of the cost of the best operation schedule found for each representative day. Note that the operation cost at day (ϕ, d), is equal to the sum, over all time-steps h, of the cost of buying the resource commodities consumed during h minus the revenue obtained by selling the supply commodities during h.

4.3.8 Formulation of a generic technology

In Subsections 4.3.2, 4.3.4 and 4.3.5, we described how the design problem may be formulated as an MILP for a subset of energy conversion and storage technologies, focusing mostly on the ones encountered in our case studies. However, we did not cover all technologies that may be encountered in an LMES. Therefore, as a first step towards developing a tool able to deal with a larger class of LMESs, we define in this section a framework for formulating the problem for a generic technology. This framework can be seen as a template that should be customized to add new technologies within the optimization tool when necessary. Using the terminology commonly used in UML, the template technology can be considered as an abstract class and each technology mentioned above is a class which extends the abstract class.

For a given type of device $\mathbf{m}$ belonging to technology p, the mandatory decision variables for a general technology are the discrete design variables $\mathbf{SD}_{\phi,\mathbf{m}}$ and the continuous variables $\mathbf{P}^{in}_{\mathbf{t},n,\mathbf{m}}$ and $\mathbf{P}^{out}_{\mathbf{t},n,\mathbf{m}}$ used to follow the commodity flows between the set of devices of type $\mathbf{m} \in \mathcal{M}_p$ and the other components of the LMES.

However, these variables may not be sufficient to fully describe the detailed internal functioning of a technology. We thus may have to introduce additional variables such as the number of active devices $\mathbf{S}_{\mathbf{t},\mathbf{m}}$, the number of devices operating in a given mode $\mathbf{Mode}_{\mathbf{t},n,\mathbf{m}}$ or the amount of energy stored in the devices $\mathbf{STO}_{\mathbf{t},\mathbf{m}}$. These additional auxiliary variables will be useful to model the internal behaviour of a the technology (rather than its exchanges with the other commodities). Let $\mathbf{V}_{\mathbf{t},\mathbf{m},a}$, $a \in \{1, \cdots, A\}$, denote the set of auxiliary (integer or continuous) variables needed to model the internal functioning of a set of devices $\mathbf{m} \in \mathcal{M}_p$ at $\mathbf{t}$.

The set of constraints relative to any technology can be roughly classified into three subsets.

First, the number of devices/number of discrete capacity units of a type of device $\mathbf{m} \in \mathcal{M}_p$ installed in the system should not exceed the maximum allowed number.

$$\sum_{\varphi=1}^{\Phi} \mathbf{SD}_{\varphi,\mathbf{m}} \leq MaxNum_{\mathbf{m}} \tag{4.36}$$

Second, in general, the maximum output power of a type of device $\mathbf{m}$ producing energy of type n is constrained by the number of installed devices/discrete capacity units. We cannot give a unique and explicit expression for the constraints describing how the design decisions affect the output power of a technology as different technologies may have different rules for this maximum output power: see e.g. Constraints (4.9)-(4.12) for CCHP units and Constraints (4.29)-(4.30) for storage devices. To represent these linear constraints in a generic way, we introduce a linear function $\mathbf{O}_\mathbf{m}$. These constraints are formulated in a general form as follows:

$$\mathbf{O}_\mathbf{m}(\sum_{\varphi=1}^{\Phi} \mathbf{SD}_{\varphi,\mathbf{m}}, \mathbf{P}^{out}_{\mathbf{t},n,\mathbf{m}}, \mathbf{V}_{\mathbf{t},n,\mathbf{m},1}, \cdots, \mathbf{V}_{\mathbf{t},n,\mathbf{m},A}) \leq 0 \tag{4.37}$$

Third, there is a last set of constraints to handle the relationship between the input power and output power of a type of technology. This relationship may again take many different forms. For instance, for a heat pump or a CCHP unit, the efficiency coefficients are constant and these constraints are relatively easy to formulate. For single-mode and dual-mode electric chillers, we have a set of convex non-linear performance curves for which a piece-wise linear approximation may be built. For thermal storage, the output power is directly determined by the input power and can be chosen within a range defined by the variable of the amount of stored energy. To represent these linear constraints in a generic way, we introduce a linear function $\mathbf{E}_\mathbf{m}$. These constraints may also involve auxiliary variables and have the following general form.

$$\mathbf{E}_\mathbf{m}(\mathbf{P}^{in}_{\mathbf{t},n,\mathbf{m}}, \mathbf{P}^{out}_{\mathbf{t},n,\mathbf{m}}, \mathbf{V}_{\mathbf{t},n,\mathbf{m},1}, \cdots, \mathbf{V}_{\mathbf{t},n,\mathbf{m},A}) \leq 0 \tag{4.38}$$

4.4 Computational experiments

We would like to assess the computational performance of a mathematical programming solver using the MILP formulation provided in Section 4.3 at solving the LMES optimal design problem. To this aim, we carried out some numerical experiments using instances based on the three cases studies reported in Section 3.5.

4.4.1 Instances

District cooling system in City A

The first set of instances are based on a DCS to be built in City A, located in the south of China, using as a starting point the RES diagram provided in Figure 3.3. We thus have the set of available technologies $\mathcal{P} = \{SMEC, DMEC, STO\_COLD\}$ and the set of energy forms $\mathcal{N} = \{ELEC, COLD, ICE\}$. The set of resource commodities is $\mathcal{C}^R = \{(ELEC, 1)\}$, the set of intermediate commodities $\mathcal{C}^I = \{(ICE, 1)\}$ and the set of supply commodities $\mathcal{C}^S = \{(COLD, 1)\}$.

The expected lifetime of the DCS is $Y = 30$ years. According to the construction plan of the district, the population of the district is anticipated to increase during the first three years and stay stable afterwards. We thus introduce $\Phi = 3$ investment

phases. Phase 1 corresponds to the first year (i.e. $\underline{y}_1 = \overline{y}_1 = 1$), Phase 2 to the second year (i.e. $\underline{y}_2 = \overline{y}_2 = 2$) and Phase 3 to the last 28 years (i.e. $\underline{y}_3 = 3$ and $\overline{y}_3 = 30$). The annual discount rate used in the objective function to actualize the future costs is set to $\gamma = 8\%$.

For each phase ϕ, we have a time series representing the predicted hourly evolution of the demand for the supply commodity $(COLD, 1)$ throughout the year. These time series are obtained by combining historical data on the weather conditions and the typical cooling consumption in the area with forecasts on the number of buildings who will connect to the DCS. In order to highlight the upward trend of the resulting cooling demand, the total annual cooling demand and the maximum hourly demand expected for each of the three phases are provided in Table 4.3.

Phase	1	2	3
Length (years)	1	1	28
Total Yearly Demand (GWh)	54.7	250.0	276.8
Max Hourly Demand (MWh)	13.6	62.2	93.8

TABLE 4.3: City A project: upward trend of the annual demand

As for the price of the resource commodity $(ELEC, 1)$, it depends on the time of the day and of the commodity produced but does not vary from one day to the next. Figure 4.6 displays the evolution of the electricity price over the course of a day. It shows among others that, within a day, three types of periods may be identified. Valley periods are usually at night and early morning and correspond to a low electricity price. Peak periods happen when the demand for electricity is high, i.e. at noon and in the evening, and correspond to a high electricity price. Finally, the other periods are referred to as flat periods. It is possible to take advantage of these price variations to reduce the total energy cost of the DCS. Thus, in practice, the DCS operators often produce ice at night during valley periods, store it for a few hours and melt it to obtain chilled water at daytime during peak periods. In City A, this strategy is in fact encouraged by the grid operator which offers during valley periods an electricity price for producing ice power lower than the one for producing cooling power. In this specific case study, the price of resource commodity $(ELEC, 1)$ thus depends on the time of the day and of the type of commodity produced.

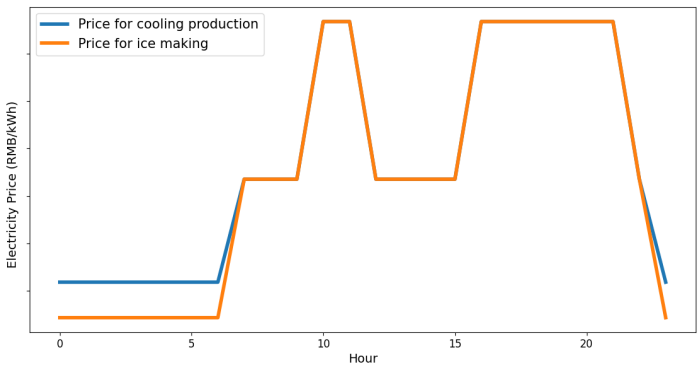


FIGURE 4.6: City A project: daily variations of the price for commodity (ELEC,1)

As the price of the resource commodity does not vary with the day in the year, we select the set of representative days $\mathcal{D}_\phi$ using only the time series representing the cooling demand. Moreover, note that City A is located in a tropical region so that customers need cooling power during most of the year. Accordingly, the number of zero-demand days per year is relatively low: over the 3 considered phases, there is on average only 35 zero-demand days per year. The set $\mathcal{D}_\phi$ should thus comprise a relatively large number of days to represent as best as possible the remaining non-zero demand days. We considered the following values of $|\mathcal{D}_\phi|$ in our numerical experiments: $\{6, 14, 22, 30, 38, 58, 78\}$. $\mathcal{D}_\phi$ comprises $|\mathcal{D}_\phi| - 4$ typical days selected through the approach described in Subsection 4.2.2 and 4 extreme days (the day with the highest total demand, the one with the highest hourly demand, the one with the lowest non-zero total demand and the one with the lowest non-zero hourly demand). In Figure 4.7, we show the cooling demand profile of the four extreme days in Phase 3. Each day is divided into $H = 24$ time steps, each one lasting $\Delta t = 1$ hour. The maximum (resp. minimum non-zero) daily and hourly demand are 2.09 GWh and 93.8 MWh (resp. 423 kWh and 1.33 kWh).

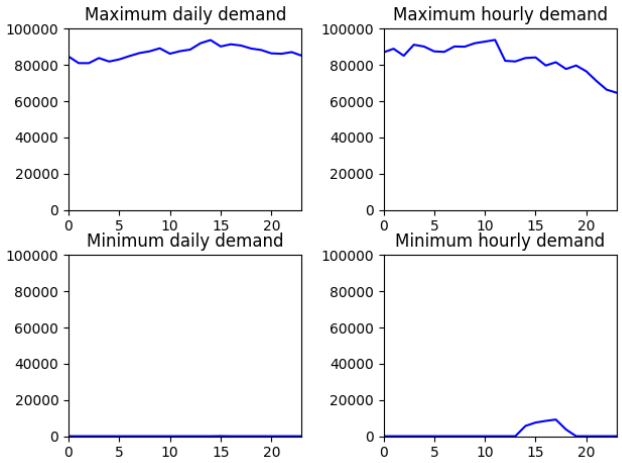


FIGURE 4.7: City A project: daily variations of the cooling demand during the 4 extreme days of Phase 3

Finally, for the resource commodity $(ELEC, 1)$, there is a contract, to be negotiated at each phase ϕ with an outside electricity provider, fixing a limit on the maximum instantaneous power that may be bought from the grid, i.e. $(ELEC, 1) \in \mathcal{C}^{R,Co}$. The maximum contracted power should stay below $MaxC_{(ELEC,1)} = 30$MW. We use a discrete unit of power equal to $C_c^{step} = 3000$kW so that $MaxUnits_{(ELEC,1)} = 10$. Moreover, the price per contracted kW for is equal to 280CNY for all phase ϕ so that $SC_{(ELEC,1),\phi} = 280 \times C_c^{step} = 840$kCNY.

Regarding the energy conversion technologies $\mathcal{P}^{CONV} = \{SMEC, DMEC\}$, we have $L_{SMEC} = 3$ types of single-mode electric chillers and $L_{DMEC} = 2$ types of dual-mode electric chillers available. For each type of device $\mathbf{m}$, the installation cost $FC_\mathbf{m}$ and the maximum production capacity $P_{n,\mathbf{m}}^{max}$, for each $n \in \mathcal{P}_p^{out}$, are shown in Table 4.4. The installation cost $FC_\mathbf{m}$ of each type is assessed according to the typical cost of chillers in the market and the regional conditions for construction. Note that the chillers of type (SMEC, 1) and (SMEC,2) have the same cooling capacity $P_{COLD,\mathbf{m}}^{max}$, but chillers of type (SMEC, 2) are less energy-efficient and less expensive than chillers of type (SMEC, 1). For each type of chiller and each commodity, the minimum output power $P_{n,\mathbf{m}}^{min}$ is equal to $0.1P_{n,\mathbf{m}}^{max}$. The performance curves of the available types of

$\mathbf{m}$	$P^{max}_{COLD,\mathbf{m}}$ (MW)	$P^{max}_{ICE,\mathbf{m}}$ (MW)	$FC_{\mathbf{m}}$ (CNY)
(SMEC,1)	9	-	1.3×10^8
(SMEC,2)	9	-	1.2×10^8
(SMEC,3)	5	-	7.9×10^7
(DMEC,1)	8	6	1.3×10^8
(DMEC,2)	5	3	8.2×10^7

TABLE 4.4: City A project: available types of chiller

SMEC and DMEC at an ambient temperature of 30°C are displayed in Figures 4.8 and 4.9. In order to build the piece-wise linear approximation of the performance curve to be used for each chiller of type $\mathbf{m}$, each form of energy n and each time-step $\mathbf{t}$, we use $B_{\mathbf{t},n,\mathbf{m}} = 4$ breakpoints. The breakpoint coordinates are obtained as described in Subsection 4.2.3.

For each device type $\mathbf{m}$, the maximum number of devices that may be installed, $MaxNum_{\mathbf{m}}$, is equal to 10 and the maintenance cost $MC_{\phi,\mathbf{m}}$ is assumed to be equal to 0 for all phase ϕ.

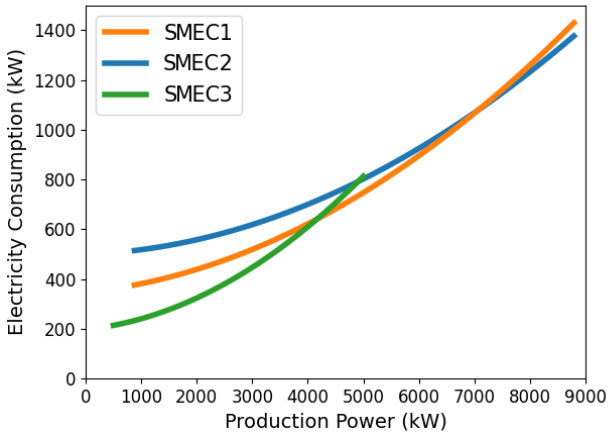


FIGURE 4.8: City A project: performance curves of the single-mode electric chillers (also termed standard chillers) at 30°C

Finally, we have a single energy storage technology and this one is of continuous nature: $\mathcal{P}^{STO} = \{STO\_COLD\} \subset \mathcal{P}^{CONT}$. The capacity of the thermal storage tank $\mathbf{m} = (STO\_COLD, 1)$ to be installed may namely take any value between 0 MWh and $MaxSize_{\mathbf{m}} = 200$ MWh. We use a discrete capacity unit $Size^{step}_{\mathbf{m}}$ equal to 5000 kWh so that $MaxNum_{\mathbf{m}} = 40$. The cost for installing a kWh of ice storage capacity is 220 CNY so that $FC_{\mathbf{m}} = 220 \times Size^{step}_{\mathbf{m}} = 1.1MCNY$. The maintenance cost $MC_{\phi,\mathbf{m}}$ is assumed to be equal to 0 for all phase ϕ.

District cooling system in City B

The second set of instances are based on a DCS to be built in City B, located in the east of China, using as a starting point the RES diagram provided in Figure 3.3. Similar to the first case study, we have the set of available technologies $\mathcal{P} = \{SMEC, DMEC, STO\_COLD\}$ and the set of energy forms $\mathcal{N} = \{ELEC, COLD, ICE\}$. The set of resource commodities is $\mathcal{C}^R = \{(ELEC, 1)\}$, the set of intermediate commodities $\mathcal{C}^I = \{(ICE, 1)\}$ and the set of supply commodities $\mathcal{C}^S = \{(COLD, 1)\}$.

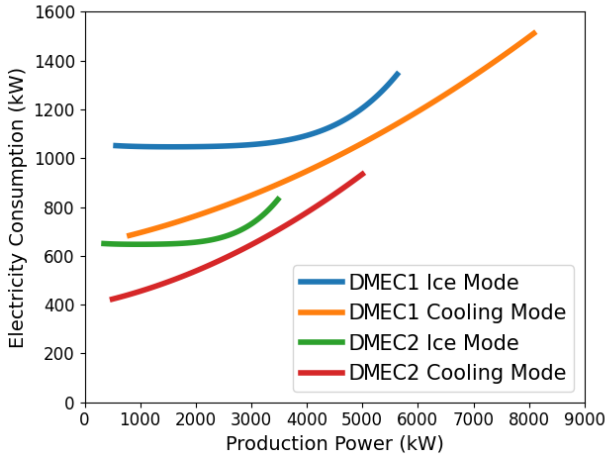


FIGURE 4.9: City A project: performance curves of the dual-mode electric chillers (also termed ice chillers) at $30°C$

The expected lifetime of the DCS is $Y = 30$ years and we anticipate the population of the district to increase during the first five years before getting stable. We thus consider $\Phi = 5$ investment phases, with the first four phases lasting one year and the fifth one lasting 26 years (i.e. $\underline{y}_5 = 5$ and $\overline{y}_5 = 30$). The annual discount rate used in the objective function to actualize the future costs is set to $\gamma = 8\%$.

Similar to City A project, we have, for each phase ϕ, a time series representing the predicted hourly evolution of the demand for the supply commodity $(COLD, 1)$ throughout the year. In order to highlight the upward trend of the resulting cooling demand, the total annual cooling demand and the maximum hourly demand expected for each of the five phases are provided in Table 4.5.

Phase	1	2	3	4	5
Length(yr)	1	1	1	1	26
Total Yearly Demand(GWh)	0.8	8.4	26.1	48.8	62.2
Max Hourly Demand(MWh)	1.4	14.0	45.3	90.0	132.0

TABLE 4.5: City B project: upward trend of the annual demand

The price of the resource commodity $(ELEC, 1)$ features some daily periodic variations but is otherwise stationary, i.e. does not vary from one day to the next. Moreover, contrary to City A project, there is no discounted electricity price to produce ice during valley periods: the electricity price thus does not depend on the produced commodity in City B project. Figure 4.10 displays the evolution of the electricity price over the course of a day.

As the price of the resource commodity does not vary with the day in the year, we select the set of representative days $\mathcal{D}_\phi$ using only the time series representing the cooling demand. City B is located in a region with a temperate climate. Thus, in contrast with City A where cooling power is needed year round, the cooling demand in City B is mainly concentrated around the summer and the average number of zero-demand days per year in City B project is high: over the 5 considered phases, there is on average 204 zero-demand days per year. This is why we use a rather small number of representative days to represent in our optimization model the remaining non-zero demand days and set the value of $|\mathcal{D}_\phi|$ to $\{6, 8, 10, 12, 14\}$. $\mathcal{D}_\phi$ comprises

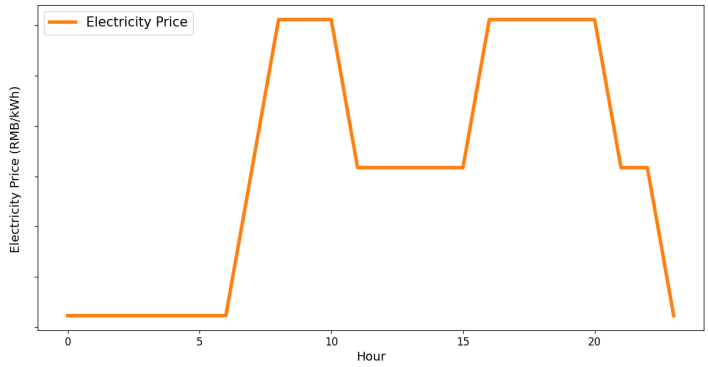


FIGURE 4.10: City B project: daily variations of the price for commodity (ELEC, 1)

$|\mathcal{D}_\phi| - 4$ typical days and 4 extreme days selected as described in Subsection 4.2.3. The profile of these four extreme days during Phase 5 is shown in Figure 4.11. Each day is divided into $H = 24$ time steps, each one lasting $\Delta t = 1$ hour. The maximum (resp. minimum non-zero) daily and hourly demand are 1.17 GWh and 132 MWh (resp. 761 kWh and 1.03 kWh).

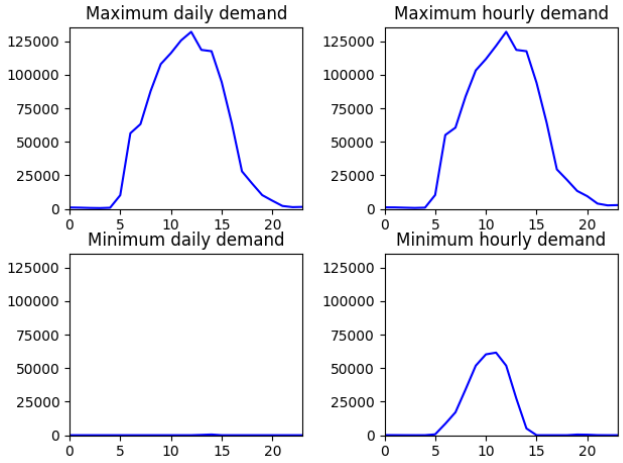


FIGURE 4.11: City B project: daily variations of the cooling demand during the 4 extreme days of Phase 5

Finally, the resource commodity $(ELEC, 1)$ belongs to $\mathcal{C}^{R,Co}$. The maximum contracted power should stay below $MaxC_{(ELEC,1)} = 30$MW. We use a discrete unit of power equal to $C_c^{step} = 3000$kW so that $MaxUnits_{(ELEC,1)} = 10$. Moreover, the price per contracted kW for is equal to 440CNY for all phase ϕ so that $SC_{(ELEC,1),\phi} = 440 \times C_c^{step} = 1.32$MCNY.

Regarding the energy conversion technologies $\mathcal{P}^{CONV} = \{SMEC, DMEC\}$, there are $L_{STDC} = 2$ types of single-mode electric chillers and $L_{ICEC} = 1$ type of dual-mode electric chiller that may be installed. These types correspond to the large-capacity chillers $(SMEC, 1)$, $(SMEC, 2)$ and $(DMEC, 1)$ used for City A project: the corresponding performance curves at an ambient temperature of 30°C are thus displayed in Figures 4.8 and 4.9. However, the installation cost $FC_{\mathbf{m}}$ in City B differs from the

	$P^{max}_{COLD,\mathbf{m}}$ (MW)	$P^{max}_{ICE,\mathbf{m}}$ (MW)	$FC_{\mathbf{m}}$ (CNY)
(SMEC,1)	9	-	1.3×10^8
(SMEC,2)	9	-	1.2×10^8
(DMEC,1)	8	6	1.3×10^8

TABLE 4.6: City B project: available types of chiller

one in City A project and is provided in Table 4.6. For each device type $\mathbf{m}$, the maximum number of devices that may be installed, $MaxNum_{\mathbf{m}}$, is equal to 10 and the maintenance cost $MC_{\phi,\mathbf{m}}$ is assumed to be equal to 0 for all phase ϕ.

Finally, we have a single energy storage technology and this one is of continuous nature: $\mathcal{P}^{STO} = \{STO\_COLD\} \subset \mathcal{P}^{CONT}$. The capacity of the thermal storage tank $\mathbf{m} = (STO\_COLD, 1)$ to be installed may namely take any value between 0 MWh and $MaxSize_{\mathbf{m}} = 180$MWh. We use a discrete capacity unit $Size^{step}_{\mathbf{m}}$ equal to 9kWh so that $MaxNum_{\mathbf{m}} = 20$. The cost for installing a kWh of ice storage capacity is 130CNY so that $FC_{\mathbf{m}} = 130 \times Size^{step}_{\mathbf{m}} = 1.17$MCNY . The maintenance cost $MC_{\phi,\mathbf{m}}$ is assumed to be equal to 0 for all phase ϕ.

Trigeneration system in City C

The last set of instances are related to the trigeneration system to be built in City C, using as a starting point the RES diagram provided in Figure 3.4. We thus have the set of available conversion technologies $\mathcal{P}^{CONV} = \{CCHP, BOILER, SMEC, DMEC, ASHP\}$, the set of available storage technologies $\mathcal{P}^{STO} = \{STO\_COLD, STO\_HEAT\}$ and the set of energy forms $\mathcal{N} = \{ELEC, GAS, HEAT, COLD, ICE\}$. The set of resource commodities is $\mathcal{C}^R = \{(ELEC,1), (GAS,1)\}$, the set of intermediate commodities $\mathcal{C}^I = \{(ELEC,2), (HEAT,1), (ICE,1)\}$ and the set of supply commodities $\mathcal{C}^S = \{(HEAT,2), (COLD,1)\}$.

The expected lifetime of the trigeneration system is $Y = 30$ years and the population of the district is expected to increase in the first four years before getting stable. Therefore, we consider $\Phi = 4$ construction phases, with the first three phases lasting one year and the fourth one lasting 27 years (i.e. $\underline{y}_4 = 4$ and $\bar{y}_4 = 30$). The annual discount rate of this project is set to $\gamma = 8\%$.

For each phase ϕ and each supply commodity, we have a time series describing the predicted hourly evolution over the year. The total yearly demand and the maximum hourly demand are given in Table 4.7 for $(HEAT,2)$ and $(COLD,1)$: in both cases, we observe a clear upward trend over the years. The heat dissipation is integrated into the numerical values of energy demand for $(HEAT,2)$.

Phase	1	2	3	4
Length (years)	1	1	1	27
Total Yearly Cooling Demand (GWh)	2.3	3.1	12.7	18.9
Max Hourly Cooling Demand (MWh)	4.6	5.8	20.2	30.2
Total Yearly Heating Demand (GWh)	14.0	24.2	31.1	38.3
Max Hourly Heating Demand (MWh)	11.0	17.3	24.6	32.6

TABLE 4.7: Trigeneration system: upward trend of the annual demand

Moreover, City C is located in central China where the climate is temperate. The cooling demand and the heating demand of the neighboring residents thus display a seasonal character and occur in two non-overlapping periods during a year. More precisely, the cooling supply period covers six months, from the 1st May to the 31st October and the heating supply period covers four months, from the 15th November to the 15th March of the next year.

The price of the resource commodity $(ELEC, 1)$ bought from the national grid varies within the day. Four periods may be distinguished: valley, flat, high and peak periods. The valley period is from 23:00 to 7:00 the next day and corresponds to a unit price of 0.27 CNY/kWh. The high period is from 9:00 to 16:00 with a price 0.85 CNY/kWh. The peak period lasts for two hours from 20:00 to 22:00 with a price 1.03 CNY/kWh. The rest of the day is termed flat period and corresponds to a unit price of 0.57 CNY/kWh. The variation of the electricity price is shown in Figure 4.12. The buying price of the resource commodity $(GAS, 1)$ natural gas is assumed to stay stable during the lifetime of the system unit and is set to is 0.27 CNY/kWh.

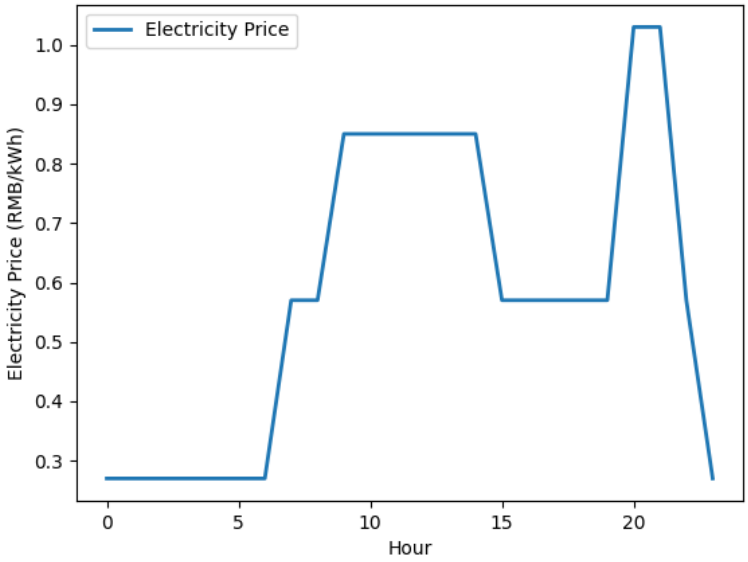
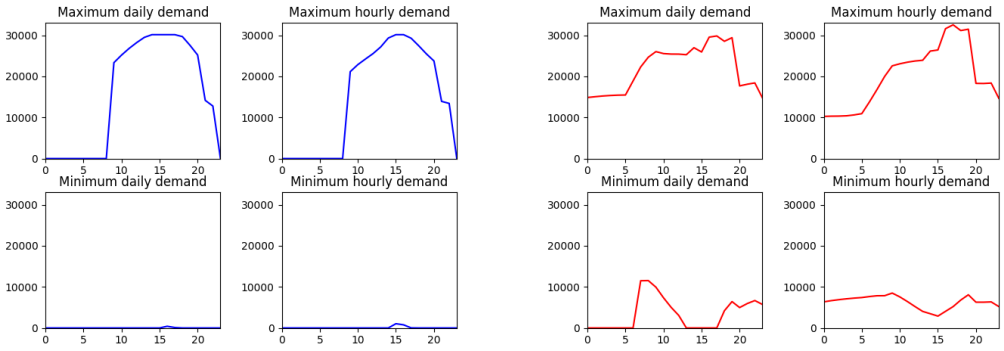


FIGURE 4.12: Trigeneration system: daily variations of the price for commodity (ELEC, 1)

As the price of the resource commodities does not vary with the day in the year, we select, for each phase ϕ, a set of representative days using only the time series corresponding to the predicted hourly evolution of the cooling demand and of the heating demand throughout the year. We thus select a set of $|\mathcal{D}_\phi| \in \{12, 16, 20, 24, 28, 32, 36\}$ representative days: $|\mathcal{D}_\phi| - 8$ typical days selected using the procedure described in Subsection 4.2.2, 4 extreme days for the cooling demand and 4 extreme days for the heating demand. Each day is divided into $H = 24$ time-steps and each time-step is of length $\Delta t = 1$ hour. Figure 4.13 displays the 4 extreme days selected for each type of demand at phase 4.

Finally, the maximum instantaneous electric power that may be bought from the electricity power is limited through a contract which may be renegotiated at the beginning of each phase. We thus have $\mathcal{C}^{R,Co} = \{(ELEC, 1)\}$. The maximum contracted power should stay below $MaxC_{(ELEC,1)} = 30$MW. We use a discrete unit of power equal to $C_c^{step} = 0.5$MW so that $MaxUnits_{(ELEC,1)} = 60$. Moreover, the price

per contracted kW for is equal to 270CNY for all phase ϕ, which gives $SC_{(ELEC,1),\phi} = 270 \times C_c^{step} = 135$kCNY.



(A) Extreme days for cooling demand (B) Extreme days for heating demand

FIGURE 4.13: Trigeneration: daily variations of the cooling and heating demand during the 4 corresponding extreme days of Phase 4

Let us now consider the available conversion technologies $\mathcal{P}^{CONV} = \{CCHP, BOILER, ASHP, SMEC, DMEC\}$.

A single type of CCHP unit ($L_{CCHP} = 1$) is available for installation in the LMES. The available model has an electricity generation capacity equal to $P^{max}_{ELEC,(CCHP,1)} = 1500$kW and thermal generation capacities equal to $P^{max}_{COLD,(CCHP,1)} = P^{max}_{HEAT,(CCHP,1)} = 1500$kW. Its minimum load rate is 50%, which means that an active CCHP unit should produce at least $P^{min}_{ELEC,(CCHP,1)} = P^{min}_{COLD,(CCHP,1)} = P^{min}_{HEAT,(CCHP,1)} = 750$kW of electric power. Its electric and thermal conversion efficiencies are load-independent and constant: $\eta_{\mathbf{t},ELEC,(CCHP,1)} = \eta_{\mathbf{t},COLD,(CCHP,1)} = \eta_{\mathbf{t},HEAT,(CCHP,1)} = 38\%$ for all $\mathbf{t}$. The purchasing price of one unit of $(CCHP, 1)$ is 7MCNY and the corresponding total installation cost is $FC_{(CCHP,1)} = 17$MCNY.

There is also a single type of boiler: $L_{CCHP} = 1$. The available model may be operated at any value between $P^{min}_{HEAT,(BOILER,1)} = 0$kW and its maximum capacity $P^{max}_{HEAT,(BOILER,1)} = 7000$kW. Its conversion efficiency coefficient is load-independent and constant: $\eta_{\mathbf{t},HEAT,(BOILER,1)} = 95\%$ for all $\mathbf{t}$. The cost for purchasing a unit of $(BOILER, 1)$ is 1.1MCNY and the total installation cost per unit is $FC_{(BOILER,1)} = 2.94$MCNY.

As for the single-mode electric chillers, $L_{SMEC} = 2$ types of devices are available. Both types of chiller have a maximum working load of $P^{max}_{COLD,(SMEC,1)} = P^{max}_{COLD,(SMEC,2)} = 3.5$MW. Chillers of type $(SMEC, 1)$ have a minimum load rate of 10% so that $P^{min}_{COLD,(SMEC,1)} = 0.35$MW. Chillers of type $(SMEC, 2)$ have a minimum load rate of 20% so that $P^{min}_{COLD,(SMEC,2)} = 0.70$MW. Their conversion efficiencies are load- and temperature dependant. They are depicted by a set of convex performance curves providing the consumed electric power as a function of the produced cooling power under different ambient temperatures. The performance curves of both types of chillers at an ambient temperature of $30°C$ are shown in Figure 4.14. The purchase price of one device of type $(SMEC, 1)$ is 2.1MCNY and the total installation cost is $FC_{(SMEC,1)} = 5.2$MCNY. The purchase price of one device of type $(SMEC, 1)$ is 1.2MCNY and the total installation cost is $FC_{(SMEC,2)} = 3.1$MCNY.

A single type of dual-mode chiller is available for installation in the system: $L_{DMEC} = 1$. Its maximum cooling capacity is $P^{max}_{COLD,(DMEC,1)} = 3.5$MW and its maximum ice-making capacity is $P^{max}_{ICE,(DMEC,1)} = 2.4$MW. The minimum load rate for

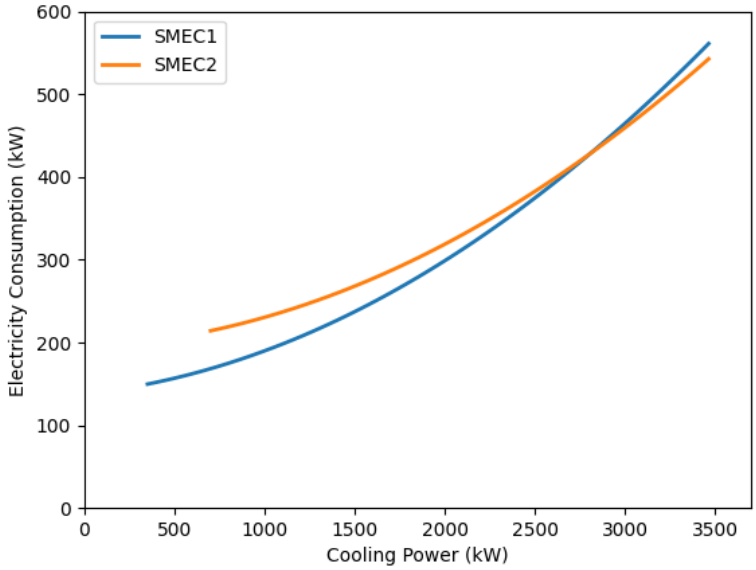


FIGURE 4.14: Trigeneration system: performance curves of the single-mode electric chillers (also termed standard chillers) at $30°C$

both modes is 30% so that $P^{min}_{COLD,(DMEC,1)} = 1.1$MW and $P^{min}_{ICE,(DMEC,1)} = 0.72$MW. Its conversion efficiency is load- and temperature dependant. It is depicted by two sets of convex performance curves providing the consumed electric power as a function of the produced cooling power (or produced ice) under different ambient temperatures. The performance curves of both modes are shown in Figure 4.14. The purchase price for one unit of $(DMEC, 1)$ is 1.8 MCNY and the total installation cost is $FC_{(DMEC,1)} = 4.4$MCNY.

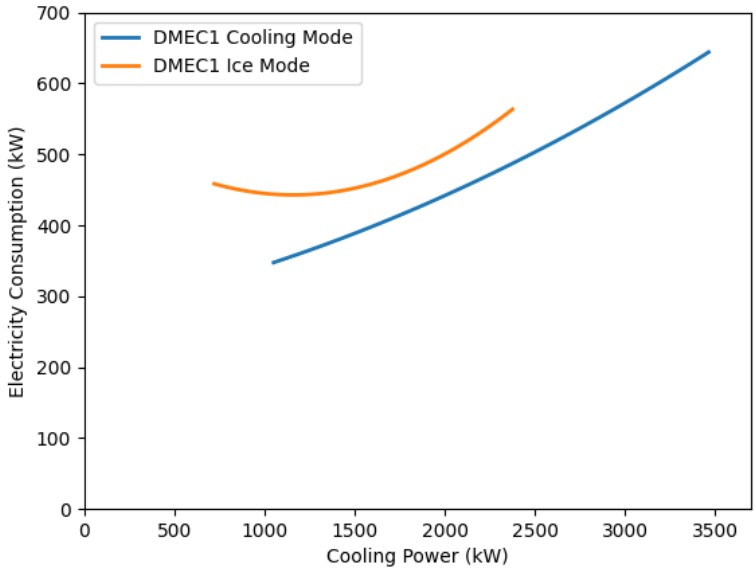


FIGURE 4.15: Trigeneration system: performance curves of the dual-mode electric chillers (also termed ice chillers) at $30°C$

There is one type of air-source heat pump available to in the catalog: $L_{ASHP} = 1$. Its heating and cooling capacities are equal to $P^{max}_{COLD,(ASHP,1)} = P^{max}_{HEAT,(ASHP,1)} =$ 1.6MW. There is no mininum loading rate so that $P^{min}_{COLD,(ASHP,1)} = P^{min}_{HEAT,(ASHP,1)} =$ 0MW. The conversion efficiency of the heating mode depends on the ambient temperature: we have $\eta_{\mathbf{t},HEAT,(ASHP,1)} = 0.064T_{outdoor,\mathbf{t}} + 1.75$, in which $T_{outdoor,\mathbf{t}}$ is the predicted ambient temperature at time-step $\mathbf{t}$. The cooling mode conversion efficiency is constant: $\eta_{\mathbf{t},COLD,(ASHP,1)} = 2$ for all $\mathbf{t}$. The purchase price for a unit of $(ASHP, 1)$ is 1.8MCNY and the total installation cost $FC_{(ASHP,1)} = 4.4$ MCNY.

The description of the available energy conversion devices is summarized in Table 4.8. The installation cost $FC_{\mathbf{m}}$ comprises both the purchase cost $PC_{\mathbf{m}}$ and the construction cost $CC_{\mathbf{m}}$ of a device. The annual maintenance cost of any installed device is set to 2.5% of its purchase cost for each year of the lifetime of the system. This gives $MC_{\mathbf{m},\phi} = 0.025 * PC_{\mathbf{m}} \sum_{\varphi=1}^{\Phi} \beta_{\varphi}$.

$\mathbf{m}$	$P^{max}_{COLD,\mathbf{m}}$	$P^{max}_{ICE,\mathbf{m}}$	$P^{max}_{HEAT,\mathbf{m}}$	$P^{max}_{ELEC,\mathbf{m}}$	$FC_{\mathbf{m}}$ (CNY)
(CCHP,1)	1.5	-	1.5	1.5	1.70×10^7
(BOILER,1)	-	-	7	-	2.94×10^6
(SMEC,1)	3.5	-	-	-	5.20×10^6
(SMEC,2)	3.5	-	-	-	3.05×10^6
(DMEC,1)	3.5	2.4	-	-	4.39×10^6
(ASHP,1)	1.6	-	1.6	-	4.38×10^6

TABLE 4.8: Trigeneration system: available types of technology

Finally, the parameters relative to the energy storage technologies $\mathcal{P}^{STO} = \{STO\_COLD, STO\_HEAT\}$ are the following.

The capacity of the heat storage tank $\mathbf{m} = (STO\_HEAT, 1)$ may be chosen at any value between 0kWh and $MaxSize_{\mathbf{m}} =$ 180MWh. We use a discrete capacity unit $Size^{step}_{\mathbf{m}}$ equal to 9000kWh so that $MaxNum_{\mathbf{m}} = 20$. The cost for installing a kWh of heat storage capacity is 17CNY, which gives $FC_{\mathbf{m}} = 17 \times Size^{step}_{\mathbf{m}} =$ 153kCNY.

The capacity of the ice storage tank $\mathbf{m} = (STO\_COLD, 1)$ may be chosen at any value between 0kWh and $MaxSize_{\mathbf{m}} =$ 180MWh. We use a discrete capacity unit $Size^{step}_{\mathbf{m}}$ equal to 9000kWh so that $MaxNum_{\mathbf{m}} = 20$. The cost for installing a kWh of ice storage capacity is 120CNY, which gives $FC_{\mathbf{m}} = 120 \times Size^{step}_{\mathbf{m}} =$ 1.08MCNY.

4.4.2 Results

In this subsection, we report the results of the computational experiments carried out on the instances based on the three case studies described above.

All tests were carried out on a machine with an Intel Xeon 2.90GHz processor with 8 cores and 16GB RAM, running under Windows 7. All the algorithms were implemented in Python. The direct resolution of the LMES design problem was done by the mathematical programming solver CPLEX 12.8 using the MILP formulation introduced in Section 4.3. We set the solver to use up to 8 threads and a maximum RAM of 11.2GB (70% of the machine's RAM) during the resolution, impose a time limit of 7200s and otherwise use the solver default settings.

The numerical results are provided in Table 4.9, Table 4.10 and Table 4.11 for the instances corresponding to City A DCS project, City B DCS project, and City C trigeneration project.

For each instance, we display the following information:

- #Var: the total number of continuous or integer decision variables involved in the MILP formulation.
- #IntVar: the total number of integer decision variables.
- #Cons: the total number of constraints.
- Gap_{LP}: the integrality gap, i.e. the percentage relative difference between the lower bound provided by the linear relaxation of the problem and the optimal solution of the problem. Note that the exact optimal objective value is found using the hierarchical decomposition algorithm and/or the generalized benders decomposition algorithm, which will be introduced in Chapters 5 and 6.
- Time: the total computation time of the algorithm before a guaranteed optimal solution of the design problem is found or the time limit is reached.
- Gap_{MIP}: the percentage residual gap between the best lower bound and the best integer feasible solution of the design problem found by the algorithm within the time limit. Note that $Gap = 0.00\%$ in case a guaranteed optimal solution of the problem was found within the time limit.

Results from Tables 4.9-4.11 first show that the direct resolution of design problems by a mathematical solver never provides a guaranteed optimal solution within the computation time limit but that near-optimal solutions (i.e. solutions displaying a residual gap below 1.0%) can be found for most instances. However, numerical difficulties arise when the number of representative days $|\mathcal{D}_\phi|$ and consequently the MILP size increase. This can be seen for the City A project instance corresponding to $|\mathcal{D}_\phi| = 78$ for which the residual gap is large (i.e. equal to 2.83%). Moreover, out-of-memory issues were encountered during the resolution for the City B project instances corresponding to $|\mathcal{D}_\phi| \in \{12, 14\}$ and for the City C project instances corresponding to $|\mathcal{D}_\phi| \in \{12, 16, 24, 28, 32, 36\}$.

Furthermore, we found that the system deployment plan given by the best found feasible solution may significantly change with the number of representative days considered for each phase in the instance. This may be due to the fact that the number of representative days is not large enough to correctly assess the operation cost (so that the design decisions depend heavily on the subset of selected representative days) or to the fact that the best found feasible solution is not the optimal one due to the existence of non-zero residual gap Gap_{MIP}. In any case, we have doubts about the quality of the system deployment plans obtained through this direct resolution of the MILP formulation by a solver.

	$\lvert\mathcal{D}_\phi\rvert$	6	14	22	30	38	58	78
MILP size	#Var	12555	29259	45963	61971	79371	121131	162984
	#IntVar	3909	9093	14277	19245	24645	37605	50565
	#Cons	21611	50411	79211	106811	136811	208811	280839
Direct resolution	Gap_{LP}	1.57%	3.26%	3.41%	1.77%	2.17%	2.06%	2.34%
	Time	7200s	7200s	7200s	7200s	7200s	7200s	7200s
	Gap_{MIP}	0.18%	0.04%	0.13%	0.39%	0.44%	0.73%	2.83%

TABLE 4.9: City A project: direct resolution by CPLEX 12.8 solver

	$\lvert\mathcal{D}_\phi\rvert$	6	8	10	12	14
MILP	#Var	13820	18380	22940	27500	32060
size	#IntVar	3625	4825	6025	7225	8425
	#Cons	23084	30764	38444	46124	53804
Direct	Gap_{LP}	5.32%	4.26%	4.58%	4.77%	4.38%
resolution	Time	7200s	7200s	7200s	2589s*	4411s*
	Gap_{MIP}	0.01%	0.02%	0.02%	0.04%	0.06%

"*" means that the computer ran out of memory before reaching the time limit.

TABLE 4.10: City B project: direct resolution by CPLEX 12.8 solver

	$\lvert\mathcal{D}_\phi\rvert$	12	16	20	24	28	32	36
MILP	#Var	76080	101424	123600	148944	175872	201216	226560
size	#IntVar	16164	21540	26244	31620	37332	42708	48084
	#Cons	86420	115220	140420	169220	199820	228620	257420
Direct	Gap_{LP}	5.11%	5.17%	5.20%	5.21%	5.15%	5.14%	5.13%
resolution	Time	5934s*	6302s*	7200s	5259s*	6155s*	5968s*	5327s*
	Gap_{MIP}	0.06%	0.26%	0.23%	0.35%	0.32%	0.16%	1.21%

"*" means that the computer ran out of memory before reaching the time limit.

TABLE 4.11: Trigeneration project: direct resolution by CPLEX 12.8 solver

4.5 Conclusion

This section investigated how the problem of optimally designing an LMES over a multi-phase investment horizon may be modeled and formulated as an MILP.

We first presented the modeling assumptions relative to the definition of the investment phases and the use of representative days to estimate the operation cost. We also explain how the convexity of the performance curves of the technologies displaying a part-load efficiency may be exploited to build operation schedules at an aggregate level (instead of operation schedules for each individual device). The main objective of this first section was to drastically reduce the number of variables and constraints needed to formulate the problem as a mathematical program.

Thanks to this, we were able to propose an MILP formulation of computationally tractable size for the problem. Note that the proposed MILP formulation relies on some additional assumptions such as: all conversion (resp. storage) technologies are of discrete (resp. continuous) nature, each technology consumes energy under a single form and all conversion technologies have linear or convex energy conversion performance curves. The proposed MILP formulation enabled us to model the three case studies reported in Chapter 3 and could be applied to a large number of LMES design problems. However, further research might be needed in order to extend this MILP formulation so as to be able to design a broader class of LMESs. In particular, it would be interesting to study LMESs involving renewable energy sources, conversion technologies consuming energy under two different forms (such as absorption chillers), conversion technologies with non-convex performance curves or inter-seasonal thermal storage in which energy is stored for long periods spanning several months. This is however left for future work.

Finally, we assessed the numerical efficiency of a mathematical solver at providing optimal solutions by directly solving the problem using the proposed MILP formulation. Our computational experiments carried out on a set of 19 large-size instances coming from the three reported real-life case studies indicated many numerical difficulties, in particular out-of-memory issues for the largest instances. This motivated us to study the development of more advanced solution approaches using decomposition-based algorithms which will be described in the next two chapters.

Chapter 5

Hierarchical decomposition algorithm

5.1 Introduction

The problem modeling proposed in Chapter 4 leads to the formulation of a large size MILP which may be solved directly by a commercial MILP solver. The computational results obtained with this direct resolution approach for our three case studies are displayed in Tables 4.11-4.11. They show that, even with the smallest number of representative days, the resolution process cannot converge before the time limit, and that, for some instances, the resolution is terminated due to out-of-memory issues. These numerical difficulties can mainly be attributed to the large number of integer variables and constraints introduced in the formulation to build the operation schedule for each representative day.

In the present chapter, we discuss a decomposition algorithm able to solve large instances of the LMES optimal design problem. This algorithm exploits the fact that the MILP problem introduced in Chapter 4 shows a special structure. Namely, the variables of the MILP model defined in Chapter 4 can be classified into two classes. The design variables refer to the decisions related to the sizing of technologies and to the contracts determining the maximum allowed power consumption of resource commodities, which are made at the beginning of each phase. The operation variables refer to decisions on the input-output of devices, inflow-outflow of commodities, the operating status of devices in each time-step. The decomposition algorithm investigated here is based on the key observation that, once the values of the design variables are determined for all construction phases, the original problem can be decomposed into a series of independent sub-problems. Each of these sub-problems takes the current value of the design variables as input and seeks to minimize, for the corresponding system layout, the operation cost for one representative or extreme day. The solution approach investigated in this chapter exploits this hierarchical structure and is based on a bi-level decomposition of the problem. The original problem is thus decomposed into an upper-level design problem and a set of lower-level operation scheduling sub-problems of smaller size. The algorithm proposed in this section can be seen as an extension of the hierarchical decomposition algorithm introduced by [Yok+15]. The proposed extended algorithm mainly aims to be more computationally efficient at solving problems involving multiple deployment phases of the LMES and short-term thermal storage capacity.

This chapter is organized as follows. In Section 5.2, we formulate the original problem in a concise and compact way to illustrate the bi-level hierarchical structure of the problem. In Section 5.3, the original hierarchical decomposition algorithm is presented. In Section 5.4, we investigate the proposed extension of the original

algorithm. The computational results obtained for our three case studies are shown in Section 5.5.

5.2 Compact reformulation and hierarchical structure of the problem

In order to explain the hierarchical decomposition algorithm in an easier and more concise way, we introduce in the present section a compact formulation of our problem.

In Chapter 4, we introduced two sets of design variables: integer variables $\mathbf{SD}_{\varphi,\mathbf{m}}$ to represent the number of devices or the number of discrete capacity units of type $\mathbf{m} \in \mathcal{M}$ added in the system at the beginning of phase ϕ, and integer variables $\mathbf{Cdis}_{\phi,\mathbf{c}}$ to represent the number of discrete units of power contracted for the resource commodity $\mathbf{c} \in \mathcal{C}^{R,Co}$ in phase ϕ. In order to obtain a more compact MILP formulation, we form a single integer variable vector by juxtaposing these design variables in a certain order. Therefore, we order the elements in the set $\mathcal{M}$ from $\mathbf{m}_1$ to $\mathbf{m}_{|\mathcal{M}|}$ and the element in the set $\mathcal{C}^{R,Cp}$ from $\mathbf{c}_1$ to $\mathbf{c}_{|\mathcal{C}^{R,Co}|}$. Let $\nu_\phi = |\mathcal{M}| + |\mathcal{C}^{R,Co}|$. The system layout at phase ϕ is then represented in the compact formulation by the following integer vector:

$$\mathbf{x}_\phi = (\sum_{\varphi=1}^{\phi} \mathbf{SD}_{\varphi,\mathbf{m}_1}, \cdots, \sum_{\varphi=1}^{\phi} \mathbf{SD}_{\varphi,\mathbf{m}_{|\mathcal{M}|}}, \mathbf{Cdis}_{\phi,\mathbf{c}_1}, \cdots, \mathbf{Cdis}_{\phi,\mathbf{c}_{|\mathcal{C}^{cont}|}}) \in \mathbb{Z}^{\nu_\phi} \tag{5.1}$$

$\mathbf{x}_\phi$ gives the total number of devices / discrete capacity units of each type of device in $\mathcal{M}$ present in the system at the beginning of phase ϕ and the number of basic instantaneous power units contracted for each resource commodity in $\mathcal{C}^{R,Co}$ at phase ϕ. The integer variable vector $\mathbf{x} = (\mathbf{x}_1, ..., \mathbf{x}_\Phi) \in \mathbb{Z}^\nu$, with $\nu = \sum_\varphi^\Phi \nu_\varphi$, describes the deployment plan over the whole investment horizon.

We denote by $\mathcal{D} = \{(\phi, d) | \phi \in \{1, ..., \Phi\}, d \in \mathcal{D}_\phi\}$ the set of all representative days. For each day $(\phi, d) \in \mathcal{D}$, we introduce the vector $\mathbf{y}_{\phi,d}$ to represent in a compact way the continuous operation variables needed to describe the operation schedule during this day. $\mathbf{y}_{\phi,d}$ thus stands for all the **Inflow**, **Outflow**, **Conso**, **Sold**, $\mathbf{P}^{in}$, $\mathbf{P}^{out}$ and **Sto** variables relative to day (ϕ, d). Similarly, we introduce the vector $\mathbf{z}_{\phi,d}$ to represent in a compact way the integer operation variables needed to describe the operation schedule during this day. $\mathbf{z}_{\phi,d}$ thus stands for all the **S** and **Mode** variables relative to day (ϕ, d). Let μ (resp. λ) denote the dimension of vector $\mathbf{y}_{\phi,d}$ (resp. $\mathbf{z}_{\phi,d}$).

Note that, in Chapter 4, we used bold letters for the decision variables and italic letters for the input parameters of the MILP formulation. Here, for the sake of readability, we change our convention. We thus use a bold letter to denote a vector of variables and an italic letter to denote a single variable. Input parameters do not appear frequently in the compact formulation, but when they appear, we will give detailed explanation to avoid ambiguity.

Let us now have a closer look at the constraints introduced in the MILP formulation described in Chapter 4. We first have a set of design constraints, i.e. constraints involving only design decisions: see Constraints (4.1)-(4.2). We then have a set of constraints coupling together the design decisions and the operation decisions relative to a given representative day (ϕ, d): see e.g. Constraints (4.9), (4.15), (4.19), (4.22), (4.25) for the energy conversion technologies, Constraints (4.29) for the energy

storage technologies and Constraints (4.5) for the resource commodities in $\mathcal{C}^{R,Co}$. Finally, all the remaining constraints may be seen as operation constraints involving only operation decisions relative to a given representative day (ϕ, d).

As a consequence, the LMES design problem can be reformulated compactly as follows:

$$\min Z_{CP} = f_0(\mathbf{x}) + \sum_{(\phi,d)\in\mathcal{D}} f_{\phi,d}(\mathbf{y}_{\phi,d}, \mathbf{z}_{\phi,d}) \tag{5.2}$$

$$h(\mathbf{x}) \leq 0 \tag{5.3}$$

$$g_{\phi,d}(\mathbf{x}_\phi, \mathbf{y}_{\phi,d}, \mathbf{z}_{\phi,d}) \leq 0 \qquad \forall (\phi,d) \in \mathcal{D} \tag{5.4}$$

$$l_{\phi,d}(\mathbf{y}_{\phi,d}, \mathbf{z}_{\phi,d}) \leq 0 \qquad \forall (\phi,d) \in \mathcal{D} \tag{5.5}$$

$$\mathbf{x} \in \mathbb{Z}^\nu \tag{5.6}$$

$$\mathbf{y}_{\phi,d} \in \mathbb{R}^\mu \qquad \forall (\phi,d) \in \mathcal{D} \tag{5.7}$$

$$\mathbf{z}_{\phi,d} \in \mathbb{Z}^\lambda \qquad \forall (\phi,d) \in \mathcal{D} \tag{5.8}$$

The objective function (5.2) involves two terms.The first term $f_0(\mathbf{x})$ corresponds to the total net present design cost of the deployment plan $\mathbf{x}$. The second term computes the sum, over all selected days, of the net present operation cost $f_{\phi,d}(\mathbf{y}_{\phi,d}, \mathbf{z}_{\phi,d})$ of the schedule computed for day (ϕ, d) and described by $(\mathbf{y}_{\phi,d}, \mathbf{z}_{\phi,d})$. Constraints (5.3) represent the design constraints. Constraints (5.4) represent the coupling constraints which link together design and operation variables. Finally, Constraints (5.5) stand for the operation constraints which are expressed using only operation variables. Constraints (5.6)-(5.8) give the domain definition of the variables. Problem (5.2)-(5.8) will be referred to as the Complete Problem (CP) in what follows.

The hierarchical decomposition algorithm uses as the upper level problem a relaxation of CP in which the design variables are kept integer but the discrete operation variables $\mathbf{z}_{\phi,d}$ are relaxed to continuous variables. The resulting problem is referred to as the semi-relaxed problem (SRP). SRP has the same number of variables and constraints as CP but a significantly smaller number of integer variables. Its resolution by an MILP solver should thus be easier than the one of CP.

$$\min Z_{SRP} = f_0(\mathbf{x}) + \sum_{(\phi,d)\in\mathcal{D}} f_{\phi,d}(\tilde{\mathbf{y}}_{\phi,d}, \tilde{\mathbf{z}}_{\phi,d}) \tag{5.9}$$

$$h(\mathbf{x}) \leq 0 \tag{5.10}$$

$$g_{\phi,d}(\mathbf{x}_\phi, \tilde{\mathbf{y}}_{\phi,d}, \tilde{\mathbf{z}}_{\phi,d}) \leq 0 \qquad \forall (\phi,d) \in \mathcal{D} \tag{5.11}$$

$$l_{\phi,d}(\tilde{\mathbf{y}}_{\phi,d}, \tilde{\mathbf{z}}_{\phi,d}) \leq 0 \qquad \forall (\phi,d) \in \mathcal{D} \tag{5.12}$$

$$\mathbf{x} \in \mathbb{Z}^\nu \tag{5.13}$$

$$\tilde{\mathbf{y}}_{\phi,d} \in \mathbb{R}^\mu \qquad \forall (\phi,d) \in \mathcal{D} \tag{5.14}$$

$$\tilde{\mathbf{z}}_{\phi,d} \in \mathbb{R}^\lambda \qquad \forall (\phi,d) \in \mathcal{D} \tag{5.15}$$

The hierarchical decomposition algorithm is based on the key observation we mentioned above: when the system deployment plan is determined, i.e., when the value of vector $\mathbf{x}$ is fixed, the problem can be decomposed into a series of independent lower-level scheduling sub-problems. Let $\mathbf{x}^\sharp$ denote a fixed system deployment plan. The operation sub-problem relative to representative day $(\phi, d) \in \mathcal{D}$, denoted by $OP_{\phi,d}(\mathbf{x}^\sharp_\phi)$, is defined as follows:

$$\min Z_{\phi,d}(\mathbf{x}_\phi^\sharp) = f_{\phi,d}(\mathbf{y}_{\phi,d}, \mathbf{z}_{\phi,d}) \tag{5.16}$$

$$g_{\phi,d}(\mathbf{x}_\phi^\sharp, \mathbf{y}_{\phi,d}, \mathbf{z}_{\phi,d}) \leq 0 \tag{5.17}$$

$$l_{\phi,d}(\mathbf{y}_{\phi,d}, \mathbf{z}_{\phi,d}) \leq 0 \tag{5.18}$$

$$\mathbf{y}_{\phi,d} \in \mathbb{R}^\mu \tag{5.19}$$

$$\mathbf{z}_{\phi,d} \in \mathbb{Z}^\lambda \tag{5.20}$$

$OP_{\phi,d}(\mathbf{x}_\phi^\sharp)$ is an MILP problem involving a number of operation variables and constraints much smaller that the one involved in CP. It is thus much easier to solve than the original complete problem CP.

5.3 Decomposition algorithm

5.3.1 Outline of the algorithm

The hierarchical decomposition algorithm proposed by [Yok+15] exploits the hierarchical structure of the problem discussed in Section 5.2. We note that a prerequisite to apply this algorithm is that all design variables should be discrete. This restriction comes from the customized branch-and-bound algorithm used to solve the upper-level design problem, which will be explained in detail in what follows.

At the upper level of the algorithm, SRP is solved by a branch-and-cut (B&C) algorithm. Since all the design variables remain discrete in SRP, each integer feasible solution $(\mathbf{x}^\sharp, \tilde{\mathbf{y}}^\sharp, \tilde{\mathbf{z}}^\sharp)$ of SRP found during the B&C tree search corresponds to a potential deployment plan described by the integer vector $\mathbf{x}^\sharp$ and complying with the design constraints (5.3) and (5.6). However, as the integer scheduling variables $z_{\phi,d}$ have been relaxed to continuous variables $\tilde{z}_{\phi,d}$ in SRP, an integer feasible solution $(\mathbf{x}^\sharp, \tilde{\mathbf{y}}^\sharp, \tilde{\mathbf{z}}^\sharp)$ of SRP is not necessarily an integer feasible solution of CP. Moreover, the feasible region of CP is a subset of the feasible region of SRP. Therefore, the objective value of SRP, $Z_{SRP}^\sharp = f_0(\mathbf{x}^\sharp) + \sum_{(\phi,d)\in\mathcal{D}} f_{\phi,d}(\tilde{\mathbf{y}}_{\phi,d}^\sharp, \tilde{\mathbf{z}}_{\phi,d}^\sharp)$, provides a lower bound to the actual net present cost $Z_{CP}^\sharp$ of the deployment plan $\mathbf{x}^\sharp$.

Hence, during the B&C tree search, each time an integer feasible solution $(\mathbf{x}^\sharp, \tilde{\mathbf{y}}^\sharp, \tilde{\mathbf{z}}^\sharp)$ of SRP is found, we should check its feasibility for CP with respect to the operation constraints (5.4)-(5.5), (5.7)- (5.8) and compute its accurate net present cost $Z_{CP}^\sharp$. This can be done by solving a sequence of independent lower level operation sub-problems. To be more precise, we solve, for each day $(\phi, d) \in \mathcal{D}$, the problem $OP_{\phi,d}(\mathbf{x}_\phi^\sharp)$, in which the operation decision variables $\mathbf{z}_{\phi,d} \in \mathbb{Z}^\lambda$ are integer. This allows us to first check the feasibility of the system layout $\mathbf{x}_\phi^\sharp$ for each selected day. Moreover, if $OP_{\phi,d}(\mathbf{x}_\phi^\sharp)$ is a feasible problem, we solve it to optimality to obtain an optimal operational schedule $(\mathbf{y}_{\phi,d}^\sharp, \mathbf{z}_{\phi,d}^\sharp)$, together with its corresponding exact operation cost $f_{\phi,d}(\mathbf{y}_{\phi,d}^\sharp, \mathbf{z}_{\phi,d}^\sharp)$.

If all sub-problems $OP_{\phi,d}(\mathbf{x}_\phi^\sharp)$, $(\phi, d) \in \mathcal{D}$ are feasible, $(\mathbf{x}^\sharp, \mathbf{y}^\sharp, \mathbf{z}^\sharp)$ is a feasible solution of the original CP with a net present cost $Z_{CP}^\sharp = f_0(\mathbf{x}^\sharp) + \sum_{(\phi,d)\in\mathcal{D}} f_{\phi,d}(\mathbf{y}_{\phi,d}^\sharp, \mathbf{z}_{\phi,d}^\sharp)$. Let Z_{CP}^{best} denote the value of the best feasible solution of CP $(\mathbf{x}^{best}, \mathbf{y}^{best}, \mathbf{z}^{best})$ found since the beginning of the algorithm, which is also called the incumbent of CP. If $Z_{CP}^\sharp$ is lower than Z_{CP}^{best}, we set $(\mathbf{x}^{best}, \mathbf{y}^{best}, \mathbf{z}^{best})$ to $(\mathbf{x}^\sharp, \mathbf{y}^\sharp, \mathbf{z}^\sharp)$ and Z_{CP}^{best} to $Z_{CP}^\sharp$ and

add Z_{CP}^{best} as a new upper cutoff value for SRP in the upper-level B&C algorithm SRP, i.e. add the constraint $f_0(\mathbf{x}) + \sum_{(\phi,d)\in\mathcal{D}} f_{\phi,d}(\tilde{\mathbf{y}}_{\phi,d}, \tilde{\mathbf{z}}_{\phi,d}) \leq Z_{CP}^{best}$ to the formulation of SRP.

Note that, whether the feasible solution $(\mathbf{x}^\sharp, \mathbf{y}^\sharp, \mathbf{z}^\sharp)$ is accepted as a new incumbent of CP or not, the potential deployment plan $(\mathbf{x}^\sharp, \tilde{\mathbf{y}}^\sharp, \tilde{\mathbf{z}}^\sharp)$ is rejected in the B&C search tree used to solve SRP at the upper level. Namely, the solver should not take the integer feasible solution $(\mathbf{x}^\sharp, \tilde{\mathbf{y}}^\sharp, \tilde{\mathbf{z}}^\sharp)$ as the new incumbent of SRP but rather should continue branching from this node.

This rejection is necessary for two reasons. First, if the current integer feasible solution $(\mathbf{x}^\sharp, \tilde{\mathbf{y}}^\sharp, \tilde{\mathbf{z}}^\sharp)$ is not rejected as an incumbent of SRP, the upper-level B&C algorithm may use the corresponding objective value $Z_{SRP}^\sharp$ as the current best known upper bound and decide to prune nodes in the subsequent iterations based on this value. However, $Z_{SRP}^\sharp$ is a lower bound of the actual net present cost and corresponds only to an approximate estimation of the cost performance of the potential deployment plan $\mathbf{x}^\sharp$. It therefore should not be used to compare $\mathbf{x}^\sharp$ with other potential deployment plans and to decide which nodes to prune in the B&C tree. Second, let $n^\sharp$ denote the node of the B&C tree on which the potential deployment plan $(\mathbf{x}^\sharp, \tilde{\mathbf{y}}^\sharp, \tilde{\mathbf{z}}^\sharp)$ is found and let $\mathcal{F}^\sharp$ be the subset of the feasible domain of SRP defined by the branching decisions made up to node $n^\sharp$. For any solution $(\mathbf{x}, \tilde{\mathbf{y}}, \tilde{\mathbf{z}})$ in $\mathcal{F}^\sharp$, we have $Z_{SRP} \geq Z_{SRP}^\sharp$. It is therefore not possible to find in $\mathcal{F}^\sharp$ an integer feasible solution of SRP with a value Z_{SRP} lower than $Z_{SRP}^\sharp$. However, as Z_{SRP} is only an approximation of the actual net present value of a deployment plan, we may find in $\mathcal{F}^\sharp$ an integer feasible solution $(\mathbf{x}^\bullet, \tilde{\mathbf{y}}^\bullet, \tilde{\mathbf{z}}^\bullet)$ of SRP corresponding to an integer feasible solution $(\mathbf{x}^\bullet, \mathbf{y}^\bullet, \mathbf{z}^\bullet)$ of CP such that $Z_{CP}^\bullet < Z_{CP}^\sharp$. In other words, we should not decide to stop the exploration in $\mathcal{F}^\sharp$ by considering only the fact that $(\mathbf{x}^\sharp, \mathbf{y}^\sharp, \mathbf{z}^\sharp)$ is integer feasible for CP as better integer feasible solutions of CP may be found in $\mathcal{F}^\sharp$. Rejecting $(\mathbf{x}^\sharp, \tilde{\mathbf{y}}^\sharp, \tilde{\mathbf{z}}^\sharp)$ as the new incumbent solution for SRP thus prevents the solver from using the value $Z_{SRP}^\sharp$ to wrongly prune nodes in the search tree, from prematurely stopping the exploration in $\mathcal{F}^\sharp$ and thus from neglecting other integer feasible solutions of CP contained in $\mathcal{F}^\sharp$. This rejection is realized by a built-in function of the mathematical solver

In fact, the upper-level B&C algorithm should never accept any incumbent of SRP in the B&C search tree and the decision to prune nodes in this tree should only be made by comparing the corresponding lower bound with the value Z_{CP}^{best} of the incumbent of CP. More precisely, a node in the B&C search tree of the upper-level problem should be pruned only if its corresponding deployment plan $(\mathbf{x}, \tilde{\mathbf{y}}, \tilde{\mathbf{z}})$ is such that $Z_{SRP} > Z_{CP}^{best}$ or if it is infeasible with respect to SRP. Therefore, a node n with objective value $Z_{SRP} < Z^{best}$ remains active and may be used for branching even if $(\mathbf{x}, \tilde{\mathbf{y}}, \tilde{\mathbf{z}})$ is an integer feasible solution of SRP. In this case, the branching will be carried out by a built-in function of the mathematical solver on an design decision variable $x_{\phi,i}$ which already has an integer value at node n. This involves that an integer potential deployment plan similar to the one found at node n may be found while exploring the sub-tree of the B&C tree rooted in n. In order to avoid evaluating multiple times the same deployment plan, we add to the formulation of SRP a no-good cut ensuring that the integer potential deployment plan found at a node of the B&C search tree is excluded from the feasible domain of SRP. This will be explained in more detail in the next two sections.

When the B&B algorithm in the upper level problem terminates, i.e. when all nodes have been closed, the incumbent of CP $(\mathbf{x}^{best}, \mathbf{y}^{best}, \mathbf{z}^{best})$ gives an optimal solution of CP.

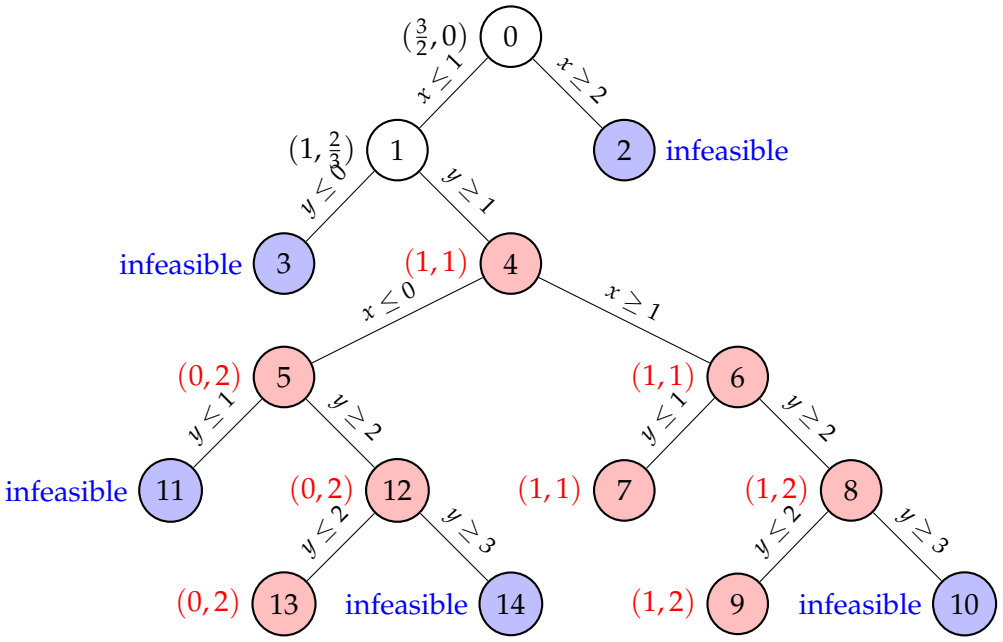


FIGURE 5.1: Example: Resolution of an integer program by a B&C algorithm with systematic rejection of the incumbent

5.3.2 Incumbent rejection

In this subsection, we use a small example to clarify what happens when the solver rejects the current integer feasible solution as incumbent and continues branching from a node where an integer feasible solution has been found.

We consider the following small integer program.

$$\min \quad y \tag{5.21}$$

$$s.t. \quad \frac{4}{3}x + y \geq 2 \tag{5.22}$$

$$-\frac{1}{3}x + y \leq 2 \tag{5.23}$$

$$x \leq \frac{3}{2} \tag{5.24}$$

$$(x, y) \in \mathbb{Z}^2 \tag{5.25}$$

The feasible domain of the linear relaxation of (5.21)- (5.25) is a triangle defined by the vertices $(0,2)$, $(1.5,0)$ and $(1.5,2.5)$. It comprises three integer feasible solutions: $(1,1)$, $(0,2)$ and $(1,2)$. As the objective function consists in minimizing the value of the integer variable y, the optimal integer solution is $(1,1)$.

Let us now solve Problem (5.21)-(5.25) with a B&C algorithm in which no integer feasible solution is accepted as incumbent during the solving process. The corresponding search tree is shown in Figure 5.1.

We first note that, if we accept the current integer feasible solution as the incumbent when suitable, the B&C algorithm terminates at Node 4 with the optimal solution $(1,1)$ and optimal value 1. However, if we reject the integer feasible solution $(1,1)$ as incumbent, the algorithm will continue and will branch from Node 4 by

adding bounds on the variable x, which already has an integer value at Node 4. Contrary to the traditional branching operation applied on fractional-valued variables, this branching will not exclude the current integer feasible solution. Therefore, we can observe that Node 6, a descendent node of Node 4, also provides the integer feasible solution $(1, 1)$. The branching actions will stop either when the LP relaxation is infeasible (Nodes 10, 11 and 14) or when the value of each variable is restrained to take a single value (Nodes 7, 9 and 13), i.e. when the feasible domain of the problem has been partitioned into subsets involving at most one integer feasible solution. As a result, the modified B&C tree explores more nodes than a standard B&C tree and returns no feasible solution at the end of exploration. Moreover, all the integer feasible solutions contained in the feasible space are considered as integer feasible solution on at least one node of the search tree.

Thanks to this small example, we see how rejecting the incumbent still guarantees the finite and optimal convergence of the B&C algorithm. First, even if we reject all found integer feasible solutions as incumbent, the B&C search tree terminates in a finite number of steps, when enough branching decisions have been made to ensure that the feasible domain has been partitioned into subsets involving at most one integer feasible solution, so that the termination of the algorithm is guaranteed theoretically. Second, all the integer feasible solutions are explored at least once during the B&C search tree. As we can see in the example, even though an integer feasible solution is found in Node 4, all the other integer feasible solutions contained in the feasible region defined by the branching decisions made up to Node 4 appear on at least one node of the sub-tree rooted at Node 4. This ensures that the optimal solution is found and evaluated at least once during the B&C search.

Let us now come back to the hierarchical decomposition algorithm mentioned above.

The simple example provided above first shows that even if we do not cut off open nodes based on a comparison between their lower bound and the current incumbent, the B&C algorithm with a systematic incumbent rejection still terminates in a finite number of steps, basically after having carried out a trivial enumeration of all integer feasible solutions. The hierarchical decomposition algorithm can be seen as an extension of this trivial enumerating algorithm in which we close nodes by comparing their lower bound with the current best known integer value for CP. The hierarchical decomposition algorithm will thus terminate in a finite number of steps but is likely to be more computationally efficient that the basic algorithm based on a full enumeration discussed above.

Moreover, thanks to the systematic incumbent rejection, when an integer feasible solution of SRP has been found at a node, we will continue branching from that node. This ensures that all the integer feasible solutions of SRP, and consequently all integer feasible solutions of CP, compatible with the branching decisions made up to that point will be explicitly considered and evaluated through the resolution of a series of operation sub-problems. The only exception corresponds to the case where the lower bound computed at a node is larger than the current best known integer value for CP. In this case, we will close the node and by doing so, we will eliminate some integer feasible solutions of CP. But as these solutions cannot be better than the current best known solution of CP, this does not affect the optimal convergence of the algorithm.

5.3.3 No-good cuts

However, the systematic rejection of the incumbent leads to some loss of computational efficiency as the same integer feasible solution of SRP (and the same integer feasible solution of CP) may be considered and evaluated multiple times during the resolution process.

To prevent this from happening, 'no-good cuts' may be added to the formulation of SRP [Wol20]. No-good cuts are applied on a set of binary variables, so we need to carry out a binary expansion of the integer design variables. More precisely, for each component $x_{\phi,i}$ of the vector $\mathbf{x}_\phi$, we introduce a set of additional binary variables, $xbin_{\phi,i,\theta}$, with $\theta \in \{0, ..., \Theta_{\phi,i}\}$, and define the binary expansion of $x_{\phi,i}$ as:

$$x_{\phi,i} = \sum_{\theta=0}^{\Theta_{\phi,i}} 2^\theta xbin_{\phi,i,\theta} \tag{5.26}$$

Here, $xbin_{\phi,i,\theta} = 1$ if coefficient 2^θ is used to compute the value of $x_{\phi,i}$ and 0 otherwise. Note that $\Theta_{\phi,i}$ is set to $\Theta_{\phi,i} = \lceil \log_2 x^{max}_{\phi,i} \rceil$, where $x^{max}_{\phi,i}$ is an upper bound of the variable $x_{\phi,i}$. This binary expansion ensures that all possible integer values of $x_{\phi,i}$ within its range has a one-to-one correspondence to the binary representation $(xbin_{\phi,i,0}, \cdots, xbin_{\phi,i,\Theta_{\phi,i}})$.

Now, if a given deployment plan $\mathbf{x}^1 = (\mathbf{x}^1_1, ..., \mathbf{x}^1_\Phi)$ is rejected, we add a 'no-good cut' using the binary expansion variables to exclude this potential deployment plan from the feasible space of SRP. This cut is obtained by partitioning the set of all indices used in the binary expansion of each vector $\mathbf{x}_\phi$, i.e., the set $\mathcal{I}_\phi = \{(i,\theta), i \in \{1, \cdots, \nu_\phi\}, \theta \in \{0, \cdots, \Theta_{\phi,i}\}\}$, into two sub-sets according to the value of the binary expansion of the given deployment plan $\mathbf{xbin^1}$: $\mathcal{I}^0_\phi = \{(i,\theta) \in I | xbin^1_{\phi,i,\theta} = 0\}$ and $\mathcal{I}^1_\phi = \{(i,\theta) \in I | xbin^1_{\phi,i,\theta} = 1\}$. Using this notation, we formulate the no-good cut as follows:

$$\sum_{\varphi=1}^{\Phi} \left(\sum_{(i,\theta)\in\mathcal{I}^0_\varphi} xbin_{\varphi,i,\theta} + \sum_{(i,\theta)\in\mathcal{I}^1_\varphi} (1 - xbin_{\varphi,i,\theta}) \right) \geq 1 \tag{5.27}$$

Constraint (5.27) enforces that any potential deployment plan considered differs from $\mathbf{x}^1$ with respect to at least one of the binary variables involved in the binary expansion of $\mathbf{x}$. Adding Constraint (5.27) to the formulation of SRP thus ensures that any potential deployment plan encountered during the upper level branch-and-cut search will appear only once, which allows us to avoid some repetitive computations.

5.4 Algorithmic improvement

The hierarchical decomposition algorithm discussed in Section 5.3 does not always outperform the generic B&C algorithm embedded in CPLEX 12.8 at solving the design problem, as will be shown by the numerical results reported in Section 5.5. There are two reasons for this inefficiency. Firstly, the presence of multiple construction phases leads to an increase in the number of potential deployment plans to be explored and consequently to an increase in the computational burden. Second, since all the operation discrete variables are relaxed to continuous variables in SRP, an integer feasible solution of SRP is not necessarily feasible with respect to the original problem CP. This means that the upper level algorithm explores a significant

number of potential deployment plan which are feasible for SRP but infeasible for CP. In this section, we will investigate two ways to handle these two factors which slow down the resolution approach.

5.4.1 Leveraging previously computed results for the lower level sub-problems

Part of the loss of efficiency comes from the fact that, over the course of the hierarchical decomposition algorithm, the solver often solves the same lower level sub-problems repetitively. This repetition is due to the multi-phase nature of CP. Namely, even if two potential deployment plans found in the upper level B&B search tree differ from each other, they may contain the same system layout for a subset of construction phases. More precisely, two potential deployment plans $\mathbf{x}^1 = (\mathbf{x}^1_1, ..., \mathbf{x}^1_\Phi)$ and $\mathbf{x}^2 = (\mathbf{x}^2_1, ..., \mathbf{x}^2_\Phi)$ may be such that $\mathbf{x}^1 \neq \mathbf{x}^2$ and $\mathbf{x}^1_\phi = \mathbf{x}^2_\phi$ for some but not all phases $\phi \in \{1, ..., \Phi\}$. Therefore, for all ϕ such that $\mathbf{x}^1_\phi = \mathbf{x}^2_\phi$ and all days $d \in \mathcal{D}_\phi$, we will have $OP_{\phi,d}(x^1_\phi) = OP_{\phi,d}(x^2_\phi)$ and the computation of the exact net present cost of $\mathbf{x}^1$ and $\mathbf{x}^2$, Z^1_{CP} and Z^2_{CP}, will involve solving a set of identical operation sub-problems. The corresponding loss of computational efficiency is all the most important in our case that these sub-problems are formulated with a scheduling horizon comprising $H > 1$ time periods and are thus MILP problems of a significantly larger size than the ones used in [Yok+15] to formulate the operation lower level sub-problems. This comes from the fact that [Yok+15] do not consider short-term (intra-day) energy storage and thus formulate operation sub-problems involving only a single scheduling period (i.e. a single hour). In contrast, in our case, the existence of energy storage leads to the formulation of inventory balance equations, which couple together the H time steps of a selected day and force us to consider multi-period scheduling sub-problems.

In order to avoid this waste of computation capacity and thus reduce the computational burden of the algorithm, we propose to improve the implementation of the hierarchical decomposition algorithm.

Excluding infeasible system layouts

A first improvement deals with the case in which the system layout proposed by a given potential deployment plan $\mathbf{x}^1 = (\mathbf{x}^1_1, ..., \mathbf{x}^1_\Phi)$ is infeasible at some phase ϕ, i.e. is such that at least one operation sub-problem $OP_{\phi,d}(\mathbf{x}^1_\phi)$ at phase ϕ is infeasible. In this case, we add a no-good cut to exclude all potential deployment plans $\mathbf{x}$ such that $\mathbf{x}_\phi = \mathbf{x}^1_\phi$.

This cut is obtained by using the same partition of the set of all indices as the one used for the no-good cuts (5.27), i.e., the sets $\mathcal{I}^0_\phi = \{(i,\theta) \in I | xbin^1_{\phi,i,\theta} = 0\}$ and $\mathcal{I}^1_\phi = \{(i,\theta) \in I | xbin^1_{\phi,i,\theta} = 1\}$. The no-good cut added to the formulation of SRP to exclude the infeasible system layout $\mathbf{x}^1_\phi$ is formulated as follows:

$$\sum_{(i,\theta)\in\mathcal{I}^0_\phi} xbin_{\phi,i,\theta} + \sum_{(i,\theta)\in\mathcal{I}^1_\phi} (1 - xbin_{\phi,i,\theta}) \geq 1 \tag{5.28}$$

Constraint (5.28) enforces that any potential system layout considered for phase ϕ differs from $\mathbf{x}^1_\phi$ with respect to at least one of the binary variables involved in the binary expansion of $\mathbf{x}_\phi$. Adding Constraint (5.28) to the formulation of SRP thus ensures that any potential deployment plan encountered during the upper level

branch-and-cut search will be considered as a feasible solution of SRP only if at least one of the components of $\mathbf{x}_\phi$ has a value different to the one it has in $\mathbf{x}^1_\phi$.

Storing the optimal value of previously solved operation sub-problems

In order to avoid repetitively solving the same operation sub-problems multiple times over the course of the algorithm, we also propose to record in a hash table the optimal value of any feasible operation sub-problem solved during the algorithm. Thus, let $\mathbf{x}^1$ be a potential deployment plan found during the upper level branch-and-cut search. For each phase ϕ, if the sub-problem $OP_{\phi,d}(\mathbf{x}^1_\phi)$ relative to day (ϕ, d) is feasible, its objective value $Z_{\phi,d}(\mathbf{x}^1_\phi)$ is stored in the hash table with a key $(\mathbf{x}^1_\phi, \phi, d)$. This information can be latter used in the algorithm to save some computational effort. Namely, suppose that $\mathbf{x}^2 = (\mathbf{x}^2_1, ..., \mathbf{x}^2_\Phi)$ is a potential design found after $\mathbf{x}^1$ during the upper level branch-and-cut search. For each phase ϕ and each $d \in D_\phi$, we check whether $(\mathbf{x}^2_\phi, \phi, d)$ is a key of the hash table. If so, i.e. if $\mathbf{x}^2_\phi = \mathbf{x}^1_\phi$, the corresponding relaxed operation cost $f_{\phi,d}(\tilde{\mathbf{y}}_{\phi,d}, \tilde{\mathbf{z}}_{\phi,d})$ can be directly replaced in the computation of Z^2_{CP} by the value $Z_{\phi,d}(\mathbf{x}^1_\phi)$ stored in the hash table so that we avoid a new resolution of $OP_{\phi,d}(\mathbf{x}^1_\phi) = OP_{\phi,d}(\mathbf{x}^2_\phi)$. If $(\mathbf{x}^2_\phi, \phi, d)$ is not a key of the hash table, we solve operation sub-problem $OP_{\phi,d}(\mathbf{x}^2_\phi)$ as an MILP and record its optimal value in the hash table if it is feasible. The searching time in a hash table is short and independent of the table size, it is thus negligible as compared to the time needed to solve an operation sub-problem as an MILP.

5.4.2 Valid inequalities

As explained in Section 5.3, an integer feasible solution of SRP $(\mathbf{x}^\sharp, \tilde{\mathbf{y}}^\sharp, \tilde{\mathbf{z}}^\sharp)$ provides a potential deployment plan which is feasible to the original design constraints and to a relaxed version of the operation constraints. However, $\mathbf{x}^\sharp$ may be infeasible with respect to some of the original operation constraints containing integer operation variables. In order to discuss this point in detail, we need to come back to the initial formulation of CP.

Indeed, most of the feasibility issues encountered when solving the operation sub-problems for a given potential deployment plan found at the upper level come from the impossibility to comply with a specific set of operation constraints: the minimum production constraints. Namely, when the integer operation variable $\mathbf{S}_{\mathbf{t},\mathbf{m}}$ representing the number of active devices of type $\mathbf{m}$ at time step $\mathbf{t}$ is relaxed to a continuous variable, the corresponding continuous energy production variable $\mathbf{P}^{out}_{\mathbf{t},n,\mathbf{m}}$ may take any value in the range $[0, P^{max}_{n,\mathbf{m}}]$ rather than in the range $\{0\} \cup [P^{min}_{n,\mathbf{m}}, P^{max}_{n,\mathbf{m}}]$. The same difficulty arises when the working range of a set of devices is linked to the decision variable $\mathbf{MODE}_{\mathbf{t},n,\mathbf{m}}$ representing the number of devices operating under a given mode n. In other words, a solution of SRP complies with the original maximum production constraints but does not have to respect the minimum production constraints. As a result, the potential deployment plans found when solving SRP at the upper level often involve only high capacity devices which are efficient at producing a large amount of supply commodity but are not able to meet a small but non-zero demand due to the minimum power restrictions.

In this subsection, we thus investigate a set of valid inequalities to tighten the formulation of SRP and reduce the gap between CP and its relaxation SRP.

These valid inequalities focus on the supply commodities that cannot be sold to outside buyers, i.e. to the commodities in $\mathcal{C}^S \setminus \mathcal{C}^{S,Sell}$. The demand for supply commodity $\mathbf{c} = (n, s)$ can be satisfied by conversion and storage devices. Let us

denote by $\mathbf{c}' = (n', s')$ the input commodity of these storage devices. For example, in the trigeneration system in Figure 3.4, the supply commodity $\mathbf{c} = (HEAT, 2)$ has an associated intermediate stored commodity $\mathbf{c}' = (HEAT, 1)$ and the supply commodity $\mathbf{c} = (COLD, 1)$ has an associated intermediate stored commodity $\mathbf{c}' = (ICE, 1)$. We define the set of conversion device types which can produce either supply commodity $\mathbf{c}$ or its corresponding intermediate stored commodity $\mathbf{c}'$ as $\mathcal{M}(\mathbf{c}) = \{(p, l) \in \mathcal{M}^{CONV} | p \in \mathcal{P}^{inflow}_{\mathbf{c}} \cup \mathcal{P}^{inflow}_{\mathbf{c}'}\}$.

We introduce the concept of oversized devices. A device is defined as oversized with respect to a supply commodity $\mathbf{c} = (n, s)$ for a representative day (ϕ, d) if it will not be able to single-handedly meet the demand for this commodity for all time steps $h \in \{0, ..., H-1\}$ of this day due to the restriction on its minimum power production. Namely, an oversized device cannot be used to satisfy the demand for $\mathbf{c}$ either because its minimum hourly production per hour of $\mathbf{c}$ is higher than the demand value of one hour in the day, or because its minimum production per hour of $\mathbf{c}'$ is higher than the total daily demand. We thus define two sets of oversized devices:

- the subset of $\mathcal{M}(\mathbf{c})$ of device types which do not produce $\mathbf{c}$ or whose minimum production per hour of $\mathbf{c}$ is too large to allow them to produce the lowest hourly demand for $\mathbf{c} = (n, s)$ during day (ϕ, d), i.e. $P^{min}_{\mathbf{m},n} > \min_{h=0,..,H-1} Dem_{\phi,d,h,n,s}$. This subset is denoted as $\mathcal{M}^{O,1}_{\phi,d,\mathbf{c}}$.
- the subset of $\mathcal{M}(\mathbf{c})$ of devices types which do not produce $\mathbf{c}'$ or whose minimum production per hour of $\mathbf{c}'$ is too large to allow them to produce the total daily demand of day (ϕ, d), i.e. $P^{min}_{\mathbf{m},n'} > \sum_{h=0}^{H-1} Dem_{\phi,d,h,n,s}$. This subset is denoted as $\mathcal{M}^{O,2}_{\phi,d,\mathbf{c}}$.

We define the set of oversized devices as the intersection of the oversized devices of these two types $\mathcal{M}^{O}_{\phi,d,\mathbf{c}} = \mathcal{M}^{O,1}_{\phi,d,\mathbf{c}} \cap \mathcal{M}^{O,2}_{\phi,d,\mathbf{c}}$. Clearly, a system layout at phase ϕ comprising devices which are all in the set $\mathcal{M}^{O}_{\phi,d,\mathbf{c}}$ will not be able to satisfy the demand for the supply commodity $\mathbf{c}$ for day $d \in \mathcal{D}_\phi$ while simultaneously complying with all the minimum power production constraints. Therefore, any feasible system layout at phase ϕ must involve at least one non oversized device, i.e. one device in $\mathcal{M}(\mathbf{c})$ not belonging to $\mathcal{M}^{O}_{\phi,d,\mathbf{c}}$. Moreover, we note that, when expressing theses constraints, there is no need to consider all days in $\mathcal{D}_\phi$. Namely, thanks to the way extreme days are selected (see Subsection 4.2.2), it is enough to focus, for each phase, on the extreme day displaying the smallest hourly demand and on the one displaying the smallest total daily demand. The constraints relative to the other days will be redundant with the ones relative to these two extreme days.

Let $\mathcal{D}^{low}_{\phi} \subset \mathcal{D}_\phi$ be the set of these two extreme days. We add the following valid inequalities to the formulation of SRP in a pre-treatment step, i.e. before starting the upper level branch-and-cut search.

$$\sum_{\mathbf{m} \in \mathcal{M}(\mathbf{c}) \setminus \mathcal{M}^{O}_{\phi,d}} \sum_{\varphi=1}^{\phi} \mathbf{SD}_{\varphi,\mathbf{m}} \geq 1 \quad \forall d \in \mathcal{D}^{low}_{\phi}, \forall \phi \tag{5.29}$$

By adding Constraint (5.29), we ensure that any system layout found in the upper level problem will contain at least one non-oversized device corresponding to the supply commodity $\mathbf{c}$ to deal with the energy demand with small values.

Moreover, due the definition of the set $\mathcal{M}^{O}_{\phi,d}$, we may encounter system layouts satisfying Constraints (5.29) but containing no devices in the set $\mathbf{m} \in \mathcal{M}(\mathbf{c}) \setminus \mathcal{M}^{O,1}_{\phi,d}$.

In this case, the only way the LMES may satisfy a small daily demand of $\mathbf{c}$ consists in producing $\mathbf{c}'$ during one (or several) hour(s) and storing it in an energy storage device. This implies that, if $\sum_{\mathbf{m}\in\mathcal{M}(\mathbf{c})\setminus\mathcal{M}^{O,1}_{\phi,d}}\sum_{\varphi=1}^{\phi}\mathbf{SD}_{\varphi,\mathbf{m}}=0$, at least one unit of discrete energy storage capacity should be installed. This leads to a second set of valid inequalities:

$$\sum_{\varphi=1}^{\phi}\sum_{\mathbf{m}'\in\mathcal{M}'(\mathbf{c}')}\mathbf{SD}_{\varphi,\mathbf{m}'}\geq 1-\sum_{\mathbf{m}\in\mathcal{M}(\mathbf{c})\setminus\mathcal{M}^{O,1}_{\phi,d}}\sum_{\varphi=1}^{\phi}\mathbf{SD}_{\varphi,\mathbf{m}}\quad\forall d\in\mathcal{D}^{low}_{\phi},\forall\phi \tag{5.30}$$

The set $\mathcal{M}'(\mathbf{c}')$ denote the set of storage devices $\mathbf{m}'=(p,l)$ such that $p\in\mathcal{P}^{STO}\cap\mathcal{P}^{outflow}_{\mathbf{c}'}$, i.e. the set of storage devices able to store commododity $\mathbf{c}'$. Constraints (5.30) state that if a system layout contains only oversized devices of Type 1, it should have at least one unit of discrete energy storage capacity corresponding to the stored commodity $\mathbf{c}'$.

These valid inequalities will contribute in improving the efficiency of the hierarchical decomposition algorithm by a priori excluding some infeasible potential deployment plans, thus saving the computational effort needed to determine that they are infeasible.

5.4.3 Summary of the improved hierarchical decomposition algorithm

Before presenting our numerical results, we provide a summarized description of the proposed extended hierarchical decomposition algorithm: see Algorithm 2. Moreover, Figure 5.2 provides a graphical description of the improved hierarchical decomposition algorithm as applied to our multi-phase deployment problem.

5.5 Computational experiments

We now focus on assessing the performance of the hierarchical decomposition algorithm, both in its initial and extended variants, at solving the problem of optimally designing an LMES over a multi-phase investment plan. This assessment is carried out by using the set of instances based on our three case studies presented in Subsection 4.4.1.

Each instance is solved by the original hierarchical decomposition algorithm presented in [Yok+15] and by the extended hierarchical decomposition discussed above. The experimental setup was the same as the one described at the beginning of Subsection 4.4.2. In both variants of the hierarchical decomposition algorithm, the upper level master problem was solved by CPLEX 12.8 solver. Incumbent Callbacks were used to control the flow of information between the upper and lower problems and Usercut Callbacks were used to set the upper cutoff value to the current value of Z^{best}_{CP}. Due to the presence of a Usercut Callback, which is classified as a "Control Callback", parallel MILP solving is turned off and CPLEX 12.8 solver uses a single thread to solve the upper level master problem. We imposed a time limit of 7200s.

Similar to what was done in Subsection 4.4.2, we provide, for each instance and each considered solution approach, the integrality gap Gap_{LP}, the computation time and the residual gap Gap_{MIP}. Moreover, we also report:

Algorithm 2: Extended Hierarchical Decomposition

Data: Parameters of commodities and technologies, construction phases and discount rate

Result: The optimal system deployment plan and the minimum total cost

For each supply commodity in $\mathcal{C}^S \setminus \mathcal{C}^{S,SelI}$, add the valid inequalities (5.29)-(5.30) to the formulation of SRP;
Set Z_{CP}^{best} to positive infinity;
Create the root node of the B&C search tree ;
while *active nodes exist in B&C search tree* **do**
  Branch from an active node in the tree ;
  if *an integer feasible solution* $(\mathbf{x}^\sharp, \tilde{\mathbf{y}}^\sharp, \tilde{\mathbf{z}}^\sharp)$ *with an objective value* $Z_{SRP}^\sharp \leq Z_{CP}^{best}$ *is found* **then**
    Set *IsNewIncumbent* to *True*;
    for *each phase* $\phi \in \{1, ..., \Phi\}$ *and each day* $d \in \mathcal{D}_\phi$ **do**
      if $(\mathbf{x}_\phi^\sharp, \phi, d)$ *is in the hash table* **then**
        Retrieve the corresponding value of $Z_{\phi,d}(\mathbf{x}_\phi^\sharp)$;
        Set $Z_{CP}^\sharp$ to $Z_{CP}^\sharp + Z_{\phi,d}(\mathbf{x}_\phi^\sharp) - f_{\phi,d}(\tilde{\mathbf{y}}_{\phi,d}^\sharp, \tilde{\mathbf{z}}_{\phi,d}^\sharp)$;
      else
        Solve Problem $OP_{\phi,d}(\mathbf{x}_\phi^\sharp)$ as an MILP;
        if $OP_{\phi,d}(\mathbf{x}_\phi^\sharp)$ *is infeasible* **then**
          Add the no-good cut (5.28) corresponding to $\mathbf{x}_\phi^\sharp$ to SRP;
          Set *IsNewIncumbent* to *False*;
          break ; `/* To break the for loop in Line 8 */`
        else
          Store the value of $Z_{\phi,d}(\mathbf{x}_\phi^\sharp)$ under the new key $(\mathbf{x}_\phi^\sharp, \phi, d)$ in the hash table;
          Set $Z_{CP}^\sharp$ to $Z_{CP}^\sharp + Z_{\phi,d}(\mathbf{x}_\phi^\sharp) - f_{\phi,d}(\tilde{\mathbf{y}}_{\phi,d}^\sharp, \tilde{\mathbf{z}}_{\phi,d}^\sharp)$;
        end
      end
      if $Z_{CP}^\sharp > Z_{CP}^{best}$ **then**
        Set *IsNewIncumbent* to *False*;
        Reject $(\mathbf{x}^\sharp, \tilde{\mathbf{y}}^\sharp, \tilde{\mathbf{z}}^\sharp)$;
        Add the no-good cut (5.27);
        break; `/* To break the for loop in Line 8 */`
      end
    end
    if *IsNewIncumbent is True* **then**
      Set $(\mathbf{x}^\sharp, \mathbf{y}^\sharp, \mathbf{z}^\sharp)$ to $(\mathbf{x}^{best}, \mathbf{y}^{best}, \mathbf{z}^{best})$ and $Z_{CP}^\sharp$ to Z_{CP}^{best} ;
      Add cut $f_0(\mathbf{x}) + \sum_{(\phi,d)\in\mathcal{D}} f_{\phi,d}(\mathbf{y}_{\phi,d}, \mathbf{z}_{\phi,d}) \leq Z_{CP}^{best}$ to SRP ;
      Reject $(\mathbf{x}^\sharp, \tilde{\mathbf{y}}^\sharp, \tilde{\mathbf{z}}^\sharp)$;
      Add the no-good cut (5.27);
    end
  end
end
return $(\mathbf{x}^{best}, \mathbf{y}^{best}, \mathbf{z}^{best})$ and Z_{CP}^{best};

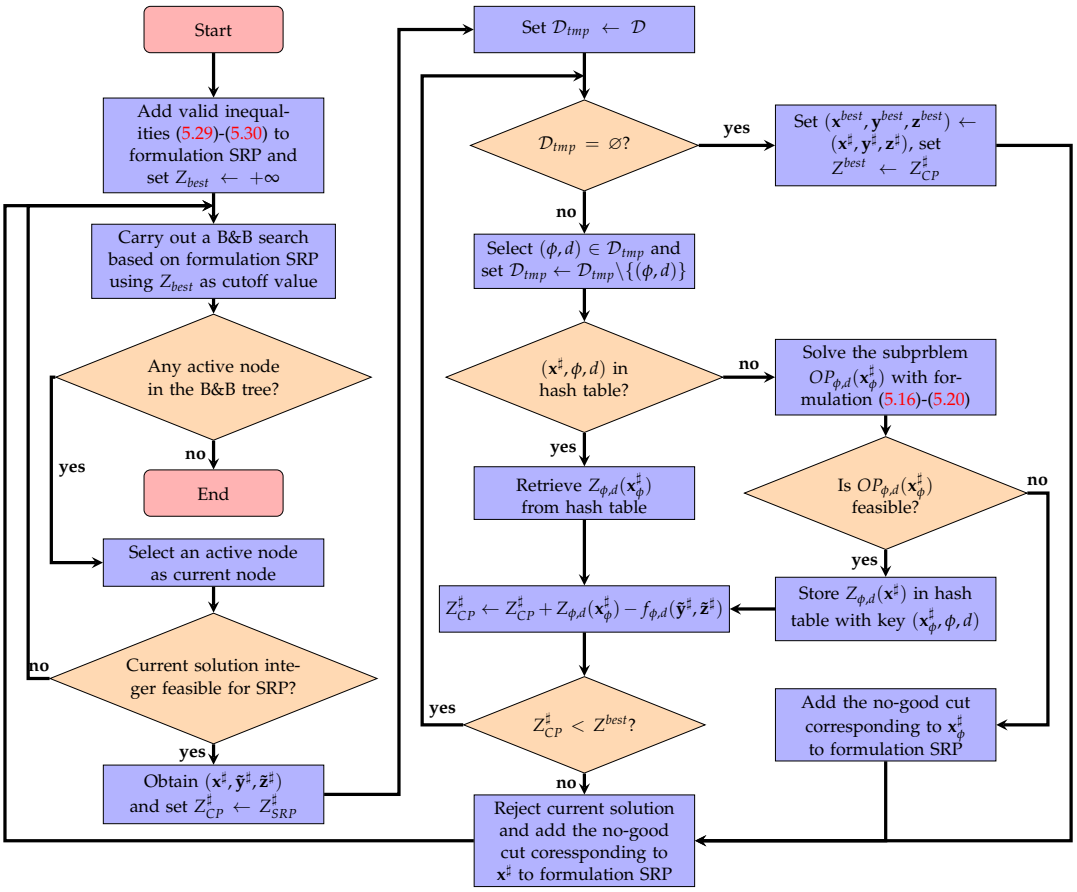


FIGURE 5.2: Flow chart of hierarchical decomposition algorithm

- $\#IFS_{SRP}$: the number of integer feasible solutions of SRP explored during the upper level branch-and-cut search before the algorithm converges or the time limit is reached,
- $\#OP$: the number of lower-level operation sub-problems solved over the course of the algorithm.

Results from Tables 5.1-5.3 first show that, in general, the initial hierarchical decomposition algorithm does not seem to be able to outperform the direct resolution by CPLEX 12.8. Namely, over the 19 considered instances, the average residual gap (Gap_{MIP}) is 1.59% when using this algorithm as compared to 0.40% when directly solving the whole problem as an MILP.

In contrast, the extended hierarchical decomposition algorithm provides a guaranteed optimal solution of the MILP problem formulated in Chapter 4 for all instances of City A project and for 4 out of 5 instances for City B project. Note that a near-optimal solution with a residual gap of 0.07% is obtained for the last instance related to City B project. Furthermore, the average computation time over the 12 instances related to City A or City B projects is significantly reduced from more than 7200s when using a direct resolution by CPLEX 12.8 or the initial hierarchical algorithm to 2949s when using the extended hierarchical decomposition algorithm.

This improved computational efficiency may be explained by three main reasons. First, the addition of valid inequalities (5.29)-(5.30) to SRP leads to a reduction of the

integrality gap. Namely, over the 12 instances, the average value of Gap_{LP} is reduced from 3.32% with the initial formulation of SRP to 2.56% with the formulation of SRP strengthened by valid inequalities (5.29)-(5.30). Even if this gap reduction is relatively small, it is obtained by adding a very limited number of constraints (4Φ) to the formulation and thus does not negatively impact the computation time needed to solve the linear relaxation of SRP at each node of the upper level B&C search tree. Moreover, adding these valid inequalities to the formulation enables us to a priori exclude from the upper level search space deployment plans which are feasible for SRP but not for CP and thus to save the computation time needed to determine that they are infeasible. Second, over the 12 considered instances, the average value of $\#\text{IFS}_{SRP}$, the number of integer feasible solutions of SRP explored during the upper level B&C search, is significantly reduced from 2467 with the initial hierarchical decomposition algorithm to 962 with the extended hierarchical decomposition algorithm. This shows the efficiency of the no-good cuts (5.28) at preventing the upper level B&C algorithm from uselessly considering multiple times a system layout which is known to be infeasible for a given investment phase. Finally, thanks to the combined used of the no-good cuts (5.28) and the recording of previously computed results in a hash table, the average number of lower-level operation sub-problems solved over the course of the hierarchical decomposition algorithm, $\#OP$, is drastically reduced from 31709 with the initial variant of the algorithm to 894 with the extended variant. Even if each of these sub-problems is a small MILP which can be solved without any numerical difficulty, dividing by a factor of 35 the number of sub-problems solved over the course of the algorithm positively impacts the overall computation time.

However, the extended hierarchical decomposition algorithm seems to have a poor performance on the instances based on the trigeneration system located in City C. Namely, even if the integrality gap was significantly reduced by adding valid inequalities (5.29)-(5.30), out-of-memory issues were encountered while solving all the instances. Moreover, the average residual gap obtained with the extended hierarchical decomposition algorithm ($\text{Gap}_{MIP} = 0.53\%$) is larger than the one obtained with the direct resolution approach ($\text{Gap}_{MIP} = 0.37\%$). This difficulty may come from the large number of design variables. Namely, the total number of design variables, excluding the binary expansion variables, is 28 (resp. 25) for the DCS project of City A (resp. City B) , while there is a total of 36 design variables for the trigeneration project. This larger number of variables leads to a larger feasible space for SRP to be explored by the B&C algorithm and thus to a larger computational effort. A second reason might be the fact that the lower bound provided by Z_{SRP} is of lower quality for this set of instances than for the City A and City B instances, which forces the B&C algorithm to explore more nodes before converging.

5.6 Conclusion

In this chapter, we first highlighted the bi-level structure of the MILP problem formulated in Chapter 4. This special structure allows to split the original large-scale MILP problem into a set of small-size MILP problems once the deployment plan of the system is determined.

We then applied the previously published hierarchical decomposition algorithm to the large-scale MILP problem formulated in Chapter 4. Note that the hierarchical decomposition algorithm requires that the upper-level variables are all discrete and that the original MILP model introduced in Chapter 4 meets this requirement.

	$\lvert\mathcal{D}_\phi\rvert$	6	14	22	30	38	58	78
MILP	#Var	12555	29259	45963	61971	79371	121131	162984
size	#IntVar	3909	9093	14277	19245	24645	37605	50565
	#Cons	21611	50411	79211	106811	136811	208811	280839
Direct	Gap_{LP}	1.57%	3.26%	3.41%	1.77%	2.17%	2.06%	2.34%
resolution	Time	7200s	7200s	7200s	7200s	7200s	7200s	7200s
	Gap_{MIP}	0.18%	0.04%	0.13%	0.39%	0.44%	0.73%	2.83%
Initial	Gap_{LP}	1.57%	3.26%	3.41%	1.77%	2.17%	2.06%	2.34%
Hier. Dec.	$\#\text{IFS}_{SRP}$	9009	1469	3017	2764	1180	1187	488
	$\#\text{OP}_{solved}$	28150	15295	15595	31555	20335	31848	29561
	Time	7200s	7200s	7200s	7200s	7200s	7200s	7200s
	Gap_{MIP}	0.10%	0.68%	1.46%	0.50%	0.88%	0.74%	1.21%
Extended	Gap_{LP}	1.56%	1.45%	1.53%	1.56%	1.56%	1.56%	1.56%
Hier. Dec.	$\#\text{IFS}_{SRP}$	892	1980	1083	874	919	908	1171
	$\#\text{OP}_{solved}$	172	581	779	515	1254	1654	2058
	Time	123s	814s	1043s	977s	990s	1984s	6822s
	Gap_{MIP}	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%

TABLE 5.1: City A project: numerical comparison of the solution methods

	$\lvert\mathcal{D}_\phi\rvert$	6	8	10	12	14
MILP	#Var	13820	18380	22940	27500	32060
size	#IntVar	3625	4825	6025	7225	8425
	#Cons	23084	30764	38444	46124	53804
Direct	Gap_{LP}	5.32%	4.26%	4.58%	4.77%	4.38%
resolution	Time	7200s	7200s	7200s	2589s*	4411s*
	Gap_{MIP}	0.01%	0.02%	0.02%	0.04%	0.06%
Initial	Gap_{LP}	5.32%	4.26%	4.58%	4.77%	4.38%
Hier. Dec.	$\#\text{IFS}_{SRP}$	4087	1603	1303	2348	1688
	$\#\text{OP}_{solved}$	45406	35193	31752	55160	34656
	Time	5162s	7200s	7200s	7200s	7200s
	Gap_{MIP}	0.00%	0.92%	0.86%	1.60%	0.66%
Extended	Gap_{LP}	4.13%	3.37%	4.10%	4.24%	4.04%
Hier. Dec.	$\#\text{IFS}_{SRP}$	1764	1432	1441	394	1660
	$\#\text{OP}_{solved}$	505	794	618	520	1287
	Time	954s	2889s	4580s	7200s	4062s
	Gap_{MIP}	0.00%	0.00%	0.00%	0.07%	0.00%

"*" means that the computer ran out of memory before reaching the time limit.

TABLE 5.2: City B project: numerical comparison of the solution methods

	$\lvert\mathcal{D}_\phi\rvert$	12	16	20	24	28	32	36
MILP	#Var	76080	101424	123600	148944	175872	201216	226560
size	#IntVar	16164	21540	26244	31620	37332	42708	48084
	#Cons	86420	115220	140420	169220	199820	228620	257420
Direct	Gap_{LP}	5.11%	5.17%	5.20%	5.21%	5.15%	5.14%	5.13%
resolution	Time	5934s*	6302s*	7200s	5259s*	6155s*	5968s*	5327s*
	Gap_{MIP}	0.06%	0.26%	0.23%	0.35%	0.32%	0.16%	1.21%
Initial	Gap_{LP}	5.11%	5.17%	5.20%	5.21%	5.15%	5.14%	5.13%
Hier. Dec.	$\#IFS_{SRP}$	4065	3019	2470	1888	1714	1512	1145
	$\#OP_{solved}$	35631	21560	33267	33418	37872	42548	47503
	Time	5286s*	3608s*	5405s*	5461s*	5938s*	6587s*	7200
	Gap_{MIP}	2.69%	2.21%	2.46%	2.41%	2.28%	2.88%	5.70%
Extended	Gap_{LP}	3.41%	3.65%	3.68%	3.66%	3.6%	3.58%	3.56%
Hier. Dec.	$\#IFS_{SRP}$	3646	2770	2271	1904	1583	1352	1243
	$\#OP_{solved}$	1501	1341	3177	1166	2294	2056	5189
	Time	2609s*	4127s*	2929s*	3244s*	3863s*	5060s*	3713s*
	Gap_{MIP}	0.50%	0.38%	0.71%	0.54%	0.53%	0.36%	0.72%

"*" means that the computer ran out of memory before reaching the time limit.

TABLE 5.3: Trigeneration project: numerical comparison of the solution methods

The hierarchical decomposition algorithm takes as upper-level master problem the semi-relaxed problem (SRP) which is obtained by relaxing all lower-level discrete variables of the original complete problem (CP). The resolution of the upper-level problem is carried out by the built-in B&C algorithm of a commercial mathematical solver. This B&C algorithm is however customized through the use of callbacks: at each node where an integer feasible solution of SRP, i.e. a potential deployment plan, is found, we solve a series of operation sub-problems (OP) to find out the real objective value corresponding to the current deployment plan. All integer solutions at the upper-level are rejected to guarantee the optimal convergence of the algorithm.

We also proposed two ways to further improve the computational efficiency of the hierarchical decomposition method. The first improvement is related to the multi-phase nature of our design problem. In order to avoid repetitively solving several times the same sub-problems, we store in a hash table the optimal value of all feasible solved sub-problems and retrieve this value to reuse it when needed over the course of the algorithm. In case a sub-problem is found to be infeasible, we also add a no-good cut to exclude the current system layout for the corresponding investment phase from the feasible space of SRP. The second improvement consists in adding valid inequalities into the formulation of SRP. Namely, due to the relaxation of discrete lower-level variables, the SRP problem does not consider the minimum load rate of devices and tends to favor the installation of large-capacity devices. The proposed valid inequalities aim at preventing the SRP from choosing only oversized devices and thus contribute in reducing the number of infeasible deployment plans explored by the upper-level B&C algorithm.

Both the initial and extended hierarchical decomposition algorithms are tested on the instances based on the three case studies introduced in Chapter 4. One one hand, the numerical results show that the initial hierarchical decomposition method is on average less efficient than the direct resolution of CP by CPLEX 12.8 solver. On the other hand, the extended hierarchical decomposition algorithm significantly

outperforms the two other methods on the instances related to the two DCS projects, providing optimal (or near optimal) solutions for all instances within the time limit. However, out-of-memory issues were still encountered for the instances related to the trigeneration project, even the ones involving a small number of representative days.

This motivated us to explore another resolution strategy based on a generalized Benders' decomposition algorithm taking advantage of a specific property of the sub-problems of the MILP problem. This solution approach will be presented in Chapter 6.

Chapter 6

Generalized Benders' decomposition algorithm

6.1 Introduction

The numerical results presented in Chapter 5 showed that the extended hierarchical decomposition algorithm is more effective at solving the LMES optimal design problem than an off-the-shelf MILP solver for the instances related to the DCS projects. However, numerical difficulties were still encountered while trying to solve the instances related to the trigeneration system. We thus investigate in this chapter another solution approach which is also based on a decomposition strategy. This approach can be seen as a generalized Benders' decomposition algorithm, more precisely as an extension of the conventional Benders' decomposition algorithm to a case where the second-stage sub-problems involve integer variables. In particular, we propose a new type of optimality and feasibility cuts exploiting the special structure of the constraints linking the design and operation variables in our problem. Thanks to these new cuts, we are able to develop a generalized Benders' decomposition algorithm converging in a finite number of iterations to an optimal solution of the original problem CP.

In this chapter, we first present in Section 6.2 the overall framework used to develop a generalized Benders' decomposition algorithm for our problem. The optimal and finite convergence of this algorithm relies on the use of strong dual bounding functions. We explain in Section 6.3 how such functions can be obtained for our problem by exploiting the special structure of the constraints coupling the design and operation variables. The resulting master problem contains a set of non-linear constraints. Section 6.4 thus investigates how this master problem may be reformulated as an MILP thanks to the introduction of a set of additional binary variables. Section 6.5 summarizes the outline of the proposed algorithm. Finally, Section 6.4 reports the results of the computational experiments carried out to assess the numerical performance of the algorithm. It also presents the outcome of a post-optimization simulation study carried out to evaluate the quality of the deployment plans obtained for the two DCS projects.

6.2 A generalized Benders' decomposition framework

Bolusani and Ralphs [BR21] recently described a framework to extend the well-known decomposition framework introduced by Benders [Ben62] to a more general class of optimization problems. In particular, they highlighted the fact that Benders' decomposition may be used even if the second-stage sub-problems do not display

the strong duality properties holding for linear and convex programs. In what follows, we discuss how this generalized Benders' decomposition framework may be applied to our LMES optimal design problem.

To this aim, we start by recalling the compact reformulation of the original problem, denoted by CP in Chapter 5.

$$Z^*_{CP} = \min f_0(\mathbf{x}) + \sum_{\phi=1}^{\Phi} \sum_{d \in \mathcal{D}_\phi} f_{\phi,d}(\mathbf{y}_{\phi,d}, \mathbf{z}_{\phi,d}) \tag{6.1}$$

$$h(\mathbf{x}) \leq 0 \tag{6.2}$$

$$g_{\phi,d}(\mathbf{x}_\phi, \mathbf{y}_{\phi,d}, \mathbf{z}_{\phi,d}) \leq 0 \qquad \forall (\phi, d) \in \mathcal{D} \tag{6.3}$$

$$l_{\phi,d}(\mathbf{y}_{\phi,d}, \mathbf{z}_{\phi,d}) \leq 0 \qquad \forall (\phi, d) \in \mathcal{D} \tag{6.4}$$

$$\mathbf{x} \in \mathbb{Z}^\nu \tag{6.5}$$

$$\mathbf{y}_{\phi,d} \in \mathbb{R}^\mu \qquad \forall (\phi, d) \in \mathcal{D} \tag{6.6}$$

$$\mathbf{z}_{\phi,d} \in \mathbb{Z}^\lambda \qquad \forall (\phi, d) \in \mathcal{D} \tag{6.7}$$

As mentioned in [BR21], a Benders' decomposition approach can be seen as a reformulation technique in which the original optimization problem is reformulated using only a subset of the original variables. This subset of variables are referred to as the first-stage variables whereas the variables not included in the reformulation are called the second-stage variables. Thus, a Benders' decomposition method essentially carries out a projection of the original optimization problem onto the subspace corresponding to the first-stage variables.

In order to explain how this reformulation/projection technique can be applied on CP, we first introduce some additional notation. Let $\mathcal{X} = \{\mathbf{x} \in \mathbb{Z}^\nu | h(\mathbf{x}) \leq 0\}$ denote the set of deployment plans feasible with respect to the design constraints (6.2) and $\mathcal{X}_\phi = \text{proj}_{\mathbf{x}_\phi}(\mathcal{X})$ the projection of $\mathcal{X}$ onto the space of the design variables relative to phase ϕ. Let $\mathcal{S}_\phi = \{(\mathbf{y}_\phi, \mathbf{z}_\phi) \in \mathbb{R}^{\mu|\mathcal{D}_\phi|} \times \mathbb{Z}^{\lambda|\mathcal{D}_\phi|} | l_{\phi,d}(\mathbf{y}_{\phi,d}, \mathbf{z}_{\phi,d}) \leq 0, \forall d \in \mathcal{D}_\phi\}$ denote the set of operation schedules relative to days $d \in \mathcal{D}_\phi$ feasible with respect to the operation constraints (6.4). Finally, $\mathcal{F}_\phi = \{(\mathbf{x}_\phi, \mathbf{y}_\phi, \mathbf{z}_\phi) \in \mathcal{X}_\phi \times \mathcal{S}_\phi | g_{\phi,d}(\mathbf{x}_\phi, \mathbf{y}_{\phi,d}, \mathbf{z}_{\phi,d}) \leq 0, , \forall d \in \mathcal{D}_\phi\}$ represents the set of system layouts and operation schedules relative to phase ϕ complying with the design, coupling and operation constraints (6.2)-(6.4).

Using this notation, CP can be equivalently formulated as:

$$Z^*_{CP} = \min_{\mathbf{x} \in \mathcal{X}} \left\{ f_0(\mathbf{x}) + \sum_{\phi=1}^{\Phi} \min_{(\mathbf{y}_\phi, \mathbf{z}_\phi) \in \mathcal{S}_\phi} \left\{ \sum_{(\phi,d) \in \mathcal{D}_\phi} f_{\phi,d}(\mathbf{y}_{\phi,d}, \mathbf{z}_{\phi,d}) | (\mathbf{x}_\phi, \mathbf{y}_\phi, \mathbf{z}_\phi) \in \mathcal{F}_\phi \right\} \right\} \tag{6.8}$$

We now replace each inner minimization problem by a function. This allows us to reformulate CP using only the first-stage design variable $\mathbf{x}$:

$$Z^*_{CP} = \min_{x \in \mathcal{X}} \left\{ f_0(\mathbf{x}) + \sum_{\phi=1}^{\Phi} Z_\phi(\mathbf{x}_\phi) \right\} \tag{6.9}$$

Function $Z_\phi : \mathbb{R}^{\nu_\phi} \to \mathbb{R} \cup \{+\infty\}$ is defined as:

$$Z_\phi(\mathbf{x}_\phi) = \begin{cases} \min\limits_{(\mathbf{y}_\phi, \mathbf{z}_\phi) \in \mathcal{S}_\phi} \left\{ \sum\limits_{d \in \mathcal{D}_\phi} f_{\phi,d}(\mathbf{y}_{\phi,d}, \mathbf{z}_{\phi,d}) | (\mathbf{x}_\phi, \mathbf{y}_\phi, \mathbf{z}_\phi) \in \mathcal{F}_\phi \right\} & \text{if } \mathbf{x}_\phi \in \text{proj}_{x_\phi}(\mathcal{F}_\phi) \\ +\infty & \text{otherwise} \end{cases}$$

where $\text{proj}_{\mathbf{x}_\phi}(\mathcal{F}_\phi) \subset \mathcal{X}_\phi$ is the projection of $\mathcal{F}_\phi$ onto the space of the first-stage variables $\mathbf{x}_\phi$. $\text{proj}_{x_\phi}(\mathcal{F}_\phi)$ can be intuitively seen as the set of system layouts for phase ϕ, $\mathbf{x}_\phi$, such that there is at least one feasible operation schedule for each day $d \in \mathcal{D}_\phi$.

Z_ϕ is thus a function that returns the sum of the cost of the best operation schedule that can be found for each day $d \in \mathcal{D}_\phi$ under the system layout $\mathbf{x}_\phi$ if one exists. In fact, $Z_\phi(\mathbf{x}_\phi)$ is the sum, over all days $d \in \mathcal{D}_\phi$, of the optimal value of the operation sub-problems $OP_{\phi,d}(\mathbf{x}_\phi)$ introduced in Section 5.2. Moreover, Z_ϕ prevents any first-stage design solution $\mathbf{x}_\phi$ that is not feasible with respect to one or several days $d \in \mathcal{D}_\phi$, i.e. which is not in $\text{proj}_{x_\phi}(\mathcal{F}_\phi)$, from being considered.

In principle, solving Problem (6.9) would provide us with an optimal solution of CP. However, this would require a closed-form expression of each function Z_ϕ. As noted in [BR21], even when such a closed form exists in theory, its description is typically of exponential size and computationally impractical.

The basic idea of a Benders' decomposition approach consists in overcoming this difficulty by replacing each function Z_ϕ by a dual bounding function $\underline{Z}_\phi$ to obtain a relaxation of Problem (6.9) called the master problem.

A dual bounding function $\underline{Z}_\phi : \mathbb{R}^{\nu_\phi} \to \mathbb{R} \cup \{+\infty\}$ is defined as a function such that $\underline{Z}_\phi(\mathbf{x}_\phi) \leq Z_\phi(\mathbf{x}_\phi)$ for all $\mathbf{x}_\phi \in \mathbb{R}^{\nu_\phi}$. This dual function is called strong at $\hat{\mathbf{x}}_\phi \in \mathbb{R}^{\nu_\phi}$ if $\underline{Z}_\phi(\hat{\mathbf{x}}_\phi) = Z_\phi(\hat{\mathbf{x}}_\phi)$.

The master problem can thus be written as:

$$\min_{x \in \mathcal{X}} \left\{ f_0(\mathbf{x}) + \sum_{\phi=1}^{\Phi} \underline{Z}_\phi(\mathbf{x}_\phi) \right\} \tag{6.10}$$

Problem (6.10) is a relaxation of Problem (6.9) (and consequently a relaxation of the initial problem CP) and thus provides a lower bound of Z^*_{CP}. The aim of a generalized Benders' decomposition algorithm will be to iteratively improve the formulation of the master problem by progressively strengthening the dual bounding function $\underline{Z}_\phi$ used for each phase ϕ. Basically, at each iteration k, we solve the master problem using the current expression of the dual bounding functions, yielding a lower bound LB^k of Z^*_{CP} and a design solution denoted by $\mathbf{x}^k$. We then evaluate $Z_\phi(\mathbf{x}^k_\phi)$, for each phase ϕ, by solving a series of operation sub-problems $OP_{\phi,d}(\mathbf{x}^k_\phi)$, obtain an upper bound $UB^k = f_0(\mathbf{x}^k) + \sum_{\phi=1}^{\Phi} Z_\phi(\mathbf{x}^k_\phi)$ of Z^*_{CP} and use the information obtained during this process to update each dual bounding function $\underline{Z}_\phi$ so as to strengthen it. The algorithm terminates when $LB^k = UB^k$.

Hooker and Ottosson [HO03] showed that the finite convergence towards an optimal solution of such a generalized Benders' decomposition algorithm is guaranteed as long as two conditions are respected. The first one is that the approximation of the dual bounding function used at iteration k for each phase ϕ is strong at each of the previously considered iterates, i.e. is such that $\underline{Z}_\phi(\mathbf{x}^j_\phi) = Z_\phi(\mathbf{x}^j_\phi)$ for all iterations j in 1...k. The second one is that each set $\text{proj}_{x_\phi}(\mathcal{F}_\phi)$ is finite, i.e. that $|\text{proj}_{x_\phi}(\mathcal{F}_\phi)| < \infty$.

The first condition can be more easily satisfied by defining $\underline{Z}_\phi$ as the maximum of the lower bounding functions obtained at each previous iteration, i.e. by defining $\underline{Z}_\phi$ at iteration k as:

$$\underline{Z}_\phi(\mathbf{x}_\phi) = \max_{1 \leq j \leq k} \underline{Z}^j_\phi(\mathbf{x}_\phi) \tag{6.11}$$

where $\underline{Z}^j_\phi$ is the dual bounding function obtained at iteration $j \leq k$. In this case, if $\underline{Z}^j_\phi$ is strong at $\mathbf{x}^j_\phi$, the dual bounding function used at a latter iteration $k \geq j$, $\underline{Z}_\phi$, will also be strong at $\mathbf{x}^j_\phi$.

The master problem (6.10) can be reformulated using auxiliary variables η_ϕ to eliminate the maximum operator involved in (6.11). We thus obtain at iteration k the following master problem denoted by MP^k.

$$\min Z_{MP^k} = f_0(\mathbf{x}) + \sum_{\phi=1}^{\Phi} \eta_\phi \tag{6.12}$$

$$x \in \mathcal{X} \tag{6.13}$$

$$\eta_\phi \geq \underline{Z}^j_\phi(\mathbf{x}_\phi) \qquad \forall \phi \in \{1, \cdots, \Phi\},\ \forall j \in \{1, \cdots, k\} \tag{6.14}$$

$$F^j_\phi(\mathbf{x}_\phi) \leq 0 \qquad \forall \phi \in \{1, \cdots, \Phi\},\ \forall j \in \{1, \cdots, k\} \tag{6.15}$$

$$\eta_\phi \in \mathbb{R} \qquad \forall \phi \in \{1, \cdots, \Phi\} \tag{6.16}$$

In this formulation, Constraints (6.14) can be understood as Benders' optimality cuts. Moreover, in order to avoid handling infinite values of the lower bounding function $\underline{Z}_\phi$, we may add a series of constraints to explicitly exclude first-stage variables $\mathbf{x}$ such that $\mathbf{x}_\phi \notin \text{proj}_{x_\phi}(\mathcal{F}_{\phi,d})$ for some phase ϕ. This may be done by adding some constraints called Benders' feasibility cuts. This is the purpose of Constraints (6.15) in which $F^j_\phi(\mathbf{x}_\phi)$ denotes a general mathematical expression involving elements of $\mathbf{x}_\phi$.

Moreover, in our problem, all the first-stage decision variables are bounded integer variables: see the design constraints (4.1)-(4.2) introduced in Chapter 4. $\mathcal{X}_\phi$ and as a consequence $\text{proj}_{x_\phi}(\mathcal{F}_\phi) \subset \mathcal{X}_\phi$ are therefore finite sets for all phases ϕ.

Hence, the main remaining challenge towards ensuring that the generalized Benders' decomposition will finitely converge towards an optimal solution is to build, for each phase ϕ, a strong lower bounding function at each iteration k, i.e. to build a function $\underline{Z}^k_\phi$ such that $\underline{Z}^k_\phi(\mathbf{x}_\phi) \leq Z_\phi(\mathbf{x}_\phi), \forall \mathbf{x}_\phi \in \mathbb{R}^{\nu_\phi}$ and $\underline{Z}^k_\phi(\mathbf{x}^k_\phi) = Z_\phi(\mathbf{x}^k_\phi)$. This is the purpose of the next section.

6.3 Strong dual bounding functions

We now investigate the development of strong dual bounding functions for our problem. This development exploits the specific structure of the coupling constraints (6.3) in CP.

We start by defining a partial order between the integer vectors describing the system layout at phase ϕ. Thus, a system layout described by vector $\mathbf{x}^\sharp_\phi$ is defined to be smaller than the system layout described by vector $\mathbf{x}^\bullet_\phi$ if $\mathbf{x}^\sharp_\phi$ is element-wise smaller than $\mathbf{x}^\bullet_\phi$, i.e. if we have $x^\sharp_{\phi,i} \leq x^\bullet_{\phi,i}$ for each $i = 1 \ldots \nu_\phi$. In this case, we note

$\mathbf{x}_\phi^\sharp \preceq \mathbf{x}_\phi^\bullet$. Two system layouts are said to be comparable if one is smaller than the other.

Moreover, we note that the constraints (6.3) coupling the design and the operation variables in CP have a specific structure. Namely, they either take the form $y_{\phi,d,h,j} \leq x_{\phi,i}$ (see e.g. Constraints (4.9), (4.15), (4.19), (4.22), (4.25) for the energy conversion technologies) or the form $z_{\phi,d,h,j} \leq c_i x_{\phi,i}$ with $c_i > 0$ (see e.g. Constraints (4.5) for the resource commodities in $\mathcal{C}^{R,Co}$ and Constraints (4.29) for the energy storage technologies.)

Let us now consider two operation sub-problems $OP_{\phi,d}(\mathbf{x}_\phi^\sharp)$ and $OP_{\phi,d}(\mathbf{x}_\phi^\bullet)$ and their feasible spaces denoted by $\mathcal{V}_{\phi,d}^\sharp$ and $\mathcal{V}_{\phi,d}^\bullet$. Thanks to the key observation on the coupling constraints mentioned above, we note that if $x_{\phi,i}^\sharp \preceq x_{\phi,i}^\bullet$, all the coupling constraints involved in $OP_{\phi,d}(\mathbf{x}_\phi^\sharp)$ have a right-hand side smaller than the one of the coupling constraints involved in $OP_{\phi,d}(\mathbf{x}_\phi^\bullet)$ and are thus stricter so that $\mathcal{V}_{\phi,d}^\sharp \subseteq \mathcal{V}_{\phi,d}^\bullet$. This inclusion relationship between their feasible regions implies that if both operation scheduling sub-problems $OP_{\phi,d}(\mathbf{x}_\phi^\sharp)$ and $OP_{\phi,d}(\mathbf{x}_\phi^\bullet)$ are feasible with optimal objective value $Z_{\phi,d}^\sharp$ and $Z_{\phi,d}^\bullet$, $Z_{\phi,d}^\sharp \geq Z_{\phi,d}^\bullet$. Moreover, if $OP_{\phi,d}(\mathbf{x}_\phi^\bullet)$ is infeasible, $OP_{\phi,d}(\mathbf{x}_\phi^\sharp)$ is also infeasible. Note that this mathematical reasoning can be intuitively related to the fact that a system layout $\mathbf{x}_\phi^\sharp$ which has more components of each type (i.e. which has more energy conversion devices of each type, a larger storage capacity, a larger contracted power....) than the system layout $\mathbf{x}_\phi^\bullet$ can be operated using the same operation schedules as the ones used with the system layout $\mathbf{x}_\phi^\bullet$ (and maybe with less expensive operation schedules). This implies that the operation cost of $\mathbf{x}_\phi^\sharp$ cannot be larger than the one of $\mathbf{x}_\phi^\bullet$.

This property can be illustrated by an example based on the case study related to the DCS project located in City B. We consider the operation cost for Day 143 of Phase 4 of this project. The number of single-mode chillers of type $(SMEC, 1)$ is fixed to 0, the ice storage capacity to 171MWh and the contracted maximum power to 15MW. Figure 6.1 shows how the operation cost for this day, $Z_{4,143}$, varies as a function of the design vector $\mathbf{x}_4 = (x_{4,1}, x_{4,2})$ where $x_{4,1}$ is the number of chillers of type $(SMEC, 2)$ and $x_{4,2}$ the number of chillers of type $(DMEC, 1)$ present in the DCS at the beginning of Phase 4. The z-axis indicates the value of the operation cost $Z_{4,143}(\mathbf{x}_4)$: a value of $Z_{4,143}(\mathbf{x}_4)$ above 550000 corresponds to an 'infinite value' indicating that the system layout $\mathbf{x}_4$ is infeasible for day (4,143). Note that, in order to obtain a more readable figure, $\mathbf{x}_4$ was considered as a continuous vector when drawing the graphical representation of $Z_{4,143}$ but the scheduling variables $\mathbf{z}_{4,143}$ were kept integer when solving $OP_{4,143}(\mathbf{x}_4)$.

We first observe that $OP_{4,143}(\mathbf{x}_4)$ is infeasible for small values of $\mathbf{x}_4$ (see the region colored in yellow on the figure). Moreover, if $OP_{4,143}(\mathbf{x}_4^\sharp)$ is infeasible for a given design vector $\mathbf{x}_4^\sharp$, it is infeasible for all smaller design vectors: see e.g. how vector (1,3) and all smaller vectors, i.e. vectors (0,3) (0,2), (0,1), (1, 2), (1,1), (1,0) and (0,0), are infeasible. We also note that $Z_{4,143}$ is a piece-wise constant function and that for a given value of $x_{4,1}$ (resp. $x_{4,2}$), $Z_{4,143}$ is non-increasing with respect to $x_{4,2}$ (resp. $x_{4,1}$). Finally, we observe that if $OP_{4,143}(\mathbf{x}_4^\sharp)$ is feasible for a given design vector $\mathbf{x}_4^\sharp$, $Z_{4,143}(\mathbf{x}_4^\sharp)$ is a lower bound for the optimal value of $OP_{4,143}(\mathbf{x}_4)$ as long as $\mathbf{x}_4 \preceq \mathbf{x}_4^\sharp$. See e.g. how $Z_{4,143}(10, 10) = 357996$ is a lower bound for $Z_{4,143}(\mathbf{x}_4)$ as long as $\mathbf{x}_4 \preceq (10, 10)$.

Let $\mathbf{x}_\phi^\sharp \in \text{proj}_{x_\phi}(\mathcal{F}_\phi)$ be a system layout at phase ϕ feasible for each day $d \in \mathcal{D}_\phi$.

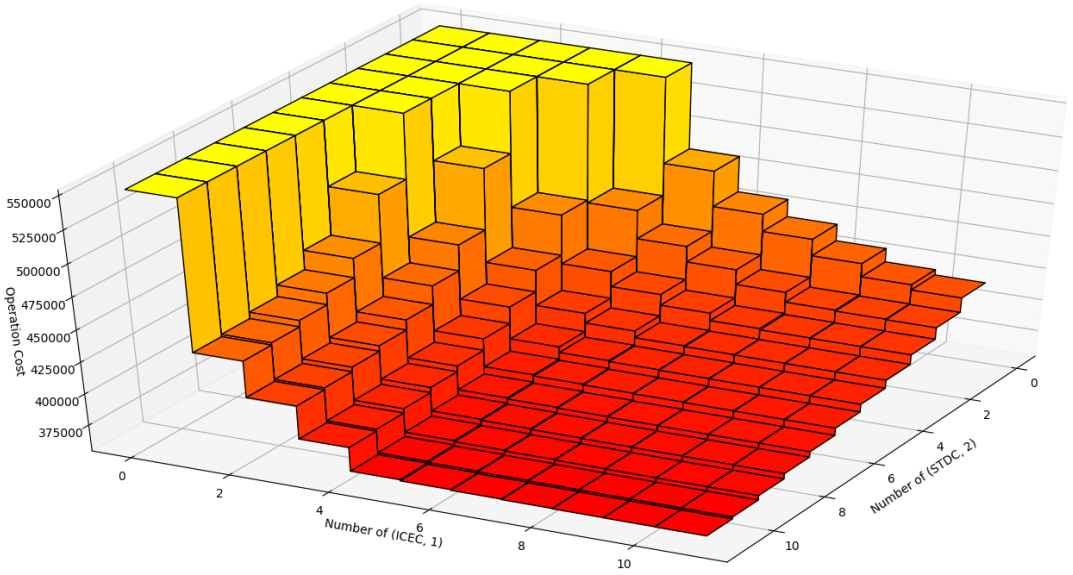


FIGURE 6.1: Operation cost of day (4,143) in the DCS of City B as a function of the number of installed chillers

As discussed above, we have, for each day $d \in \mathcal{D}_\phi$, $Z_{\phi,d}(\mathbf{x}_\phi^\sharp) \leq Z_{\phi,d}(\mathbf{x}_\phi)$ for any $\mathbf{x}_\phi$ such that $\mathbf{x}_\phi \preceq \mathbf{x}_\phi^\sharp$. In other words, $Z_{\phi,d}(\mathbf{x}_\phi^\sharp)$ is a lower bound on the operation cost of day $d \in \mathcal{D}_\phi$ as long as $\mathbf{x}_\phi$ is smaller than $\mathbf{x}_\phi^\sharp$. This allows us to define a lower bounding function for the total operation cost at phase ϕ, Z_ϕ, as follows.

$$\underline{Z}_\phi(\mathbf{x}_\phi) = Z_\phi(\mathbf{x}_\phi^\sharp)\mathbb{1}_{\mathbf{x}_\phi \preceq \mathbf{x}_\phi^\sharp} \tag{6.17}$$

where $\mathbb{1}_{\mathbf{x}_\phi \preceq \mathbf{x}_\phi^k}$ is the indicator function defined by:

$$\mathbb{1}_{\mathbf{x}_\phi \preceq \mathbf{x}_\phi^\sharp} = \begin{cases} 1 & \text{if } \mathbf{x}_\phi \preceq \mathbf{x}_\phi^\sharp \\ 0 & \text{otherwise} \end{cases} \tag{6.18}$$

We have $\underline{Z}_\phi(\mathbf{x}_\phi) = Z_\phi(\mathbf{x}_\phi^\sharp) \leq Z_\phi(\mathbf{x}_\phi)$ for any $\mathbf{x}_\phi$ smaller that $\mathbf{x}_\phi^\sharp$ and $\underline{Z}_\phi(\mathbf{x}_\phi) = 0 \leq Z_\phi(\mathbf{x}_\phi)$ for any $\mathbf{x}_\phi$ greater or incomparable to $\mathbf{x}_\phi^\sharp$. Thus $\underline{Z}_\phi(\mathbf{x}_\phi) \leq Z_\phi(\mathbf{x}_\phi)$ for all $\mathbf{x}_\phi \in \mathbb{R}^{\nu_\phi}$. $\underline{Z}_\phi$ is thus a lower bounding function of Z_ϕ. Moreover, we have $\underline{Z}_\phi(\mathbf{x}_\phi^\sharp) = Z_\phi(\mathbf{x}_\phi^\sharp)$ so that $\underline{Z}_\phi$ is strong at $\mathbf{x}_\phi^\sharp$.

Furthermore, let $\mathbf{x}_\phi^\sharp \notin \text{proj}_{x_\phi}(\mathcal{F}_\phi)$ be a system layout infeasible for some day $d \in \mathcal{D}_\phi$. As discussed above, we know than any $\mathbf{x}_\phi$ such that $\mathbf{x}_\phi$ is smaller that $\mathbf{x}_\phi^\sharp$ will also be an infeasible system layout for this day. We can thus exclude any $\mathbf{x}_\phi$ smaller that $\mathbf{x}_\phi^\sharp$ from the feasible space of the master problem by adding the following equality:

$$\mathbb{1}_{\mathbf{x}_\phi \preceq \mathbf{x}_\phi^\sharp} = 0 \tag{6.19}$$

Thus, the master problem MP^k to be solved at iteration k of a generalized Benders' decomposition algorithm may be rewritten as:

$$\min Z_{MP^k} = f_0(\mathbf{x}) + \sum_{\phi=1}^{\Phi} \eta_\phi \tag{6.20}$$

$$x \in \mathcal{X} \tag{6.21}$$

$$\eta_\phi \geq Z_\phi(\mathbf{x}_\phi^j)\mathbb{1}_{\mathbf{x}_\phi \preceq \mathbf{x}_\phi^j} \qquad \forall \phi \in \{1, \cdots, \Phi\}, \quad \forall j \in \{1, \cdots, k | \mathbf{x}_\phi^j \in \text{proj}_{x_\phi}(\mathcal{F}_\phi)\} \tag{6.22}$$

$$\mathbb{1}_{\mathbf{x}_\phi \preceq \mathbf{x}_\phi^j} = 0, \qquad \forall \phi \in \{1, \cdots, \Phi\}, \quad \forall j \in \{1, \cdots, k | \mathbf{x}_\phi^j \notin \text{proj}_{x_\phi}(\mathcal{F}_\phi)\} \tag{6.23}$$

$$\eta_\phi \in \mathbb{R} \qquad \forall \phi \in \{1, \cdots, \Phi\} \tag{6.24}$$

In this formulation, Constraints (6.22) are optimality cuts generated at each iteration j to approximate the total operation cost at phase ϕ when the iterate $\mathbf{x}_\phi^j$ belongs to $\text{proj}_{x_\phi}(\mathcal{F}_\phi)$. In case $\mathbf{x}_\phi^j \notin \text{proj}_{x_\phi}(\mathcal{F}_\phi)$, i.e. in case there is no feasible operation schedule for some day $d \in \mathcal{D}_\phi$ with the system layout $\mathbf{x}_\phi^j$, we add a feasibility cut (6.23) to forbid $\mathbf{x}_\phi^j$.

6.4 MILP reformulation of the master problem

Problem (6.20)-(6.24), as such, cannot be directly solved by an MILP solver due to the presence of a set of non-linear indicator functions $\mathbb{1}_{\mathbf{x}_\phi \preceq \mathbf{x}_\phi^j}$, $j = 1, \cdots, k$. We thus investigate in what follows a MILP reformulation of the master problem MP^k.

To this aim, we introduce two sets of binary variables. First, for each $j = 1, \cdots, k$ and each phase ϕ, the binary variable β_ϕ^j is defined as $\beta_\phi^j = 1$ if $\mathbf{x}_\phi \preceq \mathbf{x}_\phi^j$ and $\beta_\phi^j = 0$ otherwise. We thus have $\beta_\phi^j = \mathbb{1}_{\mathbf{x}_\phi \preceq \mathbf{x}_\phi^j}$. Moreover, for each $j = 1, \cdots, k$, each phase ϕ and each element $i = 1, \cdots, \nu_\phi$ of $\mathbf{x}_\phi^j$, the binary variable $\alpha_{\phi,i}^j$ is defined by $\alpha_{\phi,i}^j = 1$ if $x_{\phi,i} > x_{\phi,i}^j$ and $\alpha_{\phi,i}^j = 0$ otherwise. We thus have $\beta_\phi^j = 1$ iif $\alpha_{\phi,i}^j = 0$ for all $i = 1, \cdots, \nu_\phi$.

Thanks to these binary variables, the master problem MP^k can be reformulated as follows:

$$\min Z_{MP^k} = f_0(\mathbf{x}) + \sum_{\phi=1}^{\phi} \eta_\phi \tag{6.25}$$

$$x \in \mathcal{X} \tag{6.26}$$

$$\eta_\phi \geq Z_\phi(\mathbf{x}_\phi^j)\beta_\phi^j \qquad \forall \phi, \forall j \in \{1, \cdots, k | \mathbf{x}_\phi^j \in \text{proj}_{x_\phi}(\mathcal{F}_{\phi,d})\} \tag{6.27}$$

$$\beta_\phi^j = 0 \qquad \forall \phi, \forall j \in \{1, \cdots, k | \mathbf{x}_\phi^j \notin \text{proj}_{x_\phi}(\mathcal{F}_{\phi,d})\} \tag{6.28}$$

$$\alpha_{\phi,i}^j \leq \frac{x_{\phi,i}}{x_{\phi,i}^j + 1} \qquad \forall \phi, \forall j \in \{1, \cdots, k\}, \forall i \in \{1, \cdots, \nu_\phi\} \tag{6.29}$$

$$\alpha_{\phi,i}^j \geq \frac{x_{\phi,i} - x_{\phi,i}^j}{\overline{x}_{\phi,i} - x_{\phi,i}^j} \qquad \forall \phi, \forall j \in \{1, \cdots, k\}, \forall i \in \{1, \cdots, \nu_\phi\} \tag{6.30}$$

$$\beta_\phi^j \geq 1 - \sum_{i=1}^{\nu_\phi} \alpha_{\phi,i}^j \qquad \forall \phi, \forall j \in \{1, \cdots, k\} \tag{6.31}$$

$$\beta_\phi^j \leq \frac{\sum_{i=1}^{\nu_\phi}(1 - \alpha_{\phi,i}^j)}{\nu_\phi} \qquad \forall \phi, \forall j \in \{1, \cdots, k\} \tag{6.32}$$

$$\eta_\phi \in \mathbb{R} \qquad \forall \phi \tag{6.33}$$

$$\beta_\phi^j \in \{0, 1\} \qquad \forall \phi, \forall j \in \{1, \cdots, k\} \tag{6.34}$$

Constraints (6.27)-(6.28) are the optimality and feasibility cuts (6.22)-(6.23) in which the indicator function $\mathbb{1}_{\mathbf{x}_\phi \preceq \mathbf{x}_\phi^j}$ has been replaced by the binary variable β_ϕ^j. Constraints (6.29)-(6.30) link the value of $\alpha_{\phi,i}^j$ to the one of element i of vector $\mathbf{x}_\phi^j$. They ensure that $\alpha_{\phi,i}^j = 0$ if $x_{\phi,i} \leq x_{\phi,i}^j$ (see Constraints (6.29)) and $\alpha_{\phi,i}^j = 1$ if $x_{\phi,i} > x_{\phi,i}^j$ (see Constraints (6.30) in which $\overline{x}_{\phi,i}$ denotes an upper bound on $x_{\phi,i}$). Finally, Constraints (6.31)-(6.32) link the value of β_ϕ^j to the value of variables $\alpha_{\phi,i}^j$. Thus, Constraint (6.31) forces β_ϕ^j to be equal to 1 when all variables $\alpha_{\phi,i}^j, \forall i \in \{1, \cdots, \nu_\phi\}$, are equal to 0, i.e. when all elements of $\mathbf{x}_\phi$ are smaller that their counterpart in $\mathbf{x}_\phi^j$. Similarly, Constraint (6.32) ensures that β_ϕ^j is equal to 0 as soon as there is one element of $\mathbf{x}_\phi$ which is greater that that its counterpart in $\mathbf{x}_\phi^j$, i.e. as soon as one of the variables $\alpha_{\phi,i}^j, i \in \{1, \cdots, \nu_\phi\}$ is equal to 0. It should be mentioned that the upper bound $\overline{x}_{\phi,i}$ should numerically be set to a value large enough to ensure that the denominator of the right-hand-side of Constraints (6.30) remains strictly positive and to avoid the division-by-zero issue.

6.5 Generalized Benders' decomposition algorithm

Sections 6.2 to 6.4 described a framework for developing a generalized Benders' decomposition approach applied to our problem and provided a set of lower bounding functions displaying the sufficient mathematical properties to ensure the finite and optimal convergence of this algorithm. Overall, the outline of the algorithm can be described by Algorithm 3.

Algorithm 3: Basic generalized Benders' decomposition algorithm

Data: Parameters of commodities and technologies, construction phases, discount rate and a value of the optimality tolerance gap ϵ

Result: The optimal system deployment plan and the minimum total cost

1 Set the number of iteration k to 0;
2 Set the upper bound UB^0 to $+\infty$ and the lower bound LB^0 to $-\infty$;
3 **while** $|UB^k - LB^k| < \epsilon$ **do**
4 Set k to $k+1$;
5 Solve the master problem MP^k using formulation (6.25)-(6.34) with an MILP solver;
6 Set LB^k to Z_{MP^k} ;
7 Record the current design solution $\mathbf{x}^k = (\mathbf{x}_1^k, \cdots, \mathbf{x}_\phi^k, \cdots, \mathbf{x}_\Phi^k)$;
8 Set *IsFeasible* to *True*;
9 **for** *each phase* $\phi \in \{1, ..., \Phi\}$ **do**
10 Set $Z_\phi(\mathbf{x}_\phi^k)$ to 0;
11 **for** *each* $d \in \mathcal{D}_\phi$ **do**
12 Solve the operation sub-problem $OP_{\phi,d}(\mathbf{x}_\phi^k)$ as an MILP;
13 **if** $OP_{\phi,d}(\mathbf{x}_\phi^k)$ *is feasible* **then**
14 set $Z_\phi(\mathbf{x}_\phi^k)$ to $Z_\phi(\mathbf{x}_\phi^k) + Z_{\phi,d}(\mathbf{x}_\phi^k)$;
15 **else**
16 Add the feasibility cut (6.28) corresponding to $\mathbf{x}_\phi^k$ to MP^k;
17 Set *IsFeasible* to *False*;
18 **break** ; `/* To break the for loop started in Line 11 */`
19 **end**
20 **end**
21 **if** *IsFeasible is False* **then**
22 **break** ; `/* To break the for loop started in Line 9 */`
23 **else**
24 Add the optimality cut (6.27) corresponding to $\mathbf{x}_\phi^k$ to MP^k;
25 **end**
26 **end**
27 **if** *IsFeasible is True* **then**
28 Set $UB^k = min\{f_0(\mathbf{x}^k) + \sum_{\phi=1}^{\Phi} Z_\phi(\mathbf{x}_\phi^k), UB^{k-1}\}$;
29 **if** $|UB^k - LB^k| < \epsilon$ **then**
30 **return** $\mathbf{x}^k$ as the optimal design decision and UB^k as the minimum cost;
31 **end**
32 **end**
33 **end**

However, a direct implementation of Algorithm 3 showed some numerical inefficiencies.

First, the solution of MP^k found at Line 4 of Algorithm 3 may be far from being feasible (never mind optimal) for the operation sub-problems, especially in the first iterations of the generalized Benders' decomposition algorithm. In particular, we noted that $\mathbf{x}^k$ is sometimes neither feasible nor optimal with respect to the linear relaxation of the operation sub-problems denoted by $\widetilde{OP}_{\phi,d}(\mathbf{x}^k_\phi)$. As a consequence, when running Algorithm 3, we sometimes solve a series of operation sub-problems $OP_{\phi,d}(\mathbf{x}^k_\phi)$ as MILPs only to find out that $\mathbf{x}^k_\phi$ is infeasible with respect to the linear relaxation of one of this sub-problems. This computational effort could be avoided by first strengthening the formulation of MP^k through the use of 'classical' Benders' cuts generated using the strong duality property holding for the linear programs $\widetilde{OP}_{\phi,d}(\mathbf{x}^k_\phi)$. These cuts may be generated through a significantly smaller effort than the one required to generate cuts of type (6.27)-(6.28) as they only necessitate the resolution of a set of LPs rather than MILPs. This is why we propose to replace the resolution of MP^k with an MILP solver by the resolution of another relaxation of CP, denoted by SRP^k, through a classical Benders' decomposition algorithm implemented in an off-the-shelf MILP solver. Problem SRP^k corresponds to Problem SRP introduced in Chapter 5, i.e. to a relaxation of CP in which all integer variables $\mathbf{y}$ are relaxed to continuous variables, in which the cuts of type (6.27)-(6.28) generated up to iteration k by the generalized Benders' decomposition algorithm have been added. This allows us to obtain, at each iteration of the generalized Benders' decomposition algorithm, first-stage solutions $\mathbf{x}^k, k = 1...K$, of better quality and to solve operation sub-problems $OP_{\phi,d}(\mathbf{x}^k_\phi)$ as MILPs, which requires a significant computational effort, only to evaluate potentially 'interesting' deployment plans.

Second, similar to what was observed for the hierarchical decomposition algorithm, we often repetitively solve the same lower level sub-problems over the course of the algorithm. This repetition is due to the multi-phase nature of our problem. Namely, even if two deployment plans found by solving the master problem differ from each other, they may contain the same system layout for a subset of construction phases. This is why we record in a hash table the optimal value $Z_\phi(\mathbf{x}^k_\phi)$ of the total operation cost relative to a given phase ϕ obtained with the system layout $\mathbf{x}^k_\phi$ considered at iteration k and retrieve this cost at a later iteration $j > k$ of the algorithm in which the same system layout $\mathbf{x}^j_\phi = \mathbf{x}^k_\phi$ is used for phase ϕ.

This leads to an extended variant of the generalized Benders' decomposition algorith described in Algorithm 4.

Note that solving SRP^k using the classical Benders' decomposition algorithm embedded e.g. in CPLEX solver at the upper level amounts to adding to MP^k a set of optimality and feasibility cuts based on the resolution of the dual of each relaxed operation sub-problem $\widetilde{OP}_{\phi,d}(\cdot)$. We thus build a 'temporary' dual bounding function for each function Z_ϕ, $\phi \in \{1, \cdots, \Phi\}$, involved in Problem (6.9), which is most often not strong at iterate $\mathbf{x}^k$. However, these dual bounding functions are only used to improve the quality of iterate $\mathbf{x}^k$. Over the course of each iteration, we build a dual bounding function for each function Z_ϕ which is strong at iterate $\mathbf{x}^k$. This ensures that Algorithm 4 has a finite and optimal convergence.

Algorithm 4: Modified Generalized Benders Decomposition

Data: Parameters of commodities and technologies, construction phases and discount rate

Result: The optimal system deployment plan and the minimum total cost

For $\mathbf{c}$, add the valid inequalities (5.29)- (5.30) to the formulation of SRP^0;
Set the number of iteration k to 0.;
Set the upper bound UB^0 to $+\infty$ and the lower bound LB^0 to $-\infty$.;
while $|UB^k - LB^k| < \epsilon$ **do**
    Set k to $k+1$;
    Solve SRP^k using formulation (5.9)-(5.15) with the classical Benders' decomposition algorithm embedded in an MILP solver in which design variables $\mathbf{x}$ are the first stage variables and relaxed operation variables $(\tilde{\mathbf{y}}, \tilde{\mathbf{z}})$ are the second stage variables;
    Set $LB^k = Z_{SRP^k}$;
    Record the current design solution $\mathbf{x}^k = (\mathbf{x}_1^k, \cdots, \mathbf{x}_\phi^k, \cdots, \mathbf{x}_\Phi^k)$;
    Set *IsFeasible* to *True*;
    for *each phase* $\phi \in \{1, ..., \Phi\}$ **do**
        if $Z_\phi(\mathbf{x}_\phi^k)$ *is recorded in the hash table* **then**
            Retrieve $Z_\phi(\mathbf{x}_\phi^k)$ from the hash table;
        else
            Set $Z_\phi(\mathbf{x}_\phi^k)$ to 0;
            for *each* $d \in \mathcal{D}_\phi$ **do**
                Solve the operation sub-problem $OP_{\phi,d}(\mathbf{x}_\phi^k)$ as an MILP;
                if $OP_{\phi,d}(\mathbf{x}_\phi^k)$ *is feasible* **then**
                    set $Z_\phi(\mathbf{x}_\phi^k)$ to $Z_\phi(\mathbf{x}_\phi^k) + Z_{\phi,d}(\mathbf{x}_\phi^k)$;
                else
                    Add the feasibility cut (6.28) corresponding to $\mathbf{x}_\phi^k$ to SRP^k;
                    Set *IsFeasible* to *False*;
                    break ; `/* To break the for loop started in Line 15 */`
                end
            end
        end
        if *IsFeasible is False* **then**
            break ; `/* To break the for loop started in Line 10 */`
        end
    end
    if *IsFeasible is True* **then**
        Set $UB^k = min\{f_0(\mathbf{x}^k) + \sum_{\phi=1}^{\Phi} Z_\phi(\mathbf{x}_\phi^k), UB^{k-1}\}$;
        if $|UB^k - LB^k| < \epsilon$ **then**
            return $\mathbf{x}^k$ as the optimal design decision and UB^k as the minimum cost;
        end
    end
end

6.6 Computational experiments

We report in this section the results of the computational experiments carried out to assess the numerical performance of the extended variant of the generalized Benders' decomposition algorithm introduced in Section 6.5 and to compare it with the one of the direct resolution by an MILP solver and of the hierarchical decomposition algorithms investigated in Chapter 5. We then discuss the outcome of a post-optimization simulation study carried out to evaluate the quality of the deployment plans obtained for the two DCS projects.

6.6.1 Numerical efficiency of the extended generalized Benders' decomposition algorithm

The computational experiments were based on instances related to the three case studies introduced in Chapter 4. Each instance was solved with the extended variant of the generalized Benders' decomposition algorithm. The experimental setup was the same as the one described at the beginning of Subsection 4.4.2. We use the classical Benders' decomposition algorithm embedded in CPLEX 12.8 solver to solve SRP^k at each iteration of the algorithm.

The results are shown in Tables 6.1 - 6.4. We report, for each instance, the value of the residual gap Gap_{MIP}, the computation time and the number of iterations #Iter carried out by the generalized Benders' algorithm.

We first observe from the results displayed in Tables 6.1, 6.2 and 6.4 that the modified generalized Benders' decomposition algorithm significantly outperforms both the direct resolution approach and the extended hierarchical decomposition for all considered instances. Namely, it was able to solve to optimality the 19 considered instances within the imposed time and memory limits whereas the extended hierarchical decomposition algorithm could only solve to optimality 11 out of these 19 instances and encountered out-of-memory issues for the 7 instances related to the trigeneration project. Moreover, the modified generalized Benders' decomposition algorithm has a significantly higher efficiency than the extended hierarchical decomposition algorithm. Thus, over the 12 instances corresponding to the DCS projects in Cities A and B, the average computational time is decreased from 2279s with the extended hierarchical decomposition to 628s with the generalized Benders' decomposition, which represents a reduction of the computation time by a factor larger than 3.6.

Furthermore, since the average computational time for the 5 instances related to the DCS project in City B was less than 1 minute (see Table 6.2), we created 6 additional larger instances for this project by increasing the number of typical days taken into account to represent the demand evolution over the year. Note that the instance with $|\mathcal{D}_\phi| = 165$ takes into account the 165 days with a non-zero cooling demand in the initial time series and assigns a weight equal to 1 to each of them, it is thus equivalent to the initial problem considering the cooling demand of all 365 days of a year. We note that, although the computational time increases with the size of the instance, the generalized Benders' decomposition algorithm returns the optimal solution for these 6 new instances within the time limit.

The second observation deals with the number of iterations #Iter carried out by the generalized Benders' decomposition algorithm. It seems that, for a given case study, this number does not change when the instance size increases, i.e. when the

	$\lvert\mathcal{D}_\phi\rvert$	6	14	22	30	38	58	78
MILP	#Var	12555	29259	45963	61971	79371	121131	162984
size	#IntVar	3909	9093	14277	19245	24645	37605	50565
	#Cons	21611	50411	79211	106811	136811	208811	280839
Direct	Gap_{LP}	1.57%	3.26%	3.41%	1.77%	2.17%	2.06%	2.34%
resolution	Time	7200s	7200s	7200s	7200s	7200s	7200s	7200s
	Gap_{MIP}	0.18%	0.04%	0.13%	0.39%	0.44%	0.73%	2.83%
Initial	Gap_{LP}	1.57%	3.26%	3.41%	1.77%	2.17%	2.06%	2.34%
Hier. Dec.	$\#IFS_{SRP}$	9009	1469	3017	2764	1180	1187	488
	$\#OP_{solved}$	28150	15295	15595	31555	20335	31848	29561
	Time	7200s	7200s	7200s	7200s	7200s	7200s	7200s
	Gap_{MIP}	0.10%	0.68%	1.46%	0.50%	0.88%	0.74%	1.21%
Extended	Gap_{LP}	1.56%	1.45%	1.53%	1.56%	1.56%	1.56%	1.56%
Hier. Dec.	$\#IFS_{SRP}$	892	1980	1083	874	919	908	1171
	$\#OP_{solved}$	172	581	779	515	1254	1654	2058
	Time	123s	814s	1043s	977s	990s	1984s	6822s
	Gap_{MIP}	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Generalized	#Iter	10	7	9	8	9	9	11
Benders	Time	72s	147s	376s	593s	824s	1637s	3708s
	Gap_{MIP}	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%

TABLE 6.1: City A project: numerical comparison of the solution methods

number of typical days $|\mathcal{D}_\phi|$ increases. This means that the increase of the computation time observed when $|\mathcal{D}_\phi|$ increases mainly comes from the additional computational effort needed to solve the larger number of operation sub-problems. However, we note that the number of iterations #Iter varies significantly with the case study, e.g. from an average value of 4 iterations for City B project to an average value of 20 iterations for City C project. These variations may be first explained by the number of design variables. This one is namely smaller in City B projects (25 design variables) than in City C project (36 design variables). This means that the design feasible space is larger for City C project than for City B project, which may make it harder for the generalized Benders' decomposition algorithm to converge towards the optimal deployment plan. A second explanation might be related to the 'quality' of the deployment plans obtained while solving Problem SRP^k with a classical Benders' decomposition algorithm. This quality depends heavily on the strength of formulation SRP^k: if this formulation is strong, it is more likely to provide deployment plans which will be feasible (or even close to optimal) for the complete problem CP so that the generalized Benders' algorithm will have to carry out less iterations. This point should however be further investigated.

6.6.2 Discussion on the quality of the obtained deployment plans

Solving the MILP problem formulated in Chapter 4 to optimality does not guarantee that the obtained deployment plan is optimal with respect to the initial optimization problem. Indeed, in the MILP problem formulated in Chapter 4, the actual operation cost resulting from a given deployment plan is only computed in an approximate way. First, each initial time series representing the hourly evolution of the demand for each resource commodity or the limited availability of a supply commodity over a year is approximately represented by a limited set of representative days. Second,

	$\lvert\mathcal{D}_\phi\rvert$	6	8	10	12	14
MILP	#Var	13820	18380	22940	27500	32060
size	#IntVar	3625	4825	6025	7225	8425
	#Cons	23084	30764	38444	46124	53804
Direct	Gap_{LP}	5.32%	4.26%	4.58%	4.77%	4.38%
resolution	Time	7200s	7200s	7200s	2589s*	4411s*
	Gap_{MIP}	0.01%	0.02%	0.02%	0.04%	0.06%
Initial	Gap_{LP}	5.32%	4.26%	4.58%	4.77%	4.38%
Hier. Dec.	$\#\text{IFS}_{SRP}$	4087	1603	1303	2348	1688
	$\#\text{OP}_{solved}$	45406	35193	31752	55160	34656
	Time	5162s	7200s	7200s	7200s	7200s
	Gap_{MIP}	0.00%	0.92%	0.86%	1.60%	0.66%
Extended	Gap_{LP}	4.13%	3.37%	4.10%	4.24%	4.04%
Hier. Dec.	$\#\text{IFS}_{SRP}$	1764	1432	1441	394	1660
	$\#\text{OP}_{solved}$	505	794	618	520	1287
	Time	954s	2889s	4580s	7200s	4062s
	Gap_{MIP}	0.00%	0.00%	0.00%	0.07%	0.00%
Generalized	#Iter	4	2	2	2	3
Benders	Time	32s	23s	29s	34s	72s
	Gap_{MIP}	0.00%	0.00%	0.00%	0.00%	0.00%

"*" means that the computer ran out of memory before reaching the time limit.

TABLE 6.2: City B project: numerical comparison of the solution methods

	$\lvert\mathcal{D}_\phi\rvert$	22	30	38	58	78	165
Generalized	#Iter	4	4	4	4	4	4
Benders	Time	207s	364s	463s	991s	1593s	4370s
	Gap_{MIP}	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%

"*" includes all days with non-zero cooling demand in one year

TABLE 6.3: City B project: results obtained with the generalized Benders' decomposition algorithm on large instances

	$\lvert\mathcal{D}_\phi\rvert$	12	16	20	24	28	32	36
MILP	#Var	76080	101424	123600	148944	175872	201216	226560
size	#IntVar	16164	21540	26244	31620	37332	42708	48084
	#Cons	86420	115220	140420	169220	199820	228620	257420
Direct	Gap_{LP}	5.11%	5.17%	5.20%	5.21%	5.15%	5.14%	5.13%
resolution	Time	5934s*	6302s*	7200s	5259s*	6155s*	5968s*	5327s*
	Gap_{MIP}	0.06%	0.26%	0.23%	0.35%	0.32%	0.16%	1.21%
Initial	Gap_{LP}	5.11%	5.17%	5.20%	5.21%	5.15%	5.14%	5.13%
Hier. Dec.	$\#IFS_{SRP}$	4065	3019	2470	1888	1714	1512	1145
	$\#OP_{solved}$	35631	21560	33267	33418	37872	42548	47503
	Time	5286s*	3608s*	5405s*	5461s*	5938s*	6587s*	7200
	Gap_{MIP}	2.69%	2.21%	2.46%	2.41%	2.28%	2.88%	5.70%
Extended	Gap_{LP}	3.41%	3.65%	3.68%	3.66%	3.6%	3.58%	3.56%
Hier. Dec.	$\#IFS_{SRP}$	3646	2770	2271	1904	1583	1352	1243
	$\#OP_{solved}$	1501	1341	3177	1166	2294	2056	5189
	Time	2609s*	4127s*	2929s*	3244s*	3863s*	5060s*	3713s*
	Gap_{MIP}	0.50%	0.38%	0.71%	0.54%	0.53%	0.36%	0.72%
Generalized	#Iter	24	23	18	19	20	19	19
Benders	Time	929s	1453s	1271s	1835s	2791s	3253s	3801s
	Gap_{MIP}	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%

"*" means that the computer ran out of memory before reaching the time limit.

TABLE 6.4: Trigeneration project: numerical comparison of the solution methods

the consumption of all devices displaying non-linear performance curves is not computed exactly due to the use of a piece-wise linear approximation of these curves. A too rough estimate of the actual operation cost in the MILP problem may lead to poor design decisions and thus negatively impact the quality and relevancy of the proposed deployment plan. In this subsection, we thus seek to assess the quality of the deployment plans obtained with the proposed modeling and solution approach.

We first discuss the impact of the number of selected days $|\mathcal{D}_\phi|$ on the deployment plans. Tables 6.5-6.8 describe respectively the best feasible deployment plan found for City A, City B and City C projects obtained by solving the MILP problem formulated in Chapter 4 with the generalized Benders' decomposition algorithm as a function of the number of representative days used to estimate the operation cost. We first provide the value of the objective function (4.35), i.e. the value of the total net present cost of the deployment plan computed by the MILP model, followed by the value of the actualized design cost and the value of the actualized operation cost. We then give, for each phase ϕ in $\{1, ..., \Phi\}$, the number $\mathbf{SD}_{\phi,\mathbf{m}}$ of devices of each type to be installed in each phase, the energy storage capacity $Size^{step}_{\mathbf{m}}\mathbf{SD}_{\phi,\mathbf{m}}$ to be built and the maximum instantaneous power $C^{step}_{(ELEC,1)}\mathbf{Cdis}_{\phi,(ELEC,1)}$ to be contracted.

Results from Table 6.5 show that, for City A project, even if the total net present cost of the deployment plan slightly varies with the number of representative days, its actualized design cost and the design decisions to be made at each phase are the same in all the solutions obtained for $|\mathcal{D}_\phi|$ greater than 14 days. Similarly, we observe in Tables 6.6 - 6.7 that, for City B project, the obtained deployment plans are the same for all values of $|\mathcal{D}_\phi|$. Results from Table 6.8 also show that, for the trigeneration project, the optimal deployment plans are the same for $|\mathcal{D}_\phi|$ greater

$\lvert\mathcal{D}_\phi\rvert$	6			14			22			30			38			58			78		
Obj. value	5.50×10^8			5.74×10^8			5.74×10^8			5.75×10^8			5.74×10^8			5.73×10^8			5.74×10^8		
Design cost	2.44×10^8			2.36×10^8			2.36×10^8			2.36×10^8			2.36×10^8			2.36×10^8			2.36×10^8		
Operation cost	3.06×10^8			3.38×10^8			3.39×10^8			3.40×10^8			3.39×10^8			3.37×10^8			3.38×10^8		
Phase	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
(SMEC,1)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
(SMEC,2)	1	5	2	1	5	3	1	5	3	1	5	3	1	5	3	1	5	3	1	5	3
(SMEC,3)	0	1	0	0	1	0	0	1	0	0	1	0	0	1	0	0	1	0	0	1	0
(DMEC,1)	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
(DMEC,2)	1	0	0	1	0	0	1	0	0	1	0	0	1	0	0	1	0	0	1	0	0
Stor. cap. (MWh)	20	5	40	20	5	0	20	5	0	20	5	0	20	5	0	20	5	0	20	5	0
Cont. power (MW)	3	9	15	3	9	15	3	9	15	3	9	15	3	9	15	3	9	15	3	9	15

TABLE 6.5: City A project: deployment plan obtained as a function of the number of representative days

$\lvert\mathcal{D}_\phi\rvert$	6					8					10					12					14				
Obj. value	2.27×10^8					2.36×10^8					2.35×10^8					2.35×10^8					2.36×10^8				
Design cost	1.93×10^8					1.93×10^8					1.93×10^8					1.93×10^8					1.93×10^8				
Operation cost	3.38×10^7					4.26×10^7					4.17×10^7					4.20×10^7					4.26×10^7				
Phase	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
(SMEC, 1)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
(SMEC, 2)	0	0	1	3	3	0	0	1	3	3	0	0	1	3	3	0	0	1	4	2	0	0	1	3	3
(DMEC, 1)	1	0	1	1	1	1	0	1	1	1	1	0	1	1	1	1	0	1	0	2	1	0	1	1	1
Stor. cap. (MWh)	9	27	81	54	0	9	27	81	54	0	9	27	81	54	0	9	27	81	54	0	9	27	81	54	0
Cont. power (MW)	3	3	6	9	15	3	3	6	9	15	3	3	6	9	15	3	3	6	9	15	3	3	6	9	15

TABLE 6.6: City B project: deployment plan obtained as a function of the number of representative days

than 12 days. Overall, these results indicate that, provided the number of representative days used to estimate the operation cost is large enough and the generalized Benders' decomposition algorithm can converge within the time limit, the design decisions do not depend on the cardinality of the subset of representative days and may thus be recommended for implementation.

We then seek to further assess the impact of using a limited number of representative days and a piece-wise linear approximation of the devices' performance curves to estimate the future operation cost in our optimization model. This assessment relies on a post-optimization numerical simulation study aiming at computing as accurately as possible the actual operation cost corresponding to the deployment plan recommended by our optimization model. Due to time reasons, we carried out this assessment only on the DCS projects. Note that in these projects, all the available conversion devices have non-linear convex performance curves.

For this simulation study, we use the following procedure:

1. We retrieve the design decisions for each phase ϕ obtained when solving the MILP problem formulated in Chapter 4 to optimality. For City A project (resp.

$\lvert\mathcal{D}_\phi\rvert$	22					30					38					58					78					165				
Obj. value	2.35×10^8					2.35×10^8					2.36×10^8					2.36×10^8					2.36×10^8					2.36×10^8				
Design cost	1.93×10^8					1.93×10^8					1.93×10^8					1.93×10^8					1.93×10^8					1.93×10^8				
Operation cost	4.24×10^7					4.22×10^7					4.28×10^7					4.27×10^7					4.28×10^7					4.29×10^7				
Phase	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
(SMEC, 1)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
(SMEC, 2)	0	0	1	3	3	0	0	1	3	3	0	0	1	3	3	0	0	1	3	3	0	0	1	3	3	0	0	1	3	3
(DMEC, 1)	1	0	1	1	1	1	0	1	1	1	1	0	1	1	1	1	0	1	1	1	1	0	1	1	1	1	0	1	1	1
Stor. cap. (MWh)	9	27	81	54	0	9	27	81	54	0	9	27	81	54	0	9	27	81	54	0	9	27	81	54	0	9	27	81	54	0
Cont. power (MW)	3	3	6	9	15	3	3	6	9	15	3	3	6	9	15	3	3	6	9	15	3	3	6	9	15	3	3	6	9	15

TABLE 6.7: City B project: deployment plan obtained as a function of the number of representative days

$\lvert\mathcal{D}_\phi\rvert$	12				16				20				24				28				32				36			
Obj. value	1.64×10^8				1.64×10^8				1.62×10^8				1.63×10^8				1.63×10^8				1.63×10^8				1.64×10^8			
Design cost	5.46×10^7				5.42×10^7				5.42×10^7				5.42×10^7				5.42×10^7				5.42×10^7				5.42×10^7			
Operation cost	1.09×10^8				1.10×10^8				1.08×10^8				1.09×10^8				1.09×10^8				1.09×10^8				1.10×10^8			
Phase	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
(SMEC, 1)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
(SMEC, 2)	0	0	4	1	1	1	2	1	1	1	2	1	1	1	2	1	1	1	2	1	1	1	2	1	1	1	2	1
(DMEC, 1)	1	0	0	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
(CCHP, 1)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
(BOILER, 1)	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0
(ASHP, 1)	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0
Ice Stor. cap. (MWh)	9	9	0	18	0	0	18	18	0	0	18	18	0	0	18	18	0	0	18	18	0	0	18	18	0	0	18	18
Heat Stor. cap. (MWh)	18	0	0	45	18	0	0	36	18	0	0	36	18	0	0	36	18	0	0	36	18	0	0	36	18	0	0	36
Cont. power (MW)	1	1	1.5	2	1	1	1.5	2	1	1	1.5	2	1	1	1.5	2	1	1	1.5	2	1	1	1.5	2	1	1	1.5	2

TABLE 6.8: Trigeneration project: deployment plan obtained as a function of the number of representative days

City B project), we use the deployment plan corresponding to $|\mathcal{D}_\phi| = 78$ (resp. to $|\mathcal{D}_\phi| = 165$) representative days displayed in Table 6.5 (resp. Table 6.6).

2. We use the initial time series describing, for each phase ϕ, the hourly evolution of the cooling demand over the year and solve a sequence of 2-days scheduling problems within a rolling horizon framework. More precisely, for each phase $\phi = 1, ..., \Phi$:

   - for each day $d = 1, ..., 364$:
     - We solve a scheduling problem spanning days d and $d+1$. This scheduling problem uses as input data the system layout given for phase ϕ by the deployment plan, the cooling demand for days d and $d+1$ in the time series provided by the company technical experts and the piece-wise linear approximation of the chillers performance curve described in Subsection 4.2.3. Moreover, it considers an entering ice inventory level in the thermal storage $\mathbf{m} = (STO\_COLD, 1)$ at the beginning of day d, i.e. $\mathbf{STO}_{\phi,d,0,\mathbf{m}}$, equal to the leaving ice inventory level computed in the simulation for day $d-1$, i.e. $\mathbf{STO}^{sim}_{\phi,d-1,H,\mathbf{m}}$. For day 1, $\mathbf{STO}_{\phi,1,0,\mathbf{m}}$ is set to 0.
     - We record, for each hour $h = 0, .., H-1$ of day d, the load allocation $\mathbf{P}^{out,sim}_{\phi,d,h,n,\mathbf{m}}$ of each type of chiller together with the total operation cost $Z^{sim}_{\phi,d}$ computed by the model for day d. We also record the leaving ice inventory level, i.e. $\mathbf{STO}^{sim}_{\phi,d,H-1,\mathbf{m}}$.
   - When the scheduling problem of Day 364 and Day 365 is solved, which is the last two-day scheduling problem of the phase, the load allocation of each type of chiller of both two days, $\mathbf{P}^{out,sim}_{\phi,364,h,n,\mathbf{m}}$ and $\mathbf{P}^{out,sim}_{\phi,365,h,n,\mathbf{m}}$, as well as the corresponding total operation cost, $Z^{sim}_{\phi,364}$ and $Z^{sim}_{\phi,365}$, will be recorded.

   We finally compute, for each phase ϕ, a total 'simulated' operation cost Z^{sim}_ϕ equal to $Z^{sim}_\phi = \sum_{d=1}^{365} Z^{sim}_{\phi,d}$.

3. We use the performance curves provided by the chillers' manufacturer to compute the actual operation cost. Thus, for each phase $\phi = 1, ..., \Phi$, each day $d = 1, ..., 365$, each hour $h = 1, .., H$, each type of chiller $\mathbf{m} \in \mathcal{M}_{SMEC} \cup \mathcal{M}_{DMEC}$ and each operating mode n, we retrieve the value $\mathbf{P}^{out,sim}_{\phi,d,h,n,\mathbf{m}}$ obtained in Step 2 of the simulation procedure and use the chiller performance curve provided by its manufacturer to compute the corresponding actual electricity consumption $\mathbf{P}^{in,actual}_{\phi,d,h,ELEC,\mathbf{m}}$. This allows us to compute the total consumption of commodity (ELEC, 1)for the corresponding time-step: $\mathbf{Conso}^{actual}_{\phi,d,h,(ELEC,1)}$. We thus

obtain, for each phase ϕ, a total 'actual' operation cost Z_ϕ^{actual} equal to $Z_\phi^{actual} = \sum_{d=1}^{365} \sum_{h=0}^{H-1} \sum_{\mathbf{m} \in \mathcal{M}} \sum_{\mathbf{c} \in \mathcal{C}^R} EP_{\phi,d,h,\mathbf{c}} \mathbf{Conso}_{\phi,d,h,\mathbf{c}}^{actual}$.

The corresponding results are displayed in Table 6.9 for City A project and in Table 6.10 for City B project. We provide for each phase ϕ:

- Z_ϕ^{optim}: the value of the approximate operation cost in the objective function (4.35) of the optimization model (computed using a limited set of representative days and a piece-wise linear approximation of the chillers performance curves),
- Z_ϕ^{sim}: the value of the simulated operation cost obtained at the end of Step 2 of the simulation procedure (computing using the initial time series for the cooling demand and a piece-wise linear approximation of the chillers performance curves),
- Z_ϕ^{actual}: the value of the actual operation cost obtained at the end of Step 3 of the simulation procedure (computing using the initial time series for the cooling demand and the original non-linear chillers performance curves provided by the manufacturer),
- ϵ_ϕ^{optim}: the relative percentage difference between the value of the approximate operation cost computed by the optimization model and the value of the actual operation cost computed by the simulation procedure,
- ϵ_ϕ^{sim}: the relative percentage difference between the value of the simulated operation cost computed at Step 2 of the simulation procedure and the value of the actual operation cost computed by the simulation procedure.

The last line of each table provides the corresponding net present operation cost over the whole lifetime of the DCS.

We note from the results presented in Tables 6.9-6.10 that the approximation of the operation cost used in the optimization model is of very good quality. Namely, the relative percentage difference between Z_ϕ^{optim} and Z_ϕ^{actual}, ϵ_ϕ^{optim}, stays below 1% in all but one cases. The only large percentage difference is observed for Phase 1 of City B project. It is mainly explained by the fact that the demand in year 1 of this project is weak and that, in this case, it may be profitable to produce in one time-step an amount of ice covering the total cooling demand over the next two forthcoming days. In the optimization model, such a production strategy in which ice is kept in inventory for more than one day is not exploited due to the use of separate representative days. However, this has a limited impact on the relative percentage difference between the total approximated cost over all phases, $Z^{optim} = \sum_{\phi=1}^{\Phi} \beta_\phi Z_\phi^{optim}$, and the total actual cost over all phases, $Z_\phi^{actual} = \sum_{\phi=1}^{\Phi} \beta_\phi Z_\phi^{actual}$, as the relative difference stays below 1% for both projects. We thus conclude that the use of a large number of representative days and of a 3-segment piece-wise linear approximation of the chillers performance curves leads to an estimation of the actual electric consumption of the system in the optimization model accurate enough to provide good-quality deployment plans. It would clearly be interesting to carry out such a post-optimization simulation study for the trigeneration project.

6.7 Conclusion

This chapter was devoted to the presentation of a generalized Benders' decomposition algorithm for the LMES design problem.

Phase	Z_ϕ^{optim}	ϵ^{optim}	Z_ϕ^{sim}	ϵ^{sim}	Z_ϕ^{actual}
1	4.38×10^6	-0.75%	4.38×10^6	-0.83%	4.41×10^6
2	1.89×10^7	-0.90%	1.88×10^7	-1.12%	1.90×10^7
3	3.15×10^8	-0.70%	3.14×10^8	-0.87%	3.17×10^8
Total	3.38×10^8	-0.71%	3.38×10^8	-0.88%	3.41×10^8

TABLE 6.9: City A project: comparison of the approximated, simulated and actual operation cost

Phase	Z_ϕ^{optim}	ϵ^{optim}	Z_ϕ^{sim}	ϵ^{sim}	Z_ϕ^{actual}
1	5.17×10^4	3.85%	4.96×10^4	-0.41%	4.98×10^4
2	5.85×10^5	-0.86%	5.88×10^5	-0.26%	5.90×10^5
3	1.63×10^6	-0.22%	1.62×10^6	-0.38%	1.63×10^6
4	2.82×10^6	-0.75%	2.83×10^6	-0.45%	2.84×10^6
5	3.76×10^7	-0.98%	3.77×10^7	-0.68%	3.79×10^7
Total	4.26×10^7	-0.93%	4.28×10^7	-0.65%	4.30×10^7

TABLE 6.10: City B project: comparison of the approximated, simulated and actual operation cost

We first presented in Section 6.2 the overall framework used to develop this algorithm. This framework basically consists in reformulating the problem using only design variables and in using a set of functions to represent the impact on the operation cost at a given investment phase of the design decisions. As there is no closed-loop expression for these functions, the algorithm will iteratively build an approximation, called a dual bounding function, for each of them.

The finite and optimal convergence of a generalized Benders' decomposition algorithm requires that these dual bounding functions are strong at all the iterates. We thus showed in Section 6.3 how such dual bounding functions may be used in our problem. This was achieved by exploiting the special structure of the constraints coupling design and operation variables in the operation sub-problems. We then explained in Section 6.4 how the resulting non-linear master problem may be reformulated as an MILP thanks to the introduction of a set of additional binary variables. Finally, Section 5.3 summarized the proposed generalized Benders' decomposition algorithm and highlighted the extension of the basic variant of the algorithm we introduced in order to improve its numerical efficiency.

Finally, Section 6.6 reported the results obtained on instances based on our three case studies. These results show that the generalized Benders' decomposition algorithm clearly outperforms the solution approaches investigated in the previous chapters. In particular, using this algorithm, we were able to solve to optimality all considered instances within the time and memory limits. Furthermore, we presented the results of a post-optimization simulation study carried out to estimate the quality of the deployment plans obtained with the proposed approach. These results show that, even if some approximations and simplifications were made when modeling the actual optimization problem, the obtained deployment plans are of good quality and may be recommended for a real-life implementation.

Chapter 7

Conclusion

7.1 Conclusion

This PhD thesis aimed to develop a numerical decision-aid tool based on mathematical optimization to help our industrial partner, EDF Chine, at optimally designing a Local Multi-Energy System (LMES). The LMES optimal design problem consists in choosing a combination of devices belonging to various energy conversion and storage technologies to compose a district energy system which can satisfy the forecasted energy demands of the local residents and minimize the total cost comprising the installation cost and the operation cost over the system's lifetime.

We provided in Chapter 2 an overview of the current state of the art on this problem. In particular, an analysis of the papers proposing mathematical programming models for the LMES design problem was carried out while focusing on four important features: the modeling of the design decisions (selection and sizing of the technologies), the representation of the performance curves of the conversion devices, the incorporation of an imposed minimum load for conversion devices and the presence of storage devices. This analysis enabled us to see that none of the previously published works could meet the needs of our industrial partner in terms of problem modeling. Namely, none of the reviewed works consider that building an LMES is a multi-phase project in which the system is gradually built and extended over the years. Moreover, the modeling of the design decisions is often oversimplified by assuming that a technology may be sized by choosing its capacity within a continuous range. This is not practically relevant as the sizing of a technology depends in fact of the models available in the catalog of the equipment manufacturer.

Chapter 3 was devoted to a detailed description of the LMES design problem. We started by introducing the two main components of an LMES: the commodities and the energy conversion and storage technologies. We explained how the LMES design problem uses as a starting point a description of the LMES superstructure which can be seen as a pre-selection of the technologies and corresponding devices considered for installation in the system. We then described a number of additional aspects to be considered when designing an LMES, in particular the fact that building an LMES requires a multi-phase deployment plan and that detailed operation schedules need to be built to accurately estimate the operation cost of the system.

In Chapter 4, we discussed the problem modeling and its formulation as an MILP. The proposed model extends the current state of the arts by simultaneously considering: (i) multiple construction phases, (ii) a realistic choice of the energy conversion and storage devices among the models available at the manufacturer, (iii) a careful selection of a number of representative days sufficiently large to obtain a good estimation of the actual operation cost of the system, (iv) a realistic representation of the actual functioning of the conversion devices in terms of minimum working load and partial-load efficiency. As taking into account all these aspects of

the problem leads to the formulation of a huge mixed-integer non-linear program which, as such, is computationally intractable, we proposed a modeling approach aiming at reducing to some extent the size of the obtained mathematical program and at removing the non-linearities. This was achieved among others by using investment phases spanning several years, using a medium number of representative days to estimate the operation cost, building piece-wise linear approximations of the performance curves and exploiting the convexity of these curves. The proposed MILP formulation was tested on instances based on three case studies corresponding to two DCSs and a trigeneration system currently under development by EDF in China. However, significant numerical difficulties were encountered while trying to directly solve the proposed MILP formulation with CPLEX 12.8 solver.

We thus investigated the development of two decomposition-based approaches. These approaches exploit the bi-level nature of our problem, i.e. the fact that there is a natural hierarchy between the design decision variables used to determine the system deployment plan and the operation decision variables used to build operation schedules and to estimate the future operation cost of the system. This hierarchy translates into a special structure of the mathematical problem. Namely, once the design decisions determining the system layout at each investment phase are fixed, the problem decomposes into a set of independent operation sub-problems. Each of these sub-problems is a small-size MILP aiming at building an optimal operation schedule for a given representative day under the system layout determined for the corresponding phase.

The first algorithm was discussed in Chapter 5. It consists in an extension of the hierarchical decomposition algorithm presented by [Yok+15]. This algorithm is based on the resolution of a semi-relaxed problem (SRP), i.e. of a relaxation of the original MILP in which all the integrality restrictions on the discrete operation variables are relaxed, by the built-in branch-and-cut algorithm of an MILP solver. Each time an integer feasible solution of SRP, i.e. a potential system deployment plan involving only integer design variables, is found in the B&C search tree, a sequence of operation sub-problems (OP) in which the operation variables are kept discrete when relevant is solved. This enables to first check the feasibility of the potential system deployment plan with respect to the operation constraints and second to compute the actual operation cost of this deployment plan. The nodes in the upper level B&C search tree are only pruned when their bound exceeds the value of the actual design and operation cost of the currently best known feasible deployment plan, so that the optimality of the final solution is guaranteed. We proposed a new extension of this hierarchical decomposition algorithm to make it more efficient at solving our multi-phase design problem. This was achieved in particular through the addition of single-phase no-good cuts into the upper-level problem to forbid deployment plans using a system layout found to be infeasible for a certain investment phase, through the recording of the solution value of previously solved operation sub-problems to avoid repetitive calculations and through the use of a small set of valid inequalities. Our numerical results showed that the proposed extended hierarchical decomposition algorithm significantly outperforms both the initial hierarchical decomposition algorithm and the direct resolution by a mathematical solver on the instances based on the two DCS projects. However, for the instances based on the trigeneration system, which has a more complex structure than the two DCSs, the extended hierarchical decomposition algorithm could not find the optimal solution within the computation time and memory limits.

This is why we investigated the development of a generalized Benders' decomposition algorithm by using the generic framework provided in [BR21]. This algorithm can be seen as an extension of the conventional Benders' decomposition algorithm to the case where the second-stage sub-problems involve integer variables. It is based on the use of a master problem involving only design variables and of a set of dual bounding functions providing an estimation of the operation cost at each investment phase as a function of the system layout fixed for this phase by the design variables. The optimal and finite convergence of this algorithm requires among others that these dual bounding functions are strong. This was achieved in our case by proposing new optimality and feasibility cuts exploiting the special structure of the constraints linking the design and operation variables in our problem. Moreover, we developed an extended variant of this generalized Benders' decomposition algorithm in order to further improve its efficiency. This extended variant relies among others on the resolution of the semi-relaxed problem mentioned above by the classical Benders' decomposition algorithm embedded in an MILP solver. This allows us to improve the convergence speed of the algorithm by improving the quality of the potential deployment plan evaluated at each iteration of the generalized Benders' decomposition algorithm. Our computational results showed that the extended variant of the generalized Benders' decomposition algorithm clearly outperforms all the solution approaches investigated in the previous chapters. In particular, using this algorithm, we were able to solve to optimality all the considered instances, even the ones related to the trigeneration project, within the time and memory limits. However, solving the MILP problem to optimality does not guarantee that the obtained deployment plan is optimal with respect to the initial optimization problem as some approximations were made during the problem modeling. This is why we conducted a post-optimization simulation study to estimate the quality of the deployment plans obtained for the two DCS projects with the proposed approach. The outcome of this study was that, even if some approximations and simplifications were made when modeling the actual optimization problem, the obtained deployment plans are of good quality and may confidently be recommended for a real-life implementation.

Note that the work related to the hierarchical decomposition approach introduced in Chapter 5 was presented in several national conferences [Liu+19; Liu+20a; Liu+21b] and was published in the proceedings of an international conference [Liu+21a]. A paper version was also submitted to an international journal [Liu+].

7.2 Research perspectives

The work presented in this manuscript raises several research perspectives that would be worth further investigation.

First, an interesting direction for further research might be to study the hybridisation of the hierarchical decomposition and generalized Benders' decomposition algorithms. For instance, the optimality and feasibility cuts of the generalized Benders' decomposition algorithm could be added during the resolution of the semi-relaxed problem (SRP) by the upper level B&C algorithm of the hierarchical decomposition algorithm each time an integer feasible solution of SRP is found in the B&C search tree. This would avoid repetitively solving to optimality a sequence of variants of the semi-relaxed problem (SRP), differing from one another only by the addition of a small set of optimality and feasibility cuts, as is currently done in the generalized Benders' decomposition algorithm. However, this hybridisation is not

straightforward and raises several difficulties. Namely, generating the optimality and feasibility cuts of the generalized Benders' decomposition algorithm requires the introduction of additional binary variables into the problem formulation during its resolution by a B&C algorithm (see Section 6.4). This means that the B&C algorithm would become a branch-and-price-and-cut algorithm. Furthermore, even if using the hybrid algorithm may reduce the time needed for obtaining integer feasible solutions of the original problem, it will involve however carrying out an accurate computation (through the resolution of a set of scheduling sub-problems) of the operation cost of potential deployment plans that may be of lesser quality than the optimal solution of the semi-relaxed sub-problem SRP. This may negatively impact the overall computation time of this hybrid approach.

Second, in terms of problem modeling, the MILP formulation we proposed for the LMES design problem relies on the assumption that the performance curves of all the conversion devices are linear or convex. This convexity has two advantages. First, it allows us to consider the epigraph of the curve rather than the curve itself so that the energy consumption of a device can be accurately computed as a function of its energy production without introducing auxiliary binary variables in the formulation. Second, the operation schedule can be built at the aggregate level for each set of identical devices (using a small number of integer variables) rather than at the detailed level for each individual device (using a larger number of binary variables). However, in case this assumption on the convexity of the performance curves does not hold, the formulation becomes more difficult to deal with. The piece-wise linear approximation method discussed in Subsection 4.2.3 can still be applied to approximate the non-convex performance curves. But the exact computation of the energy consumption of a device as a function of its energy production will require the use of formulation techniques based on auxiliary binary variables or Special Ordered Set (SOS) variables (see e.g. [CPSM14]). Furthermore, it may not be possible anymore to build aggregate schedules for each set of identical devices. Namely, this aggregation is made possible by the property that all active identical devices should work at the same load when the performance curves are convex. This property is however not valid in the presence of non-convex performance curves. This would force us to build a detailed schedule for each individual device, which would require the introduction a large number of binary variables to represent the on/off status (and in some cases the operating mode) of each conversion device. Moreover, symmetry problems coming from the fact that there will exist many alternative optimal solutions that differ from one another only by the way each detailed operation schedule is assigned to one of the identical conversion devices may arise and lead to additional numerical difficulties.

Taking into account non-convex performance curves would thus complicate the resolution of the problem by the proposed decomposition algorithms, leading to a potential loss of numerical efficiency, but this resolution would still be possible. In contrast, some modeling extensions might make it impossible to use these decomposition algorithms. Namely, both the hierarchical decomposition and the generalized Benders' decomposition algorithms require that all design variables are discrete. In our case, this requirement could be rather easily met as the number of conversion devices to be installed should be chosen as an integer and the size of the energy storage to be built should be an integer multiple of a predefined small discrete capacity unit. However, in case a design variable must be considered as continuous, the optimal convergence of the two decomposition algorithms investigated in this work is not guaranteed anymore and we will have to solve the large-scale MILP problem directly with a mathematical solver. Furthermore, both decomposition algorithms are

based on the key property that, once the design decisions are fixed, the problem decomposes into a set of independent scheduling sub-problems. This property might not hold anymore if the LMES to be designed involves long-term seasonal energy storage. This technology consists e.g. in producing and storing ice in winter to use it several months later to provide cold during summer. It thus differs from the short-term daily energy storage considered in our work. Taking into account a seasonal energy storage means introducing in the MILP formulation an additional coupling (through a set of constraints) between the operation scheduling sub-problems. The development of a new solution approach might be required to handle this difficulty.

Finally, the current decision-aid tool uses as input data forecasts about the long-term evolution of the energy demand, resource commodities price and availability. It assumes that these forecasts are accurate and use this information to build a system deployment plan using a deterministic optimization approach. However, these long-term forecasts are subject to many uncertainties. For instance, the forecasted energy demand which, in our case studies, features a clear upward trend over the years might in fact decrease drastically, stay stable or simply increase in an smaller extent. In these cases, the LMES designed using the initial forecasts might be significantly oversized, leading to difficulties for meeting the demand in some time-steps (due to the minimum working load of the conversion devices) and to investment costs higher than necessary. This problem may be partly mitigated through the use of a stochastic optimization approach such as multi-stage stochastic programming. Multi-stage stochastic programming is a stochastic optimization paradigm based on the assumptions that a probabilistic description of the uncertain parameters is available and that some decisions do not have to be made right now but rather might be differed to a later point in time. When applied to our LMES design problem, this would mean that we have on hand some probabilistic information about the long-term demand forecast errors and that the design decisions relative to a future investment phase do not have to be made now but rather can be postponed to the beginning of this phase, at a point in time where more accurate information about the energy demands and resource commodities prices will be available. Such a multi-stage stochastic programming approach would typically rely on a scenario tree to represent the potential evolution of the input parameters over the investment phases. It would result in the formulation of an MILP displaying the same overall structure as the one investigated in this work. However, the size of this MILP will be drastically increased as it will be broadly proportional to the number of nodes in the scenario tree rather than to the number of investment phases involved in the investment planning horizon. Solving this MILP efficiently will thus be particularly challenging.

Bibliography

[AB16] Mohammad Ameri and Zahed Besharati. "Optimal design and operation of district heating and cooling networks with CCHP systems in a residential complex". In: *Energy and Buildings* 110 (2016), pp. 135–148.

[AFF07] Paolo Arcuri, Gaetano Florio, and Petronilla Fragiacomo. "A mixed integer programming model for optimal design of trigeneration in a hospital complex". In: *Energy* 32.8 (2007), pp. 1430–1447.

[Alg+20] Dana M Alghool, Tarek Y Elmekkawy, Mohamed Haouari, and Adel Elomri. "Optimization of design and operation of solar assisted district cooling systems". In: *Energy Conversion and Management: X* 6 (2020), p. 100028.

[ANKH19] Fadi Al-Noaimi, Reem Khir, and Mohamed Haouari. "Optimal design of a district cooling grid: structure, technology integration, and operation". In: *Engineering Optimization* 51.1 (2019), pp. 160–183.

[Bal21] Balmorel. *Balmorel*. 2021.

[BBV04] Stephen Boyd, Stephen P Boyd, and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[Ben62] J.F. Benders. "Partitioning procedures for solving mixed-variables programming problems". In: *Numerische Mathematik* 4 (1962), pp. 238–252.

[Bis+14] Aldo Bischi, Leonardo Taccari, Emanuele Martelli, Edoardo Amaldi, Giampaolo Manzolini, Paolo Silva, Stefano Campanari, and Ennio Macchi. "A detailed MILP optimization model for combined cooling, heat and power system operation planning". In: *Energy* 74 (2014), pp. 12–26.

[Bis+19] Aldo Bischi, Leonardo Taccari, Emanuele Martelli, Edoardo Amaldi, Giampaolo Manzolini, Paolo Silva, Stefano Campanari, and Ennio Macchi. "A rolling-horizon optimization algorithm for the long term operational scheduling of cogeneration systems". In: *Energy* 184 (2019), pp. 73–90.

[Bix12] Robert E. Bixby. *A Brief History of Linear and Mixed-Integer Programming Computation*. 2012.

[BR21] Suresh Bolusani and Ted K Ralphs. "A Framework for Generalized Benders' Decomposition and Its Application to Multilevel Optimization". In: *arXiv preprint arXiv:2104.06496* (2021).

[BSW07] Bjorn H Bakken, Hans I Skjelbred, and Ove Wolfgang. "eTransport: Investment planning in energy supply systems with multiple energy carriers". In: *Energy* 32.9 (2007), pp. 1676–1689.

[CEA+15] W Carmen, S Eric, M Adrian, et al. "An energy integrated, multi-microgrid, MILP approach for residential distributed energy system planning-A South Australian case-study". In: *Energy* 85 (2015), pp. 30–44.

[Che+20] Yixing Chen, Chuhao Yang, Xiao Pan, and Da Yan. "Design and operation optimization of multi-chiller plants based on energy performance simulation". In: *Energy and Buildings* 222 (2020), p. 110100.

[CPSM14] Carlos M Correa-Posada and Pedro Sánchez-Martín. "Gas network optimization: A comparison of piecewise linear models". In: *Optimization Online* (2014).

[CSL11] Monica Carvalho, Luis Maria Serra, and Miguel Angel Lozano. "Optimal synthesis of trigeneration systems subject to environmental constraints". In: *Energy* 36.6 (2011), pp. 3779–3790.

[DC21] DER-CAM. *DER-CAM*. 2021.

[DLM10] Claudia D'Ambrosio, Andrea Lodi, and Silvano Martello. "Piecewise linear approximation of functions of two variables in MILP models". In: *Operations Research Letters* 38.1 (2010), pp. 39–46.

[EA19] Valerie Eveloy and Dereje S. Ayou. "Sustainable District Cooling Systems: Status, Challenges, and Future Opportunities, with Emphasis on Cooling-Dominated Regions". In: *Energies* 12.2 (2019).

[EDF22a] EDF China. *Sanmexia urban heating*. Accessed 2022-03-28. 2022. URL: https://asia.edf.com/en/edf-in-asia/activities/energy-services-and-grid-management/sanmenxia-urban-heating.

[EDF22b] EDF China. *Sanya integrated energy*. Accessed 2022-03-28. 2022. URL: https://asia.edf.com/en/edf-in-asia/activities/energy-services-and-grid-management/sanya-integrated-energy.

[Els+17] Cristina Elsido, Aldo Bischi, Paolo Silva, and Emanuele Martelli. "Two-stage MINLP algorithm for the optimal synthesis and design of networks of CHP units". In: *Energy* 121 (2017), pp. 403–426.

[Ene21] EnergyPLAN. *EnergyPLAN*. 2021.

[FBM14] Samira Fazlollahi, Gwenaelle Becker, and François Maréchal. "Multi-objectives, multi-period optimization of district energy systems: II-Daily thermal storage". In: *Computers & Chemical Engineering* 71 (2014), pp. 648–662.

[FIC21] FICO. *Xpress Optimizer:* 2021.

[FS16] T. Falke and A. Schnettler. "Investment planning of residential energy supply systems using dualdynamic programming". In: *Sustainable Cities and Society* 23 (2016), pp. 16–22.

[Gab+18] Paolo Gabrielli, Matteo Gazzani, Emanuele Martelli, and Marco Mazzotti. "Optimal design of multi-energy systems with seasonal storage". In: *Applied Energy* 219 (2018), pp. 408–424.

[Gan+16] Wenjie Gang, Shengwei Wang, Fu Xiao, and Dian-ce Gao. "District cooling systems: Technology integration, system optimization, challenges and opportunities for applications". In: *Renewable and Sustainable Energy Reviews* 53 (2016), pp. 253–264.

[GCW19] Sebastian Goderbauer, Martin Comis, and Felix JL Willamowski. "The synthesis problem of decentralized energy systems is strongly NP-hard". In: *Computers & Chemical Engineering* 124 (2019), pp. 343–349.

[Gur21] Gurobi. *Gurobi Optimizer:* 2021.

[HO03] J. Hooker and G. Ottosson. "Logic-based Benders decomposition". In: *Mathematical Programming Series A* 96 (2003), 33–60.

[IBM21] IBM. *Cplex Optimizer:* 2021.

[IG98] RR Iyer and Ignacio E Grossmann. "Synthesis and operational planning of utility systems for multiperiod operation". In: *Computers & Chemical Engineering* 22.7-8 (1998), pp. 979–993.

[JFS14] Mark Jennings, David Fisk, and Nilay Shah. "Modelling and optimization of retrofitting residential energy systems at the urban scale". In: *Energy* 64 (2014), pp. 220–233. ISSN: 0360-5442.

[KH15] Reem Khir and Mohamed Haouari. "Optimization models for a single-plant District Cooling System". In: *European Journal of Operational Research* 247.2 (2015), pp. 648–658.

[KV21] Christian Klemm and Peter Vennemann. "Modeling and optimization of multi-energy systems in mixed-use districts: A review of existing methods and approaches". In: *Renewable and Sustainable Energy Reviews* 135 (2021), p. 110206.

[LB15] X. Li and P.I. Bardon. "Optimal design and operation of energy systems under uncertainty". In: *Journal of Process Control* 30 (2015), pp. 1–9.

[Liu+] Bingqian Liu, Côme Bissuel, François Courtot, Céline Gicquel, and Dominique Quadri. "A Mixed-Integer Linear Programming Model and Hierarchical Decomposition Approach for Optimizing the Multi-phase Deployment Plan of a District Cooling System". Submitted to *Applied Energy*.

[Liu+19] Bingqian Liu, Côme Bissuel, François Courtot, Céline Gicquel, and Dominique Quadri. "Optimal Design of a District Cooling System by Mixed-Integer Linear Programming". In: PGMO Days, Paris, France. Dec. 2019.

[Liu+20a] Bingqian Liu, Côme Bissuel, François Courtot, Céline Gicquel, and Dominique Quadri. "Optimal Design of a District Cooling System by Mixed-Integer Linear Programming". In: Congrès de la société française de Recherche Opérationnelle et d'Aide à la décision ROADEF 2021,Montpellier, France. Feb. 2020.

[Liu+20b] Zuming Liu, Mei Qi Lim, Markus Kraft, and Xiaonan Wang. "Simultaneous design and operation optimization of renewable combined cooling heating and power systems". In: *AIChE Journal* 66.12 (2020), e17039.

[Liu+21a] Bingqian Liu, Côme Bissuel, François Courtot, Céline Gicquel, and Dominique Quadri. "A Hierarchical Decomposition Approach for the Optimal Design of a District Cooling System". In: *Proceedings of the 10th International Conference on Operations Research and Enterprise Systems (ICORES 2021)*. SCITEPRESS-Science and Technology Publications. 2021, pp. 317–328.

[Liu+21b] Bingqian Liu, Côme Bissuel, François Courtot, Céline Gicquel, and Dominique Quadri. "Optimizing the Multi-phase Deployment of a District Cooling System by Mixed-Integer Linear Programming". In: Congrès de la société française de Recherche Opérationnelle et d'Aide à la décision ROADEF 2021, Mulhouse, France. Apr. 2021.

[Lop+18] Peter Lopion, Peter Markewitz, Martin Robinius, and Detlef Stolten. "A review of current challenges and trends in energy systems modeling". In: *Renewable and sustainable energy reviews* 96 (2018), pp. 156–166.

[Man14] Pierluigi Mancarella. "MES (multi-energy systems): An overview of concepts and evaluation models". In: *Energy* 65 (2014), pp. 1–17.

[MEC16] Boran Morvaj, Ralph Evins, and Jan Carmeliet. "Optimization framework for distributed energy systems with integrated electrical grid constraints". In: *Applied energy* 171 (2016), pp. 296–313.

[Meh+12] Eugenia D. Mehleri, Haralambos Sarimveis, Nikolaos C. Markatos, and Lazaros G. Papageorgiou. "A mathematical programming approach for optimal design of distributed energy systems at the neighbourhood level". In: *Energy* 44.1 (2012). Integration and Energy System Engineering, European Symposium on Computer-Aided Process Engineering 2011, pp. 96–104. ISSN: 0360-5442.

[Meh+13] Eugenia D Mehleri, Haralambos Sarimveis, Nikolaos C Markatos, and Lazaros G Papageorgiou. "Optimal design and operation of distributed energy systems: Application to Greek residential sector". In: *Renewable Energy* 51 (2013), pp. 331–342.

[OHO16] Akomeno Omu, Shanshan Hsieh, and Kristina Orehounig. "Mixed integer linear programming for the design of solar thermal energy systems with short-term storage". In: *Applied Energy* 180 (2016), pp. 313–326.

[Pow64] M.J.D. Powell. "An efficient method for finding the minimum of a function of several variables without calculating derivatives". In: *The Computer Journal* 7 (1964), pp. 155–162.

[PW06] Yves Pochet and Laurence A Wolsey. *Production planning by mixed integer programming*. Springer Science & Business Media, 2006.

[Ren+10] Hongbo Ren, Weisheng Zhou, Ken'ichi Nakagami, Weijun Gao, and Qiong Wu. "Multi-objective optimization for the operation of distributed energy systems considering economic and environmental aspects". In: *Applied Energy* 87.12 (2010), pp. 3642–3651.

[Sch+18] Thomas Schütz, Xiaolin Hu, Marcus Fuchs, and Dirk Müller. "Optimal design of decentralized energy conversion systems for smart microgrids using decomposition methods". In: *Energy* 156 (2018), pp. 250–263.

[Ste+15] David Steen, Michael Stadler, Gonçalo Cardoso, Markus Groissböck, Nicholas DeForest, and Chris Marnay. "Modeling of thermal storage systems in MILP distributed energy resource models". In: *Applied Energy* 137 (2015), pp. 782–792.

[Sö07] Jarmo Söderman. "Optimisation of structure and operation of district cooling networks in urban regions". In: *Applied Thermal Engineering* 27 (Nov. 2007), pp. 2665–2676.

[Tso+20] William W Tso, C Doga Demirhan, Clara F Heuberger, Joseph B Powell, and Efstratios N Pistikopoulos. "A hierarchical clustering decomposition algorithm for optimizing renewable power systems with storage". In: *Applied Energy* 270 (2020), p. 115190.

[UDT19] Luca Urbanucci, Francesco D'Ettorre, and Daniele Testi. "A comprehensive methodology for the integrated optimal sizing and operation of cogeneration systems with thermal energy storage". In: *Energies* 12.5 (2019), p. 875.

[VBGS15] I Van Beuzekom, M Gibescu, and JG Slootweg. "A review of multi-energy system planning and optimization tools for sustainable urban development". In: *2015 IEEE Eindhoven PowerTech* (2015), pp. 1–7.

[Vie15] Juan Pablo Vielma. "Mixed integer linear programming formulation techniques". In: *Siam Review* 57.1 (2015), pp. 3–57.

[Wak+19] Tetsuya Wakui, Moe Hashiguchi, Kento Sawada, and Ryohei Yokoyama. "Two-stage design optimization based on artificial immune system and mixed-integer linear programming for energy supply networks". In: *Energy* 170 (2019), pp. 1228–1248.

[WHY21] Tetsuya Wakui, Moe Hashiguchi, and Ryohei Yokoyama. "Structural design of distributed energy networks by a hierarchical combination of variable-and constraint-based decomposition methods". In: *Energy* 224 (2021), p. 120099.

[Wie+18] Frauke Wiese, Rasmus Bramstoft, Hardi Koduvere, Amalia Pizarro Alonso, Olexandr Balyk, Jon Gustav Kirkerud, Åsa Grytli Tveten, Torjus Folsland Bolkesjø, Marie Münster, and Hans Ravn. "Balmorel open source energy system model". In: *Energy strategy reviews* 20 (2018), pp. 26–34.

[WMF07] Céline Weber, François Maréchal, and Daniel Favrat. "Design and optimization of district energy systems". In: *Computer Aided Chemical Engineering*. Vol. 24. Elsevier, 2007, pp. 1127–1132.

[Wol20] Laurence A Wolsey. *Integer programming*. John Wiley & Sons, 2020.

[Xie+20] Shiwei Xie, Zhijian Hu, Jueying Wang, and Yuwei Chen. "The optimal planning of smart multi-energy systems incorporating transportation, natural gas and active distribution networks". In: *Applied Energy* 269 (2020), p. 115006.

[YI06] Ryohei Yokoyama and Koichi Ito. "Optimal design of gas turbine cogeneration plants in consideration of discreteness of equipment capabilities". In: (2006).

[Yok+15] Ryohei Yokoyama, Yuji Shinano, Syusuke Taniguchi, Masashi Ohkura, and Tetsuya Wakui. "Optimization of energy supply systems by MILP branch and bound method in consideration of hierarchical relationship between design and operation". In: *Energy Conversion and Management* 92 (2015), pp. 92–104.

[YZX15] Yun Yang, Shijie Zhang, and Yunhan Xiao. "Optimal design of distributed energy resource systems coupled with energy distribution networks". In: *Energy* 85 (2015), pp. 433–448.

[Zat+19] Matteo Zatti, Marco Gabba, Marco Freschini, Michele Rossi, Agostino Gambarotta, Mirko Morini, and Emanuele Martelli. "k-MILP: A novel clustering approach to select typical and extreme days for multi-energy systems design optimization". In: *Energy* 181 (2019), pp. 1051–1063.

[Zho+13a] Zhe Zhou, Pei Liu, Zheng Li, Efstratios N Pistikopoulos, and Michael C Georgiadis. "Impacts of equipment off-design characteristics on the optimal design and operation of combined cooling, heating and power systems". In: *Computers & Chemical Engineering* 48 (2013), pp. 40–47.

[Zho+13b] Zhe Zhou, Jianyun Zhang, Pei Liu, Zheng Li, Michael C Georgiadis, and Efstratios N Pistikopoulos. "A two-stage stochastic programming model for the optimal design of distributed energy systems". In: *Applied Energy* 103 (2013), pp. 135–144.