ABSTRACT

Title of Dissertation: **OPTIMUM DESIGN AND OPERATION OF COMBINED COOLING HEATING AND POWER SYSTEM WITH UNCERTAINTY**

Lei Gao, Doctor of Philosophy, 2022

Dissertation directed by: Professor Reinhard Radermacher, Department of Mechanical Engineering

Combined cooling, heating, and power (CCHP) systems utilize renewable energy sources, waste heat energy, and thermally driven cooling technology to simultaneously provide energy in three forms. They are reliable by virtue of main grid independence and ultra-efficient because of cascade energy utilization. These merits make CCHP systems potential candidates as energy suppliers for commercial buildings. Due to the complexity of CCHP systems and environmental uncertainty, conventional design and operation strategies that depend on expertise or experience might lose effectiveness and protract the prototyping process. Automation-oriented approaches, including machine learning and optimization, can be utilized at both design and operation stages to accelerate decision-making without losing energy efficiency for CCHP systems.

As the premise of design and operation for the combined system, information about building energy consumption should be determined initially. Therefore, this thesis first constructs deep learning (DL) models to forecast energy demands for a large-scale dataset. The building types and multiple energy demands are embedded in the DL model for the first time to make it versatile for prediction. The long short-term memory (LSTM) model forecasts 50.7% of the tasks with a coefficient of variation of root mean square error (CVRMSE) lower than 20%. Moreover, 60% of the tasks predicted by LSTM satisfy ASHRAE Guideline 14 with a CVRMSE under 30%.

Thermal conversion systems, including power generation subsystems and waste heat recovery units, play a vital role in the overall performance of CCHP systems. Whereas a wide choice of components, nonlinear characteristics of these components challenge the automation process of system design. Therefore, this thesis second designs a configuration optimization framework consisting of thermodynamic cycle representation, evaluation, and optimizer to accelerate the system design process and maximize thermal efficiency. The framework is the first one to implement graphic knowledge and thermodynamic laws to generate new CO_2 power generation (S-CO_2) system configurations. The framework is then validated by optimizing the S-CO_2 system's configurations under simple and complex component number limitations. The optimized S-CO_2 system reaches 49.8% thermal efficiency. This efficiency is 2.3% higher than the state of the art.

Third, operation strategy with uncertainty for CCHP systems is proposed in this thesis for a hospital with a floor area of 22,422 m^2 at College Park, Maryland. The hospital energy demands are forecasted from the DL model. And the S-CO_2 power subsystem

is implemented in CCHP after optimizing from the configuration optimizer. A stochastic approximation is combined with an autoregression model to extract uncertain energy demands for the hospital. Load-following strategies, stochastic dynamic programming (SDP), and approximation approaches are implemented for CCHP system operation without and with uncertainties. As a case study, the optimization-based operation overperforms the best load-following strategy by 14% of the annual cost. Approximation-based operation strategy highly improves the computational efficiency of SDP. The daily operating cost with uncertain cooling, heating, and electricity demands is about 0.061 $/m^2, and a potential annual cost is about 22.33 $/m^2.

This thesis fills the gap in multiple energy types forecast for multiple building types via DL models, prompts the design automation of S-CO_2 systems by configuration optimization, and accelerates operation optimization of a CCHP system with uncertainty by an approximation approach. In-depth data-driven methods and diversified optimization techniques should be investigated further to boost the system efficiency and advance the automation process of the CCHP system.

OPTIMUM DESIGN AND OPERATION OF COMBINED COOLING HEATING AND POWER SYSTEM WITH UNCERTAINTY

by

Lei Gao

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park, in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2022

Advisory Committee:
Professor Reinhard Radermacher, Chair
Associate Professor Mark Fuge
Professor Steven Gabriel
Professor Jelena Srebric
Professor Peter Sunderland
Research Professor Yunho Hwang
Research Scientist Vikrant Aute

Dedication

To my parents and sister.

Acknowledgements

Accomplishing this thesis and Ph.D. journey would never be possible without the support of my supervisors, Dr. Reinhard Radermacher and Dr. Yunho Hwang. Their high-level perspectives and profound thoughts toward thermal fluid science have consistently inspired me over the years. I am incredibly grateful to Dr. Vikrant Aute, who presents his critical thinking during the weekly group meeting. I want to express my deepest appreciation to my committee members Dr. Mark Fuge, Dr. Steven Gabriel, Dr. Jelena Srebric, and Dr. Peter Sunderland, for their invaluable advice and feedback on my research work. Their unparalleled and insightful knowledge also motivated me to explore interdisciplinary areas.

The almost six years at the Center for Environmental Energy Engineering (CEEE), University of Maryland (UMD) benefited me from daily life to research investigation. I cannot begin to express my thanks to Jan Muehlbauer and Dr. Zhiwei Huang for teaching me numerous experimental skills in the first semester. Their suggestion always perked me up whenever I stuck around a corner. I must also thank Tianyue Qiu for his energetic and cheerful attitude when we stayed late nights at Potomac Building to refresh and construct the wind tunnel. I am grateful to have Dongyu Chen's generous help when I landed at UMD and many thanks to the qualifying exam group with Yiyuan Qiao and Ransisi Huang. Their enthusiasm yielded a friendly and warm study environment. Much appreciation goes to Zhenyuan Mei, Dr. Tao Cao, Dr. Zhenning Li, and other colleagues in CEEE for the opportunity to learn diversified and disruptive innovations in the HVAC area.

Outside of CEEE, I enjoyed working with my friend Tianyuan Liu, who provided numerous insights into my research work. I would also like to extend my gratitude to Jingye Yang and Lingnan Lin for our delightful research collaborations. Notable credits go to the accompany of my roommates Lego, Mia, and Songhua Hu over the pandemic years.

Finally, I am deeply indebted to my parents and sister for their forever understanding, support, and encouragement.

Table of Contents

List of Tables

List of Figures

1 Introduction

Reproduced from the Lawrence Livermore National Lab website (LLNL 2020), Figure 1-1 reveals that the energy consumption of buildings in the U.S., including commercial and residential sectors, is around 38 Quads annually, which accounts for nearly 40% of the total energy consumption in the past two decades. Even though total primary energy consumption in buildings is stable, electricity consumption increases when recognizing the boost of electricity generation efficiency from 30% to 35%. Energy saving is urgent, and a large portion of consumption reduction could dramatically improve total energy efficiency. Therefore, high energy-efficient buildings are critical to the efficiency and sustainability of our society. A combined cooling, heating, and power (CCHP) system offers high primary energy efficiency due to cascade energy utilization and high flexibility because of main grid independence. These merits make CCHP systems capable candidates as energy suppliers for buildings.

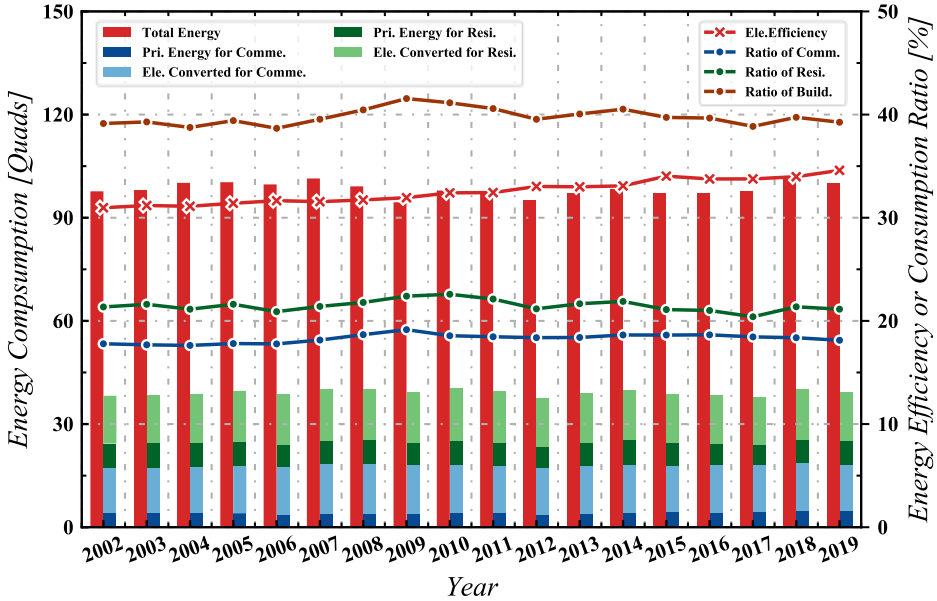

Figure 1-1. Energy consumption of the building sector in the U.S.

1.1 CCHP system

Figure 1-2 shows the significant developments of combined systems in simplified system configurations. Higher energy **E**fficiency, lower **E**conomic cost, and **E**nvironmental impact (3E) are the everlasting goals of developing energy supply systems. The idea of hybrid usage, thermal-driven cooling, heat recovery technique, and renewable energy penetration makes combined systems feasible and more sustainable. Specifically, the combined cooling, heating, and power (CCHP) system is named directly from energy demand types. It is an efficient method to provide electricity and two kinds of thermal energy simultaneously.

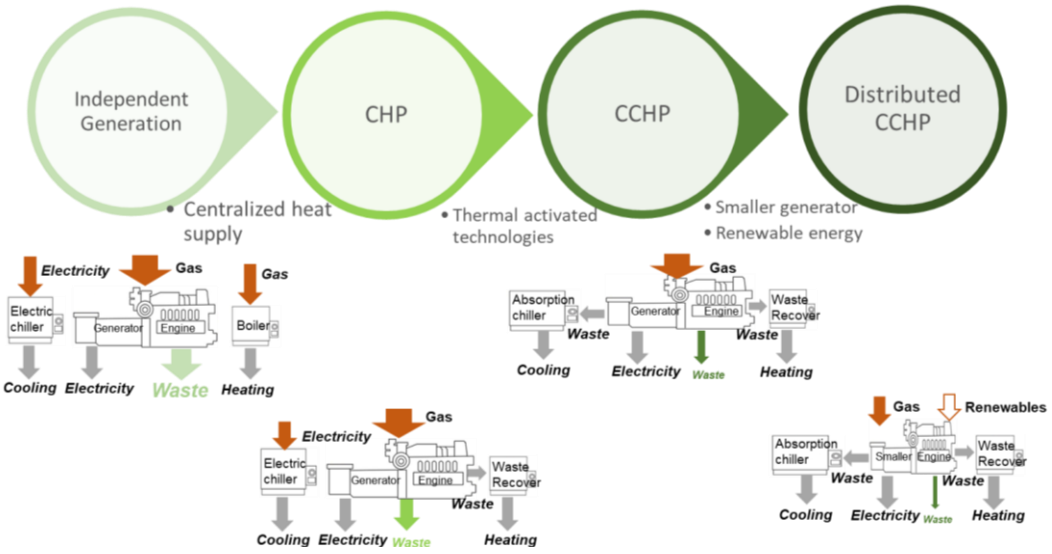

Figure 1-2. Development of combined systems

To distinguish the whole CCHP from its underneath energy conversion technologies, CCHP is called a system, and others are denoted as subsystems in this thesis. The energy flow structure in a specific CCHP system is illustrated in Figure 1-3. The top lines represent energy sources; the rectangular boxes represent energy conversion subsystems; the circular boxes represent energy demands (electricity demand, cooling

demand, and heating demand); the bottom lines represent energy storage subsystems. These four elements form a sustainable energy system and can be enriched by utilizing renewable energy, adopting various conversion technologies, and expanding functions.

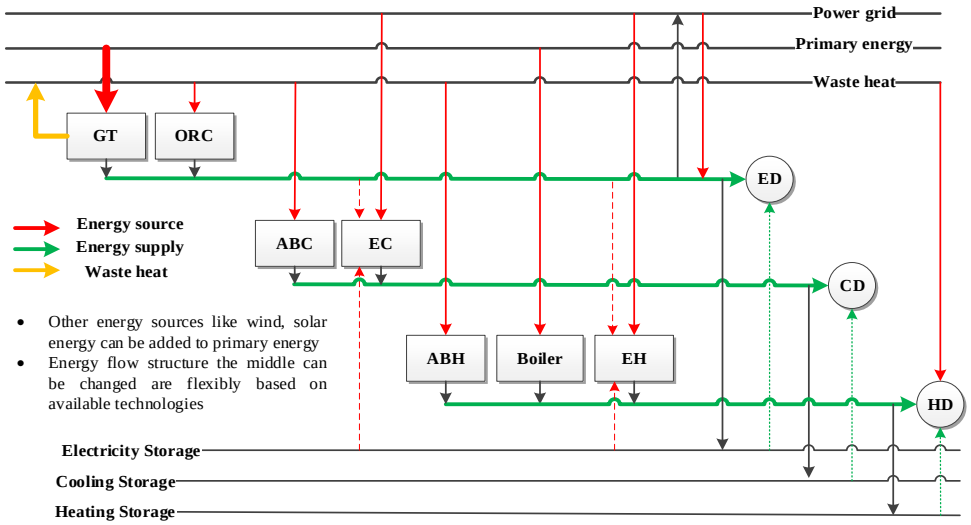

Figure 1-3. Energy flow structure of a CCHP system

In conventional CCHP systems, prime movers (PM) such as a gas turbine (GT) converts primary energy into electricity. The waste heat otherwise discharged is recovered for space heating and cooling demands through absorption cooling (ABC) or absorption heating (ABH). Auxiliary components such as an electric chiller (EC), heat pump (EH), or boiler are added to the system to fulfill extra cooling and heating demands. Overall, a CCHP system can achieve over 80% overall energy efficiency for different primary energy inputs, which is much higher than separated generation systems (Wu and Wang 2006).

There are three stages for a CCHP system to be successfully implemented, as shown in Figure 1-4: demand planning, system design, and system operation (Gu et al. 2014).

The demand planning stage considers how much energy should be provided for a target, like a commercial building. Accurately quantifying energy demand is the key to the planning process and sets the base of the following design and operation stages. This quantification can be realized through a physical model or a data-driven model. It should consider the factors that influence the needs of the building. At the system design stage, multiple subsystems are selected from the candidate pool and combined to satisfy the functionalities of a whole CCHP system. The selection of subsystems should consider criteria like energy source availability, technology maturity, and social impact benefits. The subsystems consist of various components and working mediums. Appropriate mathematical modeling of components characterizes every subsystem and sets the basis for the following stage. At the system operation stage, an operation strategy is determined regarding the strategy's feasibility, the complexity of the designed CCHP system, and the response characteristic of subsystems. The selected operation strategy coordinates subsystems to fulfill energy demand requirements continuously.

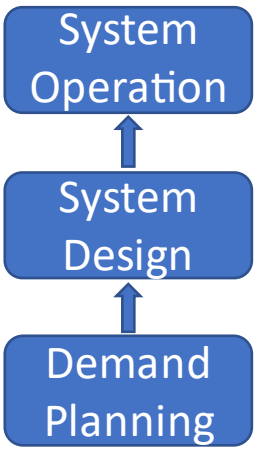

Figure 1-4. Implementation of a CCHP system for buildings in three stages

1.2 Motivation

Regardless of the stages of a CCHP system, decision-makers constantly seek efficient and effective approaches to implementing a CCHP system with 3E targets. However, due to the complexity of CCHP systems and environmental uncertainty, conventional design and operation strategies that depend on expertise or experience might lose effectiveness and protract the prototyping process. Difficulties expand in three stages from planning, design to operation.

First, well-developed physical energy forecast tools require detailed building information and environmental parameters before feeding into the simulation system. These requirements make a comprehensive evaluation of building energy consumption cumbersome. Second, the design of thermal conversion subsystems within a CCHP system consists of both configuration and performance variables design. The configuration design involves a wide choice of components and a strong coupling among components. These complexities and the interdependence between configuration and performance variable designs make researchers choose potential configurations from the candidate pool and concentrate more on performance variables design. The design process is trivial, highly dependent on experience, lose optimality, and thus challenges the automation process of system design. Therefore, it is urgent to develop an optimization framework to consolidate the two elements and yield a more automatic design process. Third, storage subsystems make the conventional operation of a well-defined CCHP powerless. Recursive optimization through dynamic programming introduces curse of dimensionality for system operation. Moreover, energy demands are not always accurate due to inaccurate weather forecasts and

occupant fluctuation. Therefore, it is desirable to implement an efficient and effective approach that can deal with system operation with uncertainty with storage subsystems.

1.3 Literature review

1.3.1 Energy demand planning

The models for building energy forecast can be categorized into physical, data-driven, and hybrid models that combine the previous two (Amasyali and El-Gohary 2018). Physical models are established based on modularized building sectors and heat and mass transfer mechanisms. Nevertheless, these models become incredibly complex when considering the complicated mechanism and the coupling characteristics of every module in buildings. The whole system performance can be evaluated only by simulating or simplifying every subsystem (X. Li and Wen 2014). In this genre, many commercial software packages have been developed to assess the energy consumption of buildings, such as DOE-2, EnergyPlus, and BLAST. However, these well-developed tools require detailed building information and environmental parameters before feeding into the simulation system. Consequently, predicting the energy consumption of any building would require an excessive amount of time, labor resources, and knowledge from experienced experts (Amasyali and El-Gohary 2018). These requirements make a comprehensive evaluation of building energy consumption cumbersome.

More importantly, building energy consumption is highly influenced by uncertain factors from weather prediction, occupant activities, building degradation, etc. (Zeyu Wang and Srinivasan 2017). These factors are hard to determine and evaluate at the

beginning of constructing the physical models. Unlike the physical models, the data-driven models have sometimes been deemed a 'black box'. Therefore, they have no specific physical meaning, such as ML approaches, which only implicitly extract features from data. Compared with physical models, these methods have the advantages of robustness, flexibility, and rapidity when applied to prediction tasks (Sun, Haghighat, and Fung 2020). Concisely, the building energy prediction task is a type of supervised regression in the viewpoint of the ML field. 'Supervised' means the data used for constructing a prediction model has labels, and 'regression' means continuous prediction values. In this genre, any algorithms that are used to deal with supervised regression in the machine learning area have the potential to be shifted to building energy prediction tasks.

The development of ML and data science manifests the opportunities of leveraging data for building energy prediction. Among all the paradigms of ML, neural networks (NNs), support vector regression (SVR), decision tree (DT), and Gaussian process regression (GPR) are the most represented ones (Domingos 2015). These techniques and algorithms have been intensively applied in building energy demand prediction since 2000 (Kalogirou and Bojic 2000; Xu, Wang, and Tang 2019; Dong, Cao, and Lee 2005; Park et al. 2016; Z. Yu et al. 2010; Zhong et al. 2019; Heo and Zavala 2012). Figure 1-5 summarizes the overall research trend of ML-based building energy prediction tasks. Specifically, Figure 1-5a shows the tendency of annually published papers since 2005. 'Traditional ML' methods include DT, GPR, ANN, SVR, and ensemble algorithms, 'others' comprises review work (Amasyali and El-Gohary 2018), statistic methods (Fan et al. 2020), feature engineering (Fan, Sun, et al. 2019), data

augmentation (Fekri, Ghosh, and Grolinger 2020), classification tasks (Westermann et al. 2020), and transfer learning (Fan, Xiao, et al. 2019). The period after 2012 engenders an exponential increase in publications compared with the linear trend of the period before 2012. This notable shift, which corresponds to the major breakthroughs of deep learning (DL), demonstrates the enhancement of ML on energy prediction tasks endued by DL.

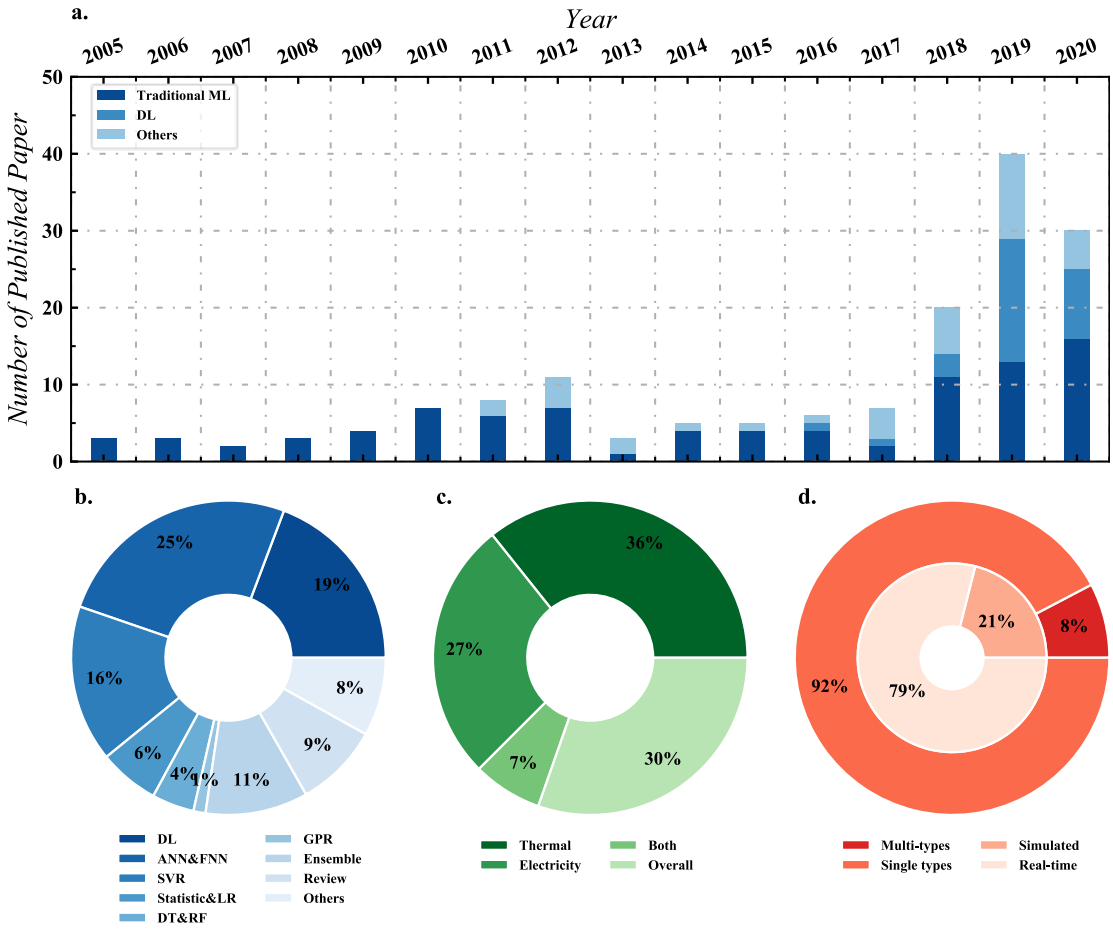

Figure 1-5. Research trend of ML in building energy prediction a. Paper published along with year b. distribution of ML algorithms c. distribution of investigated energy types d. building types and data types

Moreover, DL's capability of dealing with the high-dimensional and large-scale data structure facilitated researchers probing into energy prediction tasks by this algorithm.

Consequently, many investigations supported various DL-based algorithms for building energy prediction (Fan, Sun, et al. 2019). Among all the DL algorithms, RNN (Rahman, Srikumar, and Smith 2018) based model, notably long short-term memory (LSTM) (D. L. Marino, Amarasinghe, and Manic 2016; Kim and Cho 2019; Somu, M r, and Ramamritham 2020), showed its capability in building energy prediction tasks due to temporal characteristics along with the NN layers. DL is superior to other models for building energy prediction in various literature that conducts single energy type prediction (Rahman, Srikumar, and Smith 2018; Zhe Wang, Hong, and Piette 2020) compared with the traditional ML algorithm. To better understand the superiority of the NN-based algorithm in different tasks, Figure 1-5b sums up all 161 papers and plots the distribution of different algorithms. The NN-based algorithms, including ANN and DL, represent 44% of all documents and 50% of research papers. This distribution further manifests NN's fast spread and advantage, mainly DL-based model, over the traditional ML model. Koschwitz et al. adopted historical data from a non-residential district in Germany for training ML models to predict monthly thermal loads (Koschwitz, Frisch, and van Treeck 2018). The testing results showed that recurrent NNs yielded higher accuracy than SVM-based models. Walker et al. analyzed seven machine learning algorithms for hourly electricity demand prediction (Walker et al. 2020). The result showed that the regression tree and artificial neural network (ANN) model demonstrated higher performances than SVM and the fine-Gaussian algorithm. Wang et al. differentiated other algorithms with NN-based models as shallow machine learning (Zhe Wang, Hong, and Piette 2020). And the authors found DL-based LSTM was best for short-term prediction among all seven shallow and two deep algorithms.

These papers all validated the ability of the NN-based model, mostly DL-based model, to outperform traditional ML in building energy prediction tasks.

The primary concern of single energy type prediction aimed to explore the capability of ML for predicting overall energy consumption from thermal or electricity (Rahman, Srikumar, and Smith 2018; Zhe Wang, Hong, and Piette 2020). Simultaneously, the detailed consumption information by different sectors of buildings is compressed in this single task. In contrast, such multiple energy types prediction is significant for the operation of subsystems in buildings. The multiple energy types prediction is limited, as shown in Figure 1-5c. Here, 'overall' energy consumption represents the task targets predicting the gross sum of thermal energy and electricity. 'Both' indicate the task targets predicting energy consumption from thermal and electricity. It is observed that most of the prediction targets are a single energy type. With the capability of overcoming the limitation of physical and hybrid prediction models, a natural question is whether the ML-based regression models can accurately predict multiple energy types of building for multiple building types. On the other hand, the last pie plot in Figure 1-5 shows whether an ML algorithm is used for a single type or multiple types of building prediction. Different building types involve various energy consumption ranges and distinctive energy demand attributes. To further expand the robustness and increase the versatility of the DL model, it is also worthwhile to investigate the performance of DL-based models for multiple energy types on multiple types of buildings.

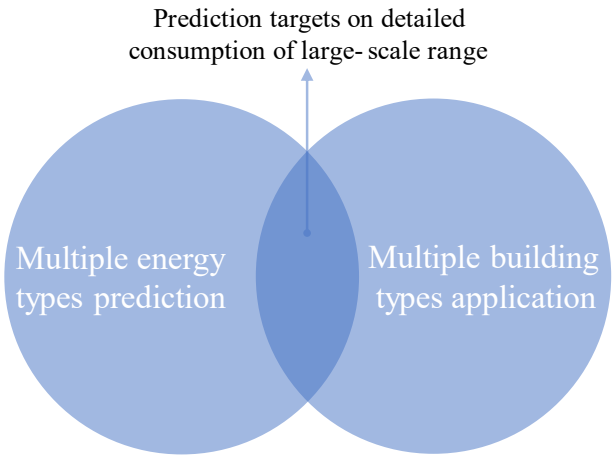

Figure 1-6. Multiple-energy-types prediction on multiple-building-types

Energy prediction model for multiple energy types and building types can be categorized in Figure 1-6. It is significant for achieving optimum design, operation, and large-scale sustainability for energy supply systems. Therefore, applications of these models in multiple energy types prediction on multiple building types are summarized. The gaps in related research works are discussed below.

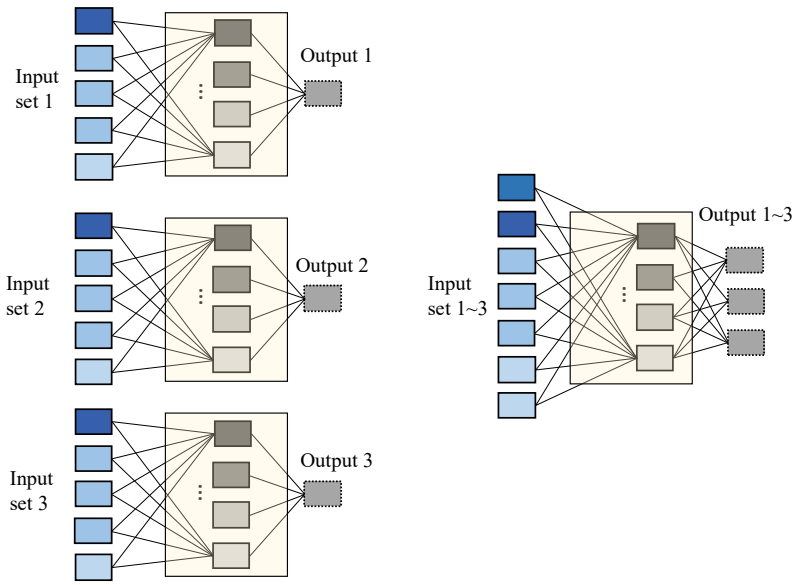

Figure 1-7. Individual model vs. single model for multiple energy types

There are two streams to predict multiple energy types for buildings, which can be viewed in Figure 1-7. The first stream is to construct numerous individual models to independently predict the corresponding energy type. The second stream is building a single model to predict multiple energy types simultaneously. It should be noticed that the ML 'black box' is not necessarily a neural network (NN) model. The other ML models can also be used. Considering that energy consumption from different sectors depends on each other (Amasyali and El-Gohary 2018). For instance, the electricity for the plugin electric vehicle interacts with space cooling or heating load (Yun et al. 2012; Chae et al. 2016). Moreover, the physical model constructed from well-established software EnergyPlus (Crawley et al. 2001) shows that factors like occupants residing in the building impact electric light, water heater, cooling, and heating load; meteorology information affects cooling and heating energy consumption at the same time.

Compared with conventional ML, parameter sharing in DL naturally helps with increasing generalization and improves the performance of similar tasks (Ruder 2017; Y. Zhang and Yang 2018). Moreover, the feature of the building type can be easily embedded into the model at the appropriate layer of NN. For a convolution neural network (CNN) or recurrent neural network (RNN) model, the characteristic of parameter sharing in neural network layers balances the performance on multiple outputs. At the same time, the multiple independent models cannot obtain this shared information. In addition, it is proved in the ML area that a single model for multiple dependent energy types output increases robustness or generalization of prediction compared with an independent model (Lu et al. 2016). And the feature of building type

can be easily embedded into the model at the appropriate layer of NN. Moreover, for complex prediction tasks, like multiple energy types prediction, a single model saves model construction time (Y. Zhang and Yang 2018) compared with multiple independent models. There are ten energy types and 16 types of commercial buildings for the dataset in this thesis. The number of individual models would be 160. It would be even larger when considering the locations of different cities.

At the same time, the patterns of occupant or meteorology information might not be easily extracted or obtained as the input features of the model (Zhao and Magoulès 2012). A reasonable way to predict multiple energy types is to construct a single model where the dependent historical multiple energy types can be explored as additional information instead of multiple individual models. Seyedzadeh et al. conducted a comprehensive study to compare the most popular ML models for predicting simulated cooling and heating load (Seyedzadeh et al. 2019). The authors continued to propose specific parameter tuning, optimization, and input feature ranking methods for these ML models and addressed the energy retrofit problem (Seyedzadeh et al. 2020). These research work together to provide a valuable reference for ML model construction and parameter tuning. Therefore, a single model should be constructed to make predictions with shared representation in DL to generate enriched bond information within the different energy types.

One mainstream addresses numerous building type problems by emphasizing the properties of the buildings. Chou and Bui constructed an ensembled model containing ANN and SVR to predict thermal load (Chou and Bui 2014). This model demonstrated

efficiency, effectiveness, and accuracy at the planning stage. Whereas every property of buildings containing wall area, height, compactness, etc., was used as input features in the model. The broad and gross selection of features made the model cumbersome and had few differences from the physical model. Robinson et al. (Robinson et al. 2017) further decreased the feature number to five and adopted a gradient boosting regression model to predict national data from the Commercial Buildings Energy Consumption Survey (Yalcintas and Ozturk 2007). However, the granularity of data is one year. Therefore, this treatment is profitable for energy suppliers but invalid for operation guidance on the demand side of buildings.

The other branch investigates this task by categorizing building types. Culaba et al. developed a k-mean algorithm for the clustering building type and SVR for predicting 30 mixed-used buildings (Culaba et al. 2020). The clustering combined with the predicting method can distinguish the consumption behavior of different building types. However, there is only gross energy consumption considered. Hawkins et al. categorized campus building types by classroom, lecture hall, sixth form college, and university (Hawkins et al. 2012). Activity-based benchmarks and building internal environment types were chosen as input features of the ANN model. The authors analyzed the importance of parameters in the ANN model and confronted a low statistical significance of these parameters due to the high degree of noise in training data. Nonetheless, the prediction results of their work indicated the potential of the NN-based model for analyzing building energy even though significant improvement is needed for this model. In terms of the NN model, the feature of building type can be easily embedded into the model at the appropriate layer of NN.

1.3.2 System design optimization

System design has always been widely investigated in engineering areas, such as topology generation, material design, 2D shape synthesis (Regenwetter, Nobari, and Ahmed 2022), and circuit synthesis (Guo, Herber, and Allison 2019). System design concerns two significant aspects called configuration variable and performance variable. For thermodynamic system design, system configuration contains information on specific components (like compressor, heat exchanger, turbine, and valve) and their connection. And the configuration of a thermodynamic system is a heterogeneous graph when considering the multiple types of components. The performance variables include capacity, pressure ratio, pinch point temperature, evaporating or condensing temperature, efficiency, and other factors influencing the system performance. A system must go through an iterative process between configurations and performance variables to obtain a satisfying design. An automatic way to generate this satisfying design is through an optimization process viewed in Figure 1-8.

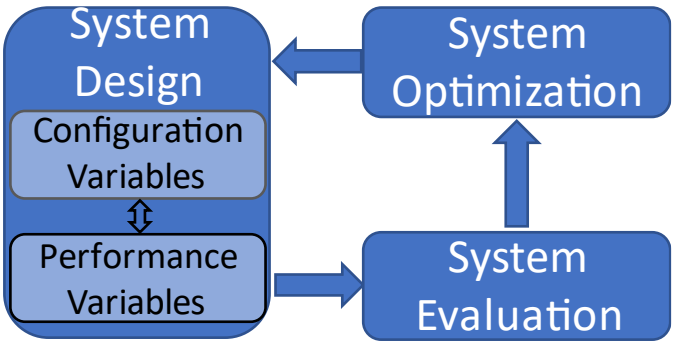

Figure 1-8. Overview of system design optimization

The system design, evaluation, and optimization are responsible for defining a system in an appropriate data structure, solving a system with interested metrics, and providing iteration direction for system design. The performance variables, like pressure ratio, efficiency, effectiveness, approach temperature, etc., is always continuous values in the thermodynamic system design area, whereas the configuration is unstructured data that a graph can represent. The optimization for system design becomes complex due to the mix of different types of variables. Since the configuration cannot be separated from the performance variables when conducting design optimization, researchers can optimize the performance variables for a specific configuration and then tune the configuration. This bi-level optimization process takes a long time to converge. Therefore, the conventional approach for thermodynamic design optimization is first selecting the candidate configurations based on the researchers' expertise. And optimization is then conducted with a focus on performance variables. Therefore, design optimization that includes the configuration aspect for thermodynamic systems is still under investigation compared to other engineering design optimization, like 2D shape synthesis, circuit synthesis, and chemical processing. The following subsection summarizes the research works targeted at configuration optimization in the area of the thermal energy system.

Overall, there are three approaches of configuration optimization for energy conversion systems. The first approach is a task-oriented method. The objective of the optimization is to reach specific temperatures with minimum economic costs. Part of the system remains optimized in this genre, such as the heat exchanger network (HEN) (Yee and Grossmann 1990; Lazzaretto and Segato 2001). This method is always

adopted in chemical processing applications (Toimil and Gómez 2017; Klemeš et al. 2020). The flowsheet is sometimes rigid, and the whole network's functionality completes part of the entire process. In thermal engineering applications, the target is to obtain the optimum temperature distribution at a system level (Jiang et al. 2018) or a single heat exchanger level (Z. Li, Aute, and Ling 2019). Pinch point (Sanaye and Niroomand 2007) or discrete parameter (Z. Li, Aute, and Ling 2019) methods are applied to simulate the network. The complexity comes from the network structure, and the functionality of every vertex that represents the thermodynamic component is the same -- heat transfer. A homogeneous configuration is determined when each component has same properties (material, functionality, representation, etc.). Whereas a heterogeneous configuration is defined as those where two or more types of components exist in the system. In other words, the targeted configuration is a homogeneous one for heat exchanger design. Therefore, it is not applicable for design optimization of the whole thermal energy conversion system, where heterogeneous configuration is required to represent the thermodynamic cycle.

The second approach adopts superstructure, which first emerged in the process synthesis area with the concept of containing various advanced system configurations (Yeomans and Grossmann 1999). In other words, the optimization variables at the configuration level are on-off signals of the connecting tubes so that only part of the superstructure works. With this assumption, components used or not in the system can be described using binary variables. The problem can be attributed to the nonlinear component performance equations as mixed-integer nonlinear programming (MINLP) problems. A matrix table is used to represent candidate configurations. Grekas and

Frangopoulos adopted an automatic synthesis process for the energy system based on graph theory, and the superstructure of a combined cycle was examined the effectiveness of the approach (Grekas and Frangopoulos 2007). Elsido et al. proposed a superstructure-based Rankine cycle that allowed the whole system to perform a techno-economic optimization (Elsido, Mian, and Martelli 2017). Lee et al. designed a superstructure-based optimization study utilizing LNG cryogenic exergy (Lee et al. 2017). This structure contains 1,024 possible process alternatives. The best process generates 40% more power than a simple organic Rankine cycle (ORC) system with a stochastic optimizer. Yu et al. proposed integrating ORC with HENs considering a superstructure with optional turbine bleeding and regeneration (H. Yu et al. 2017). The optimized system improves the best configuration design by 13%. Bao et al. conducted a simultaneous optimization among nine configurations for a three-stage condensation ORC system (Bao, Zhang, Lin, et al. 2018). Bao et al. proposed a simultaneous approach to achieve the optimal components and compositions of the zeotropic mixture at the same time (Bao, Zhang, Yuan, et al. 2018). The result shows the system with zeotropic mixture has better performance than that of systems with pure fluid. Cao et al. designed a superstructure based CCHP system for minimizing the operation cost under transient conditions (Cao, Hwang, and Radermacher 2017). Alvarado et al. simultaneously optimized the selection and operation of technologies for distributed energy systems in buildings (Cedillos Alvarado et al. 2016). It allowed decision-makers to select optimal technology from a limited portfolio of different technologies and defined an operational schedule that minimized the whole life costing and carbon emission. However, all the superstructure-based approaches are predefined based on

researchers' expertise, which might exclude the optimum configuration initially. Moreover, the problem scales up with a large number of candidates, making solving a mixed-integer nonlinear optimization problem time-consuming.

In the last approach, the heuristic search or superstructure-free method depends on less experience to generate new configurations. This method differs from the superstructure method for generating new configurations by a deterministic algorithm. General rules are implemented for generating a new possible configuration (e.g., it must consist of a specific component or circle). Therefore, the method is stochastic instead of deterministic. However, instead of a mathematical programming approach to discover configurations from the integrated search space, an heuristic or metaheuristic algorithm is adopted to generate candidate configurations, then evaluated by an optimization algorithm. Wright and Zhang designed a genetic algorithm to optimize the configuration of the HVAC system (J. Wright et al. 2008) and validated this approach through experiments (J. Wright and Zhang 2008). The algorithm had an 81% probability of finding a feasible system design when the component set was fixed as a boundary condition on the search. However, the core part is designing a homogeneous graph representing a ventilation network like HEN mentioned before. Only flow rate and energy mix or split are needed to be considered, which is insufficient for whole system optimization. Similarly, Zhang et al. adopted an artificial bee colony algorithm for HVAC optimization problems and validated it with two HVAC example problems (Xin Zhang, Fong, and Yuen 2013). Voll et al. proposed a superstructure-free framework for a distributed energy supply system by evolutionary algorithms (EA) (Voll 2014). Subsystems were hierarchized and categorized based on functionality, and

specific rules were designed to implement EA mutation. The hierarchy-supported approach can automatically identify complex solutions such as cogeneration and trigeneration. Wang et al. further developed this approach by combing an energy conversion hierarchy, which allows for generic replacement rules (L. Wang et al. 2014). Toffolo upgraded from HEATSEP and developed a superstructure-free codification of the Rankine cycle (Andrea Toffolo 2014). This codification assumed that all sub-cycles in configurations are constructed from a four-component primary ORC system. From this assumption and cascade mechanism, the newly generated configuration never violates the connection constraint as well as the thermodynamic constraint without a priori specification. The authors (A. Toffolo, Rech, and Lazzaretto 2018) further applied this method for a two-pressure level Rankine cycle optimization and validated the effectiveness of this approach by different working fluids and temperatures of the heat source. However, the basic cycle only considered a four-component structure, which limited the configuration potential. For instance, the method would be powerless if a heat exchanger, compressor, or turbine is connected continuously in a basic cycle. It is noteworthy that the optimization work in this category heavily relies on experience or expertise since generating possible configurations obeys the rule settled by researchers. Therefore, an approach that depends on less expertise is needed to optimize configuration of thermal energy system.

1.3.3 System operation optimization

The load-following methods are the most straightforward strategy for operating a CCHP system, as illustrated in Figure 1-9. The performance of PM is shown by the solid line, which represents the relationship between generated electricity and

recoverable heat from the exhaust of PM (P.J. Mago and Chamra 2009). Any point A or B represents the energy demand at any time. The cooling load is also transferred to the heating load through thermally activated technologies such as ABC. Let's take point A as an example. Regulating PM to point A′ means that the system operates under the following thermal load (FTL), while point A″ means the following electricity load (FEL). It is the same for point B. Overall, the load-following method matches the energy demand and supply strategy. It is observed that energy would be wasted no matter which load-following method is used. Thus, some variants from the basic load-following method like the hybrid method (P.J. Mago, Chamra, and Ramsay 2010; Fang, Wang, and Shi 2012; Pedro J. Mago and Hueffed 2010) and the following minimum distance method (C. Y. Zheng, Wu, and Zhai 2014) are used to reduce energy wasted. Afzali and Mahalec (Afzali and Mahalec 2018) developed a novel performance curve for determining the optimal control strategy for a CCHP system based on the load-following method. Energy price, fuel consumption, and CO_2 emission rates were all considered in the novel curve. The case study shows that this methodology leads to the best operation result compared to other load-following strategies based on 3E objectives. However, the load-following method and its variants are predefined and not optimal operation strategies. They can be only used for capacity optimization from the design point of view. In addition, subsystems are oversimplified sometimes, and storage is never investigated in this method. Therefore, it is mainly suitable for simple systems and static optimization models.

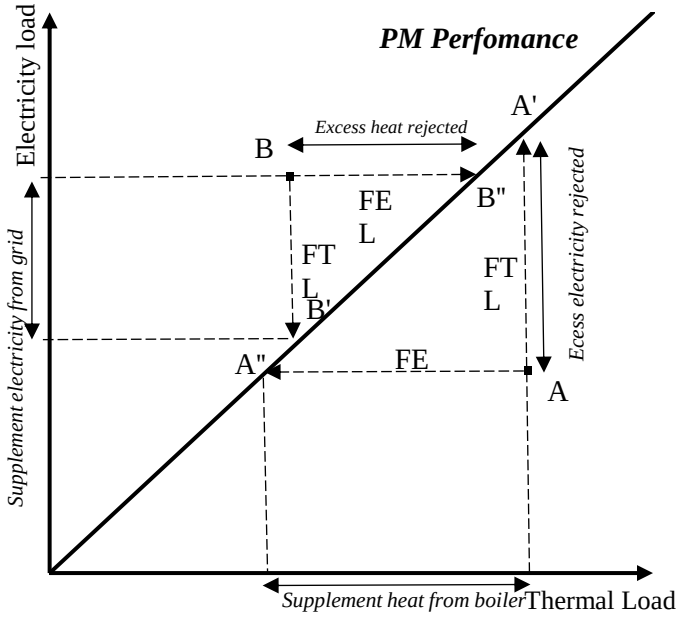

Figure 1-9. Load-following method

The storage subsystem effectively mitigates fluctuation in energy demands and peak shaving characteristics of CCHP systems. Wang and Ma (S. Wang and Ma 2008) stated that the optimization related to storage subsystems was a dynamic optimization. A trajectory of decisions should be determined in the optimization process rather than a single-point optimization. In other words, the optimization with storage requires significantly increased computation resources compared with a system without a storage subsystem.

Zheng et al. considered heating and cooling storage subsystems and built a mixed-integer nonlinear programming (MINLP) model in GAMS to deal with the dynamic optimization problem (X. Zheng et al. 2018). The author solved the optimal size and operation strategy through the Lindo optimizer targeting minimum cost and CO2 emissions. Zheng et al. proposed a novel operation strategy for the thermal storage

strategy in a CCHP system (C. Y. Zheng et al. 2017). This strategy determined the operational state of the power generation unit based on the energy demand and the state of the storage subsystem. This method was a complex variant of the following demand method by counting the storage energy into the PM. The simulation results show that charging or discharging as much as possible stored thermal energy would improve system performance by up to 20% as compared with the system without thermal storage. Hajabdollahi compared four different storage subsystem configurations when optimizing the capacity and operation strategies of a CCHP system (Hajabdollahi 2015). The optimal case shows that the total annual profit would increase most for adopting cooling and heating storage devices. Gu et al. investigated the storage subsystem in the system optimization with load uncertainty (Gu et al. 2015). The authors found the storage subsystem acted better under more considerable uncertainty of thermal demand. In addition to the traditional sensible thermal storage subsystem (Liu et al. 2015), some new storage technologies such as phase change material and compressed air have been applied to the CCHP. Jabari et al. used MINLP to optimize the energy and exergy efficiency of an adiabatically compressed air energy storage-based CCHP system (Jabari, Nojavan, and Mohammadi Ivatloo 2016). They resulted in 21.8% and 22.4% lower operational costs under cooling and heating modes. Latent heat storage is another choice. Ruan et al. adopted an ice storage subsystem in building a CCHP system. The system had similar energy efficiency but lower operation cost by using a simple linear programming optimization model (Ruan et al. 2016).

The system operated under real-time data should consider discrepancies between practical and predicted demands. However, the above operation methods are based on

predicted or default energy demand, energy resources, and cost incentives. Therefore, they cannot identify optimal real-time management with uncertain conditions. More specifically, uncertainty from energy demand prediction, energy market price, and activity schedule should be considered. Stochastic solutions are usually preferred over deterministic solutions since the former provides a reliable operation strategy in real-world applications (C. Marino et al. 2018).

Farmani formulated a deterministic and stochastic operation optimization problem (Farmani et al. 2018). The Monte Carlo method was used to produce many scenarios to simulate uncertainties, including weather conditions and forecast prices. Results show that the proposed smart operation strategy could significantly reduce the operating cost by considering the correlation between historical data. However, energy storage is not used in the CCHP system, which dilutes the effectiveness of this approach. Zhou et al. investigated the robust operation optimization strategy for an integrated community energy system through the mixed-integer linear programming (MILP) model (Zhou et al. 2018). The uncertainty of renewable energy, including wind and solar, and the uncertainty of the price market were considered in the robust optimization. The confidence intervals of uncertain parameters were predicted via the Gaussian process method. Marino et al. investigated the CCHP system operation strategy under the uncertainty of electricity (C. Marino et al. 2018). The uncertain demand was assumed to follow a normal distribution. Two-stage optimization was implemented, in which the uncertainty occurred in the second stage. A hybrid sample average approximation algorithm was used with an enhanced benders decomposition algorithm. Ji et al. applied a fuzzy-risk-explicit interval parameter programming

method to solve the system operation problem under random energy price (Ji et al. 2018). The method can be expanded to tackle fuzzy and interval uncertainties in terms of various cost coefficients, forecasted demands, decision makers' risk attitudes, and other microgrid system management uncertainties. Kuang et al. implemented a stochastic dynamic solution to investigate the performance of a CCHP system with storage with uncertain renewable resources and demands (Kuang, Zhang, and Sun 2019). The results indicate that the CCHP system with random demands and renewable energy shows operation cost reduction by 1.66% compared with the deterministic optimization.

Most of these research works evaluated energy demands, energy resources, and cost incentives independently. The uncertainties are extracted in the form of confident intervals. However, multiple energy types in buildings are interdependent. And there is a lack of investigations on uncertainties of all kinds of demands. Moreover, the computational cost of implementing stochastic dynamic programming increases exponentially with the numbers of storage and discretization. Therefore, it is necessary to implement an efficient and effective approach for investigating the uncertainties by considering all types of energy demands and multiple energy storages.

1.4 Review summary and gaps

Compared with a single energy type on one type of building, there is insufficient research for predicting multiple energy types on an hourly basis for multiple types of buildings in a single ML model (Amasyali and El-Gohary 2018; Zhao and Magoulès 2012; Sun, Haghighat, and Fung 2020). All prediction models are built specifically for

one kind of building or long-term energy consumption, which might not provide operational guidance for buildings or clusters. Second, few efforts have been devoted to multiple energy types predicting, needless to say, the model performance for multiple energy types and multiple types of buildings. Table 1-1 lists the major research works that implemented DL for building energy forecast. These research works focused on electricity or cooling demand forecast for only one types of buildings. Even though an independent model can be built by traditional ML for energy consumption from different sectors and then combine these models to predict multiple energy types, it would be cumbersome and lose interrelationship among different energy types. Meanwhile, sparsity characteristics in the dimension of the building type would be troublesome when feeding into traditional ML models.

Table 1-1. Major research works by applying DL for building energy forecast

Authors	Granularity	Input feature	Output	Dataset scale*	Model
Kim, 2019	1 minute	electricity	electricity	2 m	CNN LSTM
Marino, 2016	1 minute	electricity	electricity	2 m	LSTM
Herrero, 2017	1 minute	cooling	cooling	0.35 m	LSTM
Rahman, 2018	1 hour	electricity	electricity	0.3 m	LSTM
Cai, 2019	1 hour	electricity	electricity	24 k	CNN
Wang, 2019	15 minutes	electricity	electricity	46k	LSTM
Wen, 2020	1 hour	electricity	electricity	10 k	GRU
This thesis	1 hour	electricity, cooling, heating, building type	electricity, cooling, heating	130 m	CNN LSTM

* The dataset is in million (m) or thousand (k) scale.

The design optimization concerns two aspects, including configuration specification and performance variables design. The configuration design aspect required more investigation compared with the performance variables design aspect. The new configuration generating process can be categorized into three approaches by the prerequisite knowledge for developing candidate configurations, as shown in Table 1-2. The configuration variables for task-oriented and superstructure methods are discrete and explicitly expressed in the optimization function. Therefore, mathematic optimization-based nonlinear solvers can be used to tackle design problem in these two approaches. However, it might take longer to find an optimum and might screen out the optimum configuration at the beginning. On the other hand, the configuration variables are implicitly programmed in the configuration generation process for the heuristic/metaheuristic-based method. Therefore, the optimization algorithm does not need to deal with discrete variables but focuses more on designing the configuration generation process.

Table 1-2. Four approaches for configuration optimization of energy systems

Authors	Approach	Prerequisite knowledge		Variable type	Model
		Graph theory	Thermodynamics		
Wright, 2008	Task-oriented	Yes	Expertise	Continuous and explicit discrete	MINLP
Elsido, 2017	Superstructure	No	Expertise, integrated cycles	Continuous and explicit discrete	MINLP
Toffolo, 2014	Heuristic	No	Expertise, basic cycle	Continuous and implicit discrete	NLP
This thesis	Metaheuristic	Yes	1st and 2nd laws	Continuous and implicit discrete	NLP

Table 1-3 lists major operation optimization works for CCHP systems. Load-following and predefined methods are the basic operation strategies. They cannot guarantee an

optimum operation even though they are widely used for CCHP system operation due to their ease of implementation. Dynamic optimization is another feature of CCHP operation optimization with a storage subsystem. Besides, the mismatch between prediction and actual need must be considered for optimum operation. When studying operation optimization, uncertain energy demands should be evaluated. However, research work seldom considered any of these uncertainty factors.

Table 1-3. Major operation optimization works for CCHP systems

Authors	Storage	Uncertainty	Method	Objective	Model
Mago, 2009	No	No	Load-following	Cost	LP
Wang, 2010	No	No	Load-following	Cost, Emission, Efficiency	LP
Caliano, 2017	Yes	No	Predefined strategy	Cost	LP
Zheng, 2014	Yes	No	Minimum distance	Cost	LP
Cho, 2010	Yes	No	Energy dispatch	Cost, Emission, Efficiency	NLP
Hu, 2014	No	Yes	Probability constrained	Cost, Emission, Efficiency	LP
Hans, 2015	Yes	Yes	Model predictive control	Cost	MIP
Kang, 2018	No	Yes	Robust optimization	Life-cycle performance	NLP
Aghdam, 2020	Yes	Yes	Chance constrained	Emission	LP
This thesis	Yes	Yes	Stochastic dynamic programming	Cost	MILP

Based on the summary of literature review, the main gaps are summarized as follows:

- Successfully integrating a versatile ML model at the demand planning stage can alleviate the dependence on expertise and accelerate the implementation process of CCHP. This multiple energy types of prediction on multiple buildings by DL in this thesis would fill the gap of detailed guidance for the energy supply system of buildings.

- It is necessary to develop an optimization framework to consolidate the two elements, including configuration and performance variables, to yield a more automatic design process. This thesis uses least rules to generate system configuration and implement a metaheuristic approach to optimize system thermal efficiency.

- There is a need to thoroughly study the optimum operation of CCHP systems with uncertainties, from uncertain demands sampling to efficient operation strategy.

1.5 Research scope

The overall objective of this thesis is to implement CCHP system to further improve the energy efficiency and operational cost for commercial buildings. To efficiently and effectively implement a CCHP system with 3E targets, multiple automation-oriented approaches, including ML and optimization, are utilized to accelerate decision-making at different stages. Specifically, 1) at demand planning stage in Chapter 2, a versatile DL model was built to investigate the performance of detailed energy consumption on a large scale of buildings simultaneously; 2) at design optimization stage in Chapter 3, this thesis developed a graphic and thermodynamic knowledge-based design

optimization framework is to optimize the configuration of power systems; 3) at operation stage in Chapter 4, this thesis proposed operation optimization approach for CCHP systems with considering uncertainty from multiple energy demands.

2 Energy Demand Planning

Different ML models for building energy consumption, focusing on DL and traditional ML, are summarized in the review section. As energy consumptions associated with different sectors of buildings are satisfied by different energy conversion systems, such as distributed energy systems or combined systems (Gao, Hwang, and Cao 2019), accurately characterizing and forecasting multiple energy types of buildings are crucial and essential in optimally sizing and operating such systems (Rathor and Saxena 2020). The more detailed information obtained from different building sectors, the more energy conversion systems would gain by preparation. In other words, every section in the buildings has a specific objective to achieve rather than targeting a gross number that sums up all energy demands. Therefore, this multiple energy prediction is beneficial for the detailed and precise management of buildings. Meanwhile, single building design and planning are insufficient (Bourdeau et al. 2019). Overall, creating a model to predict a cluster of buildings with multiple energy types is meaningful.

In this thesis, DL models are proposed to investigate the performance of multiple energy types on multiple building types simultaneously. The DL models are trained and tested on a large scale of datasets simulated by EnergyPlus over 16 DOE building types in 936 cities in the US. And each dataset contains ten types of energy with hourly consumption in a typical year.

2.1 Original data analysis and visualization

To investigate the performance of the DL model on multiple energy types for multiple building types. The dataset should vary in both dimensions of energy type and building

type. The dataset used for making large-scale predictions comes from Open Energy Information Database (OpenEI) (OpenEI 2020). In this dataset, 16 different types of commercial buildings were simulated in 936 cities in the United States through EnergyPlus. Figure 2-1 shows the distribution of selected cities on the map. Therefore, there is a total of 14,976 building files. The building geometry was based on DOE building archetypes, and the building location and thermal settings can be found in reference (Hendron and Engebrecht 2010). The granularity is one hour for each energy consumption with a total length of a whole year period (8,760 hours). In detail, the simulated energy consumption includes total gas, total electricity, and other sectional energy consumption. The total data points are around 130 million considering 8,760 hours in each building files.

Table 2-1. Prediction tasks

Total energy	Sectional electricity	Sectional gas
Electricity, Gas	Fan, Electric cooling, Electric heating, interior light, other interior equipment	Gas heating, water heater, other interior equipment

The sectional consumption includes eight energy demands, such as fan, light, cooling, heating, water heater, and other interior equipment, shown in Table 2-1. These ten types of energy consumption are the target values of our multiple energy prediction tasks. In addition, the corresponding weather datasets for each location are fetched from the TMY3 dataset (Wilcox and Marion 2008).

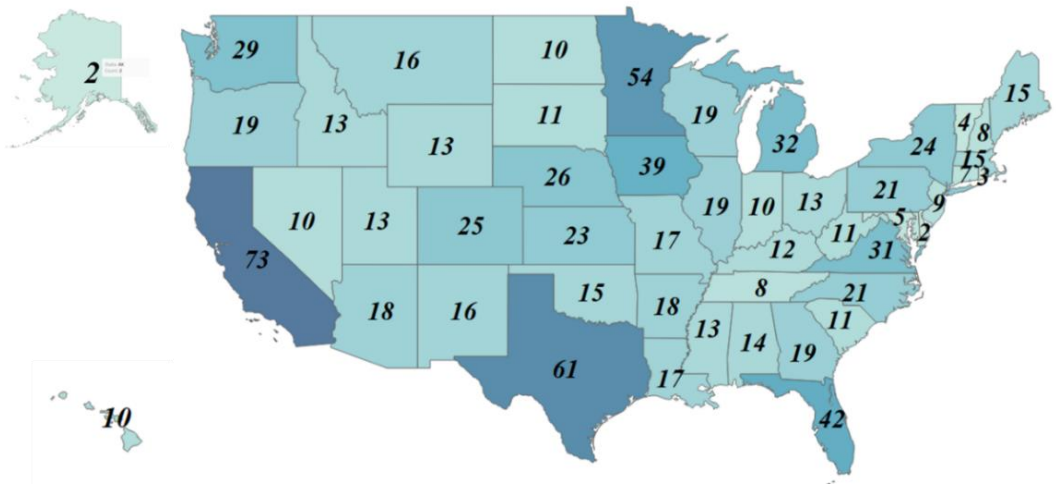

Figure 2-1. Distribution of selected cities in the US

As mentioned earlier, different building types involve a various range of energy consumption and distinctive energy demand attributes. The original 130 million data points were grouped by building and energy types. To better view and compare the influence of building types on total energy consumption and consumption from different sectors, these 16 types of buildings were categorized into three groups based on the maximum consumption levels shown in Table 2-2. Given this category, Figure 2-2 plots the energy consumption distributions of different building types and various prediction tasks statistically. Besides the diversity of energy consumption from different building types, it is easy to observe that gas and electricity consumption distributions are quite different. The total gas consumption has a broader range and higher variance. There are more outliers (colored star marks above maximum value represented by horizontal line) for gas-related consumption than electricity. Moreover, some of the sectional consumptions remain almost zero, such as 1) gas consumption of interior equipment and water heater in all building types except full restaurants, apartments, quick restaurants, small and large hotels, and 2) electricity consumption of

heaters in full restaurants, quick restaurant, and small office. These inconsistent and non-equivalent characteristics in different building types and different sectional consumptions make accurately predicting multiple energy types hard.

Table 2-2. Building category

Group number	I	II	III
Total gas consumption range x [kWh] Total electricity consumption range y [kWh]	x<200 or y<100	200<x<1000 and 100<y<500	1000<x and 500<y
Building type	Full restaurant Apartment Quick restaurant Small hotel Small office	Middle office Outpatient Retail store Strip Mall Warehouse	Hospital Large hotel Large office Primary school Secondary school Supermarket

As the cornerstone of the DL-based method, historical consumption data plays a vital role in constructing the model and how well the model would perform. As a result, feature selection on the input side should be made carefully. As summarized in (Sun, Haghighat, and Fung 2020), the input feature includes meteorological information, historical energy consumption data, time index, and building category in Table 2-3. Primarily, determined and predictable information is used as the input feature. In this study, the selected features for the DL models include meteorology information, historical consumption, and timestamps, such as months, dates, and hours. Specifically, meteorology information includes solar radiation in both normal and horizontal directions, cloud coverage and opacity, temperature, humidity, pressure, and wind. Overall, together with building type, 22 features were used as the input of the DL models. A correlation analysis is conducted within all features except timestamps to investigate the relationship between these features. The meteorology and energy

consumption information was extracted from a small hotel in Baltimore, Maryland, in the U.S. Figure 2-3 shows the covariance of different features. It is seen that the same type of energy (gas or electricity) has a strong relationship with each other. Some gas or electricity consumption has a more significant covariance than the influence of meteorology. A fixed time step of 24 hours was used to compare these two DL models, and this parameter varied from 12 to 48 hours in the parametric study Section 2.7.

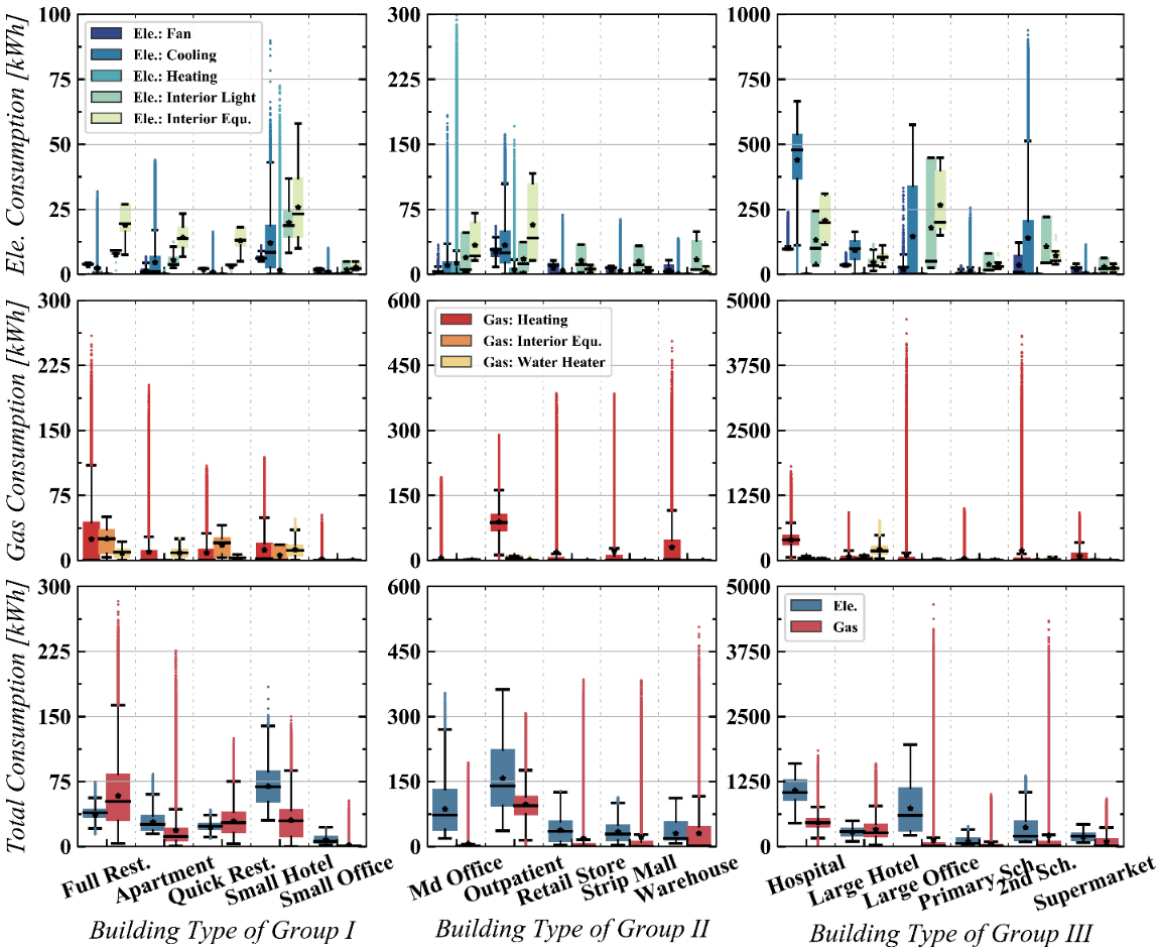

Figure 2-2. Energy consumption distribution for different buildings in Baltimore Washington Airport

Table 2-3.Model input data category

Meteorological information	Historical data	Time index	Building category
Dry bulb temperature Dew point temperature Station pressure Wind speed Extraterrestrial radiation on a horizontal surface (ETR) Extraterrestrial radiation normal to the sun (ETRN) Total sky cover (Tot Cld)	Total electricity Electricity for fan Electricity for cooling demand Electricity for heating demand Electricity for lighting Electricity for auxiliary equipment Total gas Gas for heating Gas for auxiliary equipment	Month Date Hour	One-hot codify from 1 to 16

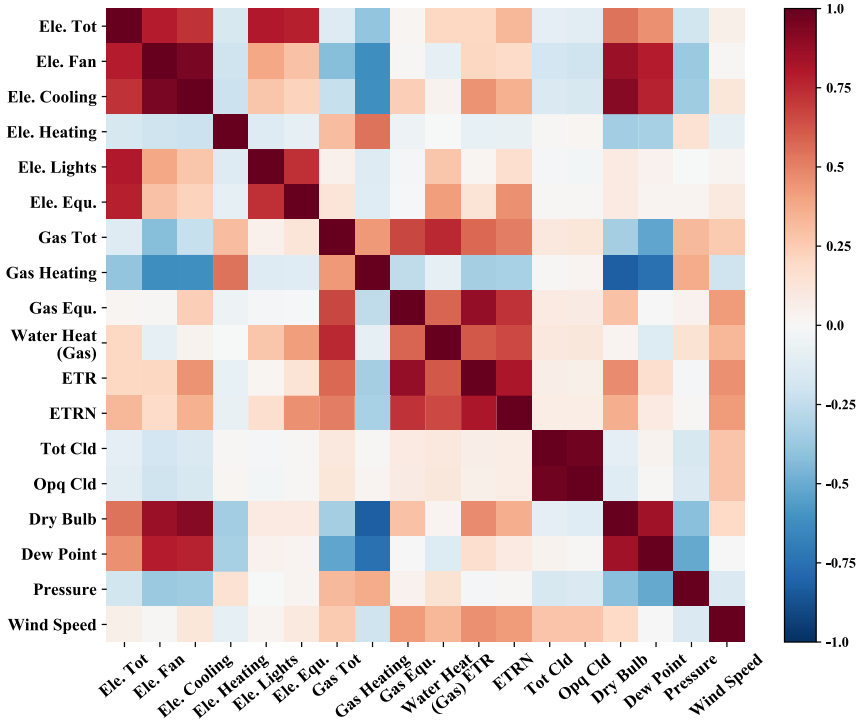

Figure 2-3. Correlation between different features and sectional consumption

Two questions should be answered before training the DL models for these multiple tasks. First, how to split the dataset into a training set and a testing set? In this study, to improve the robustness of the model for multiple tasks prediction on different building types, the split happens within the cities and building types rather than the

hours within a year. In other words, if one type of building in some city is randomly chosen to be a portion of the training set, the whole year (8760 hours) data of this building, including all sectional consumptions, will all be only used for model training (testing). Therefore, every piece of data from this building, including the information about the building itself, will never leak into the testing (training) set. Figure 2-4 shows detailed steps of the dataset split. The building type is encoded by one-hot technique. The training dataset is first sampled based on input steps and batch size. Then shuffling the sample order of feed-in to increase the robustness of the models further, but it should be acknowledged that the temporal order in each training sample is maintained. Second, should future meteorology information be used for input of training and testing?

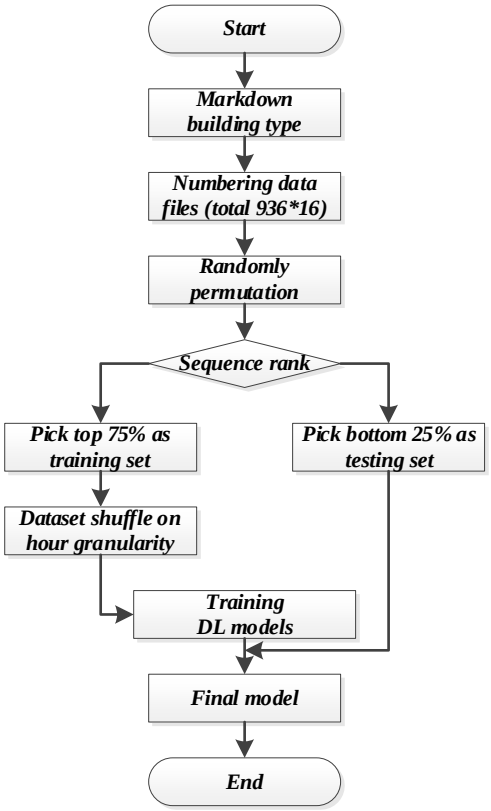

Figure 2-4. Dataset split upon training and testing

As shown in Figure 2-3, energy consumption is strongly related to meteorology information. It is reasonable to include future information as input for real-time cases since future meteorology information is predicted rather than real-time information. However, for simulated energy consumption data, the future meteorology information is the ground truth reference for future consumption. Therefore, to eliminate information leakage and avoid fitting the simulation tool, future meteorology information will never be used as an input feature in our model. Figure 2-5 shows the final data structure in spatial, temporal, and feature dimensions.

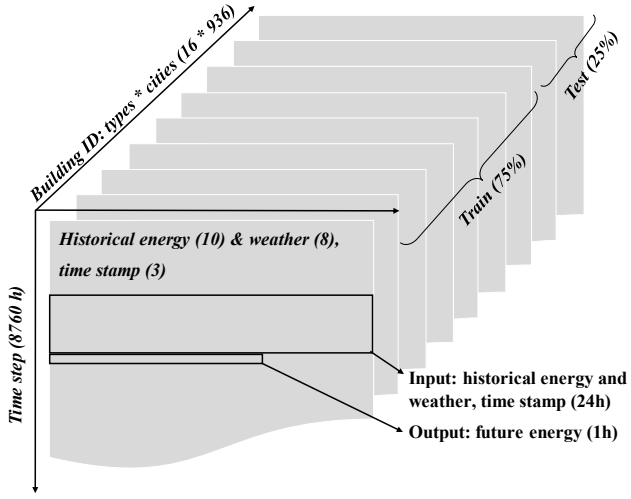

Figure 2-5. Feed-in data structure

2.2 ML models

There are over millions of data points, as mentioned earlier. It will take longer to train and optimize a none-DL model over this million-scale level dataset. Therefore, a small subset of data (10%) was used to train, test, and tune the none-DL models. The best models were selected from these models. Then the best none-DL models were compared with the two proposed DL models in terms of accuracy. The selected none-

DL models include Random Forest (RF), Support Vector Regression (SVR), Gaussian Process Regression (GPR), Extreme Gradient Boost (XGBoost), and artificial neural network (ANN). And the optimized parameters range of grid search is listed in Table 2-4.

Decision trees are a popular method for various machine learning tasks. In particular, trees that are grown very deep tend to learn highly irregular patterns. Random forests are a way of averaging multiple deep decision trees, trained on different parts of the same training set, with the goal of reducing the variance (Hastie, Tibshirani, and Friedman 2009). The training algorithm for random forests applies the general technique of bootstrap aggregating, or bagging, to tree learners (James et al. 2013).

Support vector machine (SVM) is proposed for classification, and SVR (support vector regression) is a regression model based on SVM (Gunn 1998). The purpose of this method is to obtain an optimal hyperplane and the optimization goal is to minimize the farthest "distance" from the sample point to the hyperplane.

Gaussian Process Regression (GPR) is a machine learning method developed based on Bayesian theory and statistical learning theory (Hensman, Matthews, and Ghahramani 2014). GPR does not require an explicit form of objective function because the posterior distribution of the objective function can be automatically learned by fitting training data after assuming it follows a Gaussian process of a specified mean function and covariance function. The non-parametric model assumptions include noise (regression residuals) and Gaussian process prior, and the solution is based on Bayesian inference.

K-nearest neighbor (KNN) algorithm is one of the most basic methods in the data-mining field (J.-J. Wang et al. 2008). The core idea of the KNN algorithm for regression is that the value of a sample is only related to the k most adjacent samples in the feature space. The main point of obtaining a reasonable KNN regression model is to choose the appropriate distance measurement method, K value and classification decision rules.

Extreme Gradient Boosting (XGBoost) is an ensemble learning technique which can combine multiple prediction models in a systematic way to improve their performance (Chen and Guestrin 2016). It adds multiple predictors sequentially which is different to random forest. The new model can be constructed by paying attention to the prediction error (the deviation between the true label and the prediction label) of a trained model. By trying to correct the prediction error of the previous model, XGBoost can improve its accuracy.

Table 2-4. None-DL models and parameters for grid search

Model	Grid search parameter
RF	Max depth: [5, 10, 15]; min samples leaf: [1, 2, 4]; min samples split: [1, 2, 4]; estimators: [80, 90, 100]
SVR	epsilon: [0.001, 0.005, 0.01, 0.1]; C: [0.1, 1, 5, 10] gamma: [0.0005, 0.001, 0.005, 0.01, 0.05]; max iteration =1000
GPR	alpha: [0.01, 0.02, 0.04, 0.06, 0.08, 0.1, 0.12, 0.14, 0.16] kernel: [RBF(0.01), RBF(0.1), RBF(1.0), RBF(10.)]
KNN	N neighbors: [2, 4, 6, 8, 10, 12, 14, 16, 18, 20]
XGBoost	n estimators: [80, 90, 100, 120]; min child weight: [0.5, 1, 2, 4] max depth: [1, 2, 3, 4]; gamma: [0.2, 0.5, 1, 1.5, 2]; subsample: [0.4, 0.6, 0.8, 1.0]
ANN	batch size =512; hidden size: [(25,), (50,), (100,)] solver: ['sgd', 'adam']; alpha: [0.0001, 0.005, 0.01]

The mathematical formulation of these two DL models was first formalized as follows. The mapping input refers to the parameters that affect energy consumption in the historical period (including multiple energy types). The output is the energy consumption that needs to be predicted for a while. The energy consumption prediction mapping can be defined as:

$$\widehat{\boldsymbol{E}}_{\succ T} = \mathcal{F}(\boldsymbol{\Omega}, \boldsymbol{X}_{\prec T}; \boldsymbol{\Theta}) \tag{2-1}$$

where $\boldsymbol{\Omega}$ **is** the set of building features, specifically building type in our study**,** $\boldsymbol{X}_{\prec \boldsymbol{T}}$ is the measurable physical information, such as meteorology information, before time *T*; $\widehat{\boldsymbol{E}}_{\succ \boldsymbol{T}}$ is the output energy to be predicted at time *T*, and $\boldsymbol{\Theta}$ is the learnable parameters of the model. The nonlinear mapping $\boldsymbol{\mathcal{F}}(\cdot; \boldsymbol{\Theta})$ can be obtained by minimizing the expectation of empirical loss function in the definition domain of the collected dataset $\boldsymbol{\mathcal{D}}$. The training process is formalized as:

$$\begin{cases} \boldsymbol{\Theta} = \underset{\boldsymbol{\Theta}}{argmin}\{\mathbb{E}_{\{\boldsymbol{\Omega}, \boldsymbol{X}_{\prec T}, \boldsymbol{E}_{\succ T}\} \sim \mathcal{D}}(\mathcal{L})\} \\ \mathcal{L}(\boldsymbol{E}_{\succ T}, \widehat{\boldsymbol{E}}_{\succ T}; \boldsymbol{\Theta}) = \|\boldsymbol{E}_{\succ T} - \boldsymbol{\mathcal{F}}(\boldsymbol{\Omega}, \boldsymbol{X}_{\prec T}; \boldsymbol{\Theta})\|^2 \end{cases} \tag{2-2}$$

where $\mathbb{E}$ is the mathematical expectation; $\{\boldsymbol{\Omega}, \boldsymbol{X}_{\prec T}, \boldsymbol{E}_{\succ T}\} \sim \mathcal{D}$ indicates data samples from the buildings; $\mathcal{L}$ is an empirical loss function, generally expressed as the difference between the real energy $\boldsymbol{E}_{\succ T}$ and the prediction results $\widehat{\boldsymbol{E}}_{\succ T}$.

Only the energy prediction for the next time step (*t* is the time step index after discretization) is considered. Therefore, the output time step is to be 1. The model can then be written as Equation 2-3 ~ Equation 2-5, where N_t, N_c, and N_E are the input time-steps, the number of input features, and the output features, respectively.

$$\widehat{\boldsymbol{E}}_{t+1} = \boldsymbol{\mathcal{F}}(\boldsymbol{\Omega}, \boldsymbol{X}_{t-N_t+1 \prec t}; \boldsymbol{\Theta}) \tag{2-3}$$

$$\boldsymbol{X}_{t-N_t+1<t} = [X_{j,k}], 1 \le j \le N_t, 1 \le k \le N_c \,; j, k \in \mathbb{N} \quad \text{2-4}$$

$$\widehat{\boldsymbol{E}}_{t+1} = [E_j]^T, 1 \le j \le N_E \,; j \in \mathbb{N} \quad \text{2-5}$$

In the following description, the detailed structures of two deep learning models are shown first, and then all mathematical operators applied in LSTM and CNN are defined. The network training strategy is presented finally.

2.3.1 Convolution neural network

In general, a CNN model is built upon the NN architecture and consists of convolution, down-sampling, and full connection layers. Shifting to the multiple energy types prediction on multiple types of buildings, similar to the image recognition task that treats red, green and blue layers as input channels, all sectional consumptions are treated as the input channels. The simplified CNN structure is depicted in Figure 2-6. A convolutional layer is a basic operation in a convolutional neural network. It obtains the local information of the image by a convolution kernel of a specific size acting on the part of the image area. Local connection and weight sharing are two essential features of convolutional neural networks. A 2-D kernel is in general used for image process. It strides in two directions to calculate the convolutional output. Therefore, it can extract the spatial information of the input signal. The energy consumption only expands in time direction. Then, unlike the traditional 2-D convolution, this study uses 1-D convolution, and the convolution kernel moves in a single direction in the time series data.

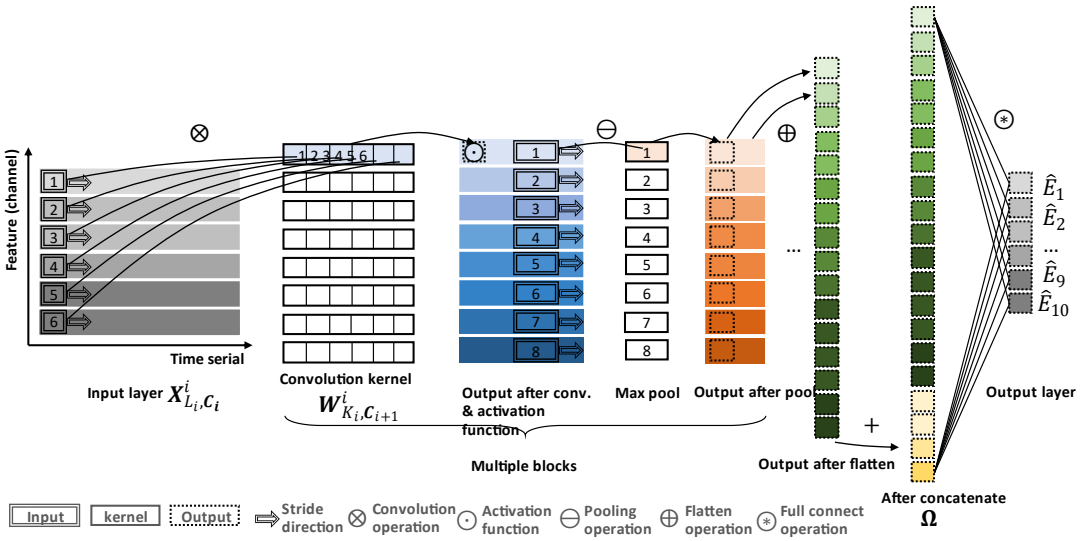

Figure 2-6. CNN structure for multiple energy types prediction on multiple types of buildings

Before making a detailed description, some remarks are needed for the tensors (multi-dimension vectors) that flowed in CNN. Let $\boldsymbol{X}^i_{L_i,C_i}$ denote the output tensor of *i*-th layer in the network. It can also be viewed as a 2-D matrix with the size of $L_i \times C_i$. The superscript L_i is the length of the output tensor, and C_i is the number of channels for layer *i*-th. Especially when the subscript $i=0$ means the input end tensor, $L_0=N_t$, and $C_0=N_c$.

$$\boldsymbol{X}^{i+1}_{L_{i+1},C_{i+1}} = \boldsymbol{W}^i_{K_i,C_{i+1}} * \boldsymbol{X}^i_{L_i,C_i} + \boldsymbol{b}^i_{L_{i+1},C_{i+1}} \tag{2-6}$$

where $\boldsymbol{X}^i_{L_i,C_i}$ represents the input features of *i*-th layer, that is the output features of the previous layer; $\boldsymbol{W}^i_{K_i,C_{i+1}}$ is referred to as the convolution kernel weight. It should be indicated that $*$ is a 1-D convolutional operation defined as follows.

$$\left[\boldsymbol{W}^i_{K_i,C_{i+1}} * \boldsymbol{X}^i_{L_i,C_i}\right]_{j,k} = \sum_{q=1}^{C_i}\sum_{p=1}^{\kappa} X^i_{\delta j+p-1,q} W^i_{p,q,k}, 1 \le j \le L_{i+1}, 1 \le k \le C_{i+1} \tag{2-7}$$

where C_{i+1} is the output channel. $\boldsymbol{b}^{i}_{L_{i+1},C_{i+1}}$ is the corresponding bias of the layer. L_{i+1} is the output length, which can be calculated by $[\frac{L_i-\kappa+2\rho}{\delta}]+1$. Here, κ, δ and ρ denote convolutional filters' kernel size, stride size, and padding size.

A simple elementwise operation $\sigma(\cdot)$ called activation function is applied to all components of the output tensor. In addition, to solve the problem of vanishing gradient in deep neural networks, the leaky rectified linear unit (leakyReLU) is commonly used as an activation function instead of a sigmoid function, which can be written as

$$LeakyRELU: \sigma(x) = \begin{cases} x, & x \geq 0 \\ 0.01x, & x \leq 0 \end{cases} \quad \text{2-8}$$

Upon the pooling layer, a down-sampling layer is employed to compress redundant features of convolutional layers, thereby reducing the situation of overfitting. The 1-D max pooling operation is used to divide the features of each channel into some non-overlapping segments and returns the maximum value. The component of max-pooling on features $\boldsymbol{X}^{i+1}_{L_{i+1},C_{i+1}}$ can be expressed by

$$X^{i+1}_{j,k} = max\{X^{i}_{q,k} | \delta(j-1) < q \leq \delta j\}, 1 \leq j \leq L_{i+1}, 1 \leq k \leq C_{i+1} \quad \text{2-9}$$

where δ is the length of the non-overlapping segment and the output length L_{i+1} can be calculated by L_i/δ. After these operations of several convolutional blocks, the high-level features with a shorter length and more channels are extracted layer by layer. At last, pooling layer, the multi-dimensional tensor should be flattened to a 1-D vector $\boldsymbol{X}^{N_L}$, where N_L is the layer index of the last pooling layer and then concatenated with the building features $\boldsymbol{y}$. This operation would strongly affect building types on the multiple energy types prediction.

$$x_{p\cdot L_n+q+}^{N_L} = \{[X_{q,p}^{N_L}, \boldsymbol{y}]|0 < q \leq L_{N_L}, 0 \leq p < C_{N_L}\} \quad 2\text{-}10$$

Where **y** is the dummy variable encoded as:

$$y_i = 1_{\mathrm{M}}(u_i) = \begin{cases} 1, & when\ u_i \in \boldsymbol{\Omega} \\ 0, & when\ u_i \notin \boldsymbol{\Omega} \end{cases} \quad 2\text{-}11$$

As for fully connected layers, the full connection mainly refits the features to reduce the loss of feature information, and the calculation method is the same as Equation 2-7. Two layers are applied in this architecture. The first layer with a nonlinear activation function is to reorganize the extracted features from the convolutions, and the second layer without an activation function is prepared to output the final target result.

$$\widehat{\boldsymbol{E}}_{t+1} = \boldsymbol{W}_{\mathrm{II}} \cdot LeakyReLU(\boldsymbol{W}_I \cdot \boldsymbol{X}^{N_L} + \boldsymbol{b}_I) + \boldsymbol{b}_{\mathbf{II}} \quad 2\text{-}12$$

Therefore, the learnable parameters in the RNN model can be collected as:

$$\boldsymbol{\Theta} = \{\boldsymbol{W}_i, \boldsymbol{b}_i, \boldsymbol{W}_I, \boldsymbol{b}_I, \boldsymbol{W}_{\mathrm{II}}, \boldsymbol{b}_{\mathrm{II}} | i = 0, \cdots, N_L\} \quad 2\text{-}13$$

2.3.2 Long short-term memory

Long short-term memory (LSTM) is built based on recurrent neural network (RNN) architecture, whose layer deepens along with the time direction shown in Figure 2-7. It is thus suitable for dealing with temporal-dependent tasks. The original input information from the current layer, together with hidden information from the previous layer, is used for the input of the current layer. In detail, LSTM constructs mechanisms such as memory cell *c*, input gate *i*, forget gate *f*, and output gate *o*. The combination of these gates and memory units dramatically improves the ability of the neural network to process long time series data. The structure between them is shown in Figure 2-8. Let $\boldsymbol{X}_t^l$ denote the input vectors of the network $\boldsymbol{X}_{t-N_t+1<t}^l$ at time step *t* in layer *l*. Then the calculation formula for each memory cell is expressed as follows:

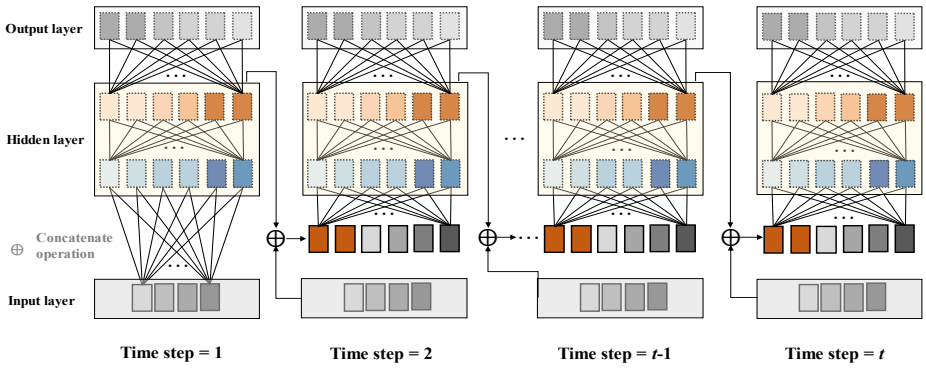

Figure 2-7. An unfold two-layer RNN structure

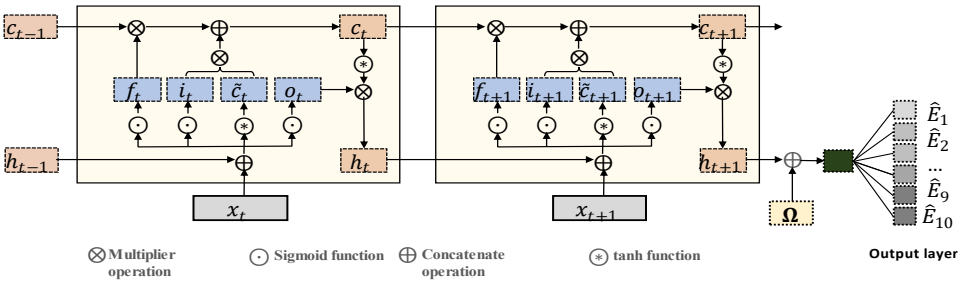

Figure 2-8. LSTM structure

$$\begin{cases} \boldsymbol{i}_t^l = \text{sigmoid}\left(\boldsymbol{W}_{Xi}^l \boldsymbol{X}_t^l + \boldsymbol{W}_{hi}^l \boldsymbol{h}_{t-1}^l + \boldsymbol{W}_{ci}^l \boldsymbol{c}_{t-1}^l + \boldsymbol{b}_i^l\right) \\ \boldsymbol{f}_t^l = \text{sigmoid}\left(\boldsymbol{W}_{Xf}^l \boldsymbol{X}_t^l + \boldsymbol{W}_{hf}^l \boldsymbol{h}_{t-1}^l + \boldsymbol{W}_{cf}^l \boldsymbol{c}_{t-1}^l + \boldsymbol{b}_f^l\right) \\ \boldsymbol{c}_t^l = \boldsymbol{f}_t^l \circ \boldsymbol{c}_{t-1}^l + \boldsymbol{i}_t^l \circ \tanh\left(\boldsymbol{W}_{Xc}^l \boldsymbol{X}_t^l + \boldsymbol{W}_{hc}^l \boldsymbol{h}_{t-1}^l + \boldsymbol{b}_c^l\right) \\ \boldsymbol{o}_t^l = \text{sigmoid}\left(\boldsymbol{W}_{Xo}^l \boldsymbol{X}_t^l + \boldsymbol{W}_{ho}^l \boldsymbol{h}_{t-1}^l + \boldsymbol{W}_{co}^l \boldsymbol{c}_t^l + \boldsymbol{b}_o^l\right) \end{cases} \quad 2\text{-}14$$

$$\boldsymbol{h}_t^l = \boldsymbol{o}_t^l \circ tanh(\boldsymbol{c}_t^l)$$

$$\text{sigmoid}(x) = \frac{e^x}{e^x + 1} \quad 2\text{-}15$$

where, $\boldsymbol{i}_t$, $\boldsymbol{f}_t$, $\boldsymbol{c}_t$, $\boldsymbol{o}_t$ represent the output vector of the input gate, forget gate, memory cell, and output gate, respectively. All the $\boldsymbol{W}$ and $\boldsymbol{b}$ with subscript are the weight matrix and bias vector of the response; sigmoid and tanh are the Logistic and hyperbolic tangent functions, respectively. ∘ represents element-wise multiplication operations. The input gate controls the strength of the new input into the memory cell $\boldsymbol{c}$, the forget gate controls the strength of the memory unit to maintain the value of the previous time, and

the output gate controls the strength of the output memory cell. Like the CNN model, building features $\boldsymbol{y}$ are concatenated with the last layer and the last step (T) with the hidden output layer.

$$\boldsymbol{H}_t^L = [\boldsymbol{h}_t^L, \boldsymbol{y}] \tag{2-16}$$

Then, a fully connected layer is applied between the hidden state and the output power at time step t+1, which also includes a nonlinear activation function and Linear operation as:

$$\widehat{\boldsymbol{E}}_{t+1} = \sigma(\boldsymbol{W}_{\text{out}}^L \boldsymbol{H}_t^L + \boldsymbol{b}_{\text{out}}^L) \tag{2-17}$$

where and $\boldsymbol{W}_{\text{out}}$ and $\boldsymbol{b}_{\text{out}}$ are the weight and bias of the output layer, respectively. Therefore, the learnable parameters in the LSTM model can be collected as:

$$\begin{aligned}&\boldsymbol{\Theta}\\&= \boldsymbol{W}_{Xi}^l, \boldsymbol{W}_{hi}^l, \boldsymbol{W}_{ci}^l, \boldsymbol{b}_i^l, \boldsymbol{W}_{Xf}^l, \boldsymbol{W}_{hf}^l, \boldsymbol{W}_{cf}^l, \boldsymbol{b}_f^l, \boldsymbol{W}_{Xc}^l, \boldsymbol{W}_{hc}^l, \boldsymbol{b}_c^l, \boldsymbol{W}_{Xo}^l, \boldsymbol{W}_{ho}^l, \boldsymbol{W}_{co}^l, \boldsymbol{b}_o^l,\end{aligned} \tag{2-18}$$

In the training process, the derivatives concerning the weights in the kernel were calculated based on reference (Goodfellow, Bengio, and Courville 2016). The first-order gradient-based stochastic optimization algorithm Adamax (Kingma and Ba 2017) was used to optimize the parameters in the two DL models, which performs well for most nonconvex optimization, large-volume datasets, and high-dimensional space. Mathematically, the parameter updates by:

$$\begin{cases} \Delta\boldsymbol{\Theta} \leftarrow \boldsymbol{\nabla}_{\boldsymbol{\Theta}} \mathcal{L}(\boldsymbol{E}_{>T}, \widehat{\boldsymbol{E}}_{>T}; \boldsymbol{\Theta}) \\ \mathbf{m} \leftarrow \beta_1 \cdot \mathbf{m} + (1-\beta_1) \cdot \Delta\boldsymbol{\Theta} \\ \boldsymbol{u} \leftarrow \max(\beta_2 \cdot u, |\Delta\boldsymbol{\Theta}|) \\ \boldsymbol{\Theta} \leftarrow \boldsymbol{\Theta} - \alpha_0/(1-\beta_1)\mathbf{m}/(u+\varepsilon) \end{cases} \tag{2-19}$$

The Huber loss function is expressed as follows:

$$\begin{aligned}&\mathcal{L}(\boldsymbol{E}_{>T}, \widehat{\boldsymbol{E}}_{>T}; \boldsymbol{\Theta})\\&= \begin{cases} \frac{1}{2}\|\boldsymbol{E}_{>T} - \boldsymbol{\mathcal{F}}(\boldsymbol{\Omega}, \boldsymbol{X}_{<T}; \boldsymbol{\Theta})\|^2, & |\boldsymbol{E}_{>T} - \boldsymbol{\mathcal{F}}(\boldsymbol{\Omega}, \boldsymbol{X}_{<T}; \boldsymbol{\Theta})| \le 1 \\ |\boldsymbol{E}_{>T} - \boldsymbol{\mathcal{F}}(\boldsymbol{\Omega}, \boldsymbol{X}_{<T}; \boldsymbol{\Theta})| - \frac{1}{2}, & otherwise \end{cases}\end{aligned} \tag{2-20}$$

2.3 Parameters setting of DL models

Both of the DL models were implemented via the python-based package PyTorch (Paszke et al. 2017). One laptop with i7-8750H 2.2 Hz CPU, 16 G memory, and Nvidia GeForce 1060 8G graphic card was used as the model implementation platform. The network structure was deemed as hyperparameters of the models. Except for the network structure, the settings of other parameters are listed in Table 2-5, which includes the dropout rate in both NN structures, initial learning rate, learning rate decay ratio, and teacher forcing ratio (TFR). TFR is a regularization strategy for training the LSTM model. It describes the possibility of using the model output from the previous time step as the input of the current time step. The last two parameters in this table are the stop criteria of the training process. Overall, the scale of the parameter number measures the complexity of the model and the computational efficiency of training.

Table 2-5. Settings of general hyperparameters

Parameter	CNN	LSTM
Batch size	1,024	1,024
Dropout rate	0.2	0.2
TFR	/	0.5
Initial learning rate	0.001	0.001
Learning rate decay	0.1/ 10 epochs	0.1/ 10 epochs
Minimum loss decrease	$5*10^{-5}$ / epoch	$5*10^{-5}$ / epoch
Patience of early stop	5 epochs	5 epochs

2.4.1 Preliminary analysis

Table 2-6 and Table 2-7 summarize the detailed structure parameter of CNN and LSTM models. And the number of parameters that need to be determined in each layer was also outlined. For the CNN model, the same one-dimension kernel was used for every convolution layer. The kernel width is 3, the striding size is 2, and no padding. Batch

norm was used for regularization. For the LSTM model, the hidden layer size was 20. There was a full connection layer connected with output. It is worthwhile to mention that even though the CNN model has over four times of parameters as the LSTM model, the computational time of the CNN model would not necessarily be longer than LSTM as a result of GPU acceleration.

Table 2-6. NN structure for CNN model

Kernel type	Kernel dimension [size, stride, padding]	Channel [in, out]	Parameter #
Convolution	[3*1, 2, 0]	[21, 64]	4,096
Convolution	[3*1, 2, 0]	[64, 128]	24,704
Batch Norm	/	[128, 128]	/
Leaky ReLU	/	[128, 128]	/
Max pool	[2*1, 2, 0]	[128, 128]	/
Flatten	/	[128, 256]	/
Concatenate	/	[256, 272]	/
Full connection	/	[272, 10]	2,730
Total	/	/	31,530

Table 2-7. NN structure for LSTM model

Kernel type	Channel [in, out]	Parameter #
LSTM	[22, 20]	3,440
LSTM	[20, 20]	3,280
Concatenate	[20, 36]	/
Full connection	[36, 10]	370
Total	/	7,090

2.4.2 Parametric study

A parametric study was carried out to study how the hyperparameters would influence the performance of these two DL models. The investigated hyperparameters included the input step, the number of NN layers, and the loss function. Table 2-8 and Table 2-9 list these parameters' detailed settings. The convolution kernels in different CNN

models were designed to be the same for eliminating the impact of kernel size on performance.

Table 2-8. Parametric table of CNN and LSTM models

	Hyperparameters		
Model	Input step	NN layer	Loss function
CNN	12, 24, 36, 48	1, 2, 3, 4	L2, Huber
LSTM	12, 24, 36, 48	1, 2, 3, 4	L2, Huber

Table 2-9. Model size under different parameter settings

Hyperparameters	Settings	Parameter number of CNN	Parameter number of LSTM
Step	12	30,250	7,090
	24	31,530	7,090
	36	34,090	7,090
	48	35,370	7,090
Layer	1	16,298	3,800
	2	31,530	7,090
	3	261,418	10,360
	4	655,146	13,640

2.4 Evaluation metrics

Many indicators can be used to evaluate the prediction performance of the ML model. There are two categories. The first category is absolute error, including mean absolute error (MAE), mean square error (MSE), and root mean square error (RMSE). The second category is the relative error, which is dimensionless, like the coefficient of variation of the root mean square error (CVRMSE), R square (R^2), and mean absolute percentage errors (MAPE). The error value calculated by CVRMSE is larger than that of MRMSE and MAPE under the same ground truth and prediction value (Zhe Wang, Hong, and Piette 2020). CVRMSE is recommended by ASHRAE (ASHRAE 2020).

Therefore, MAE and CVRMSE are chosen as evaluation criteria. The detailed expressions of MAE and CVRMSE are listed below.

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|\widehat{y_i} - y_i| \quad 2\text{-}21$$

$$CVRMSE = \frac{\sqrt{\frac{1}{N}\sum_{i=1}^{N}(\widehat{y_i} - y_i)^2}}{\bar{y}} \quad 2\text{-}22$$

Where $\widehat{\boldsymbol{y_i}}$, $\boldsymbol{y_i}$ and $\bar{\boldsymbol{y}}$ are the predicted value, ground truth, and mean value, respectively.

2.5 Prediction results

The best non-DL model is selected based on the minimum average CVRMSE value of all building and energy types. The best performance after grid search on each none-DL model can be viewed in Table 2-10. It is observed that the best none-DL model is ANN with a mean CVRMSE of 0.566. Then ANN is used to train and test on the same dataset used as the DL models. In the following section, the performance between DL and none-DL models was compared, and the results of accuracy and computational efficiency for both CNN and LSTM models were described. Last, the influence of hyperparameters listed in Table 2-10 was compared with the basic settings of these two DL models.

Table 2-10. Optimized none-DL models and Performance comparison

Model	Optimized parameters	Mean CVRMSE
RF	Max depth: 15; min samples leaf: 1; min samples split: 2; estimators: 100	0.82
SVR	epsilon: 0.1; C: 5 gamma: 0.005; max iteration =1000	0.82
GPR	alpha: 0.1; kernel: RBF(1.0)	0.79
KNN	N neighbors: 10	1.27
XGBoost	n estimators: 120; min child weight: 2 max depth: 1; gamma: 0.5; subsample: 0.6	0.64
ANN	batch size =512; hidden size: 100 solver: 'adam'; alpha: 0.005	0.57

2.6.1 Case study of whole year performance

Multiple energy prediction yields all energy consumption from different sectors simultaneously. Considering that the testing set contains 3,744 building files and ten tasks in 8760 hours need to be examined in each building file, it would be inefficient to display all the prediction results. Therefore, for a clear perspective of the models' performance on this multiple energy prediction, the prediction performance of total energy consumption tasks on a particular building for a whole year is visualized. The building performance is viewed from scatter plot and time series plot for both ANN and two DL models. Then, for a better view of the performance of different models on different buildings and total energy consumptions (total electricity and total gas), the best and worst performance of these three models concerning the CVRMSE values are plotted.

2.6.1.1 Performance on a large office

Figure 2-9 through Figure 2-11 show the performances of the ANN model, CNN model, and LSTM model for the prediction tasks of total gas and total electricity consumption, respectively. The building is a large office located at Baltimore-Washington International airport in Maryland, U.S. Figure 2-9a-Figure 2-11a and Figure 2-9b-Figure 2-11b depict the ground-truth value and predicted value in a year. Figure 2-9c-Figure 2-11c and Figure 2-9d-Figure 2-11d illustrate the comparison via scatter plots, which show the unsatisfied accuracy of the ANN model and CNN model on the total energy consumption tasks. Part e in Figure 2-9 through Figure 2-11 zoom in on the performance during summertime. It is observed that the CNN model and ANN models can hardly reveal the real consumption volume in the summertime.

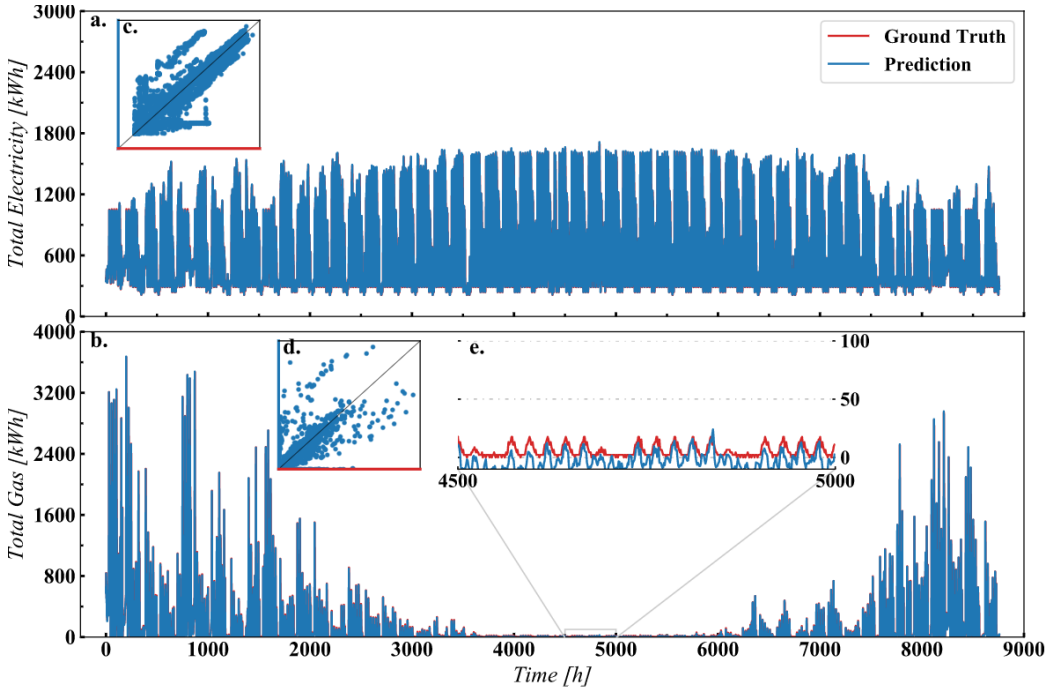

Figure 2-9. Performance of ANN model a. Performance on total electricity consumption b. Performance on total gas consumption c. Scatter plot of ground truth and prediction for electricity d. Scatter plot of ground truth and prediction for gas consumption e. Performance on a low gas consumption time region

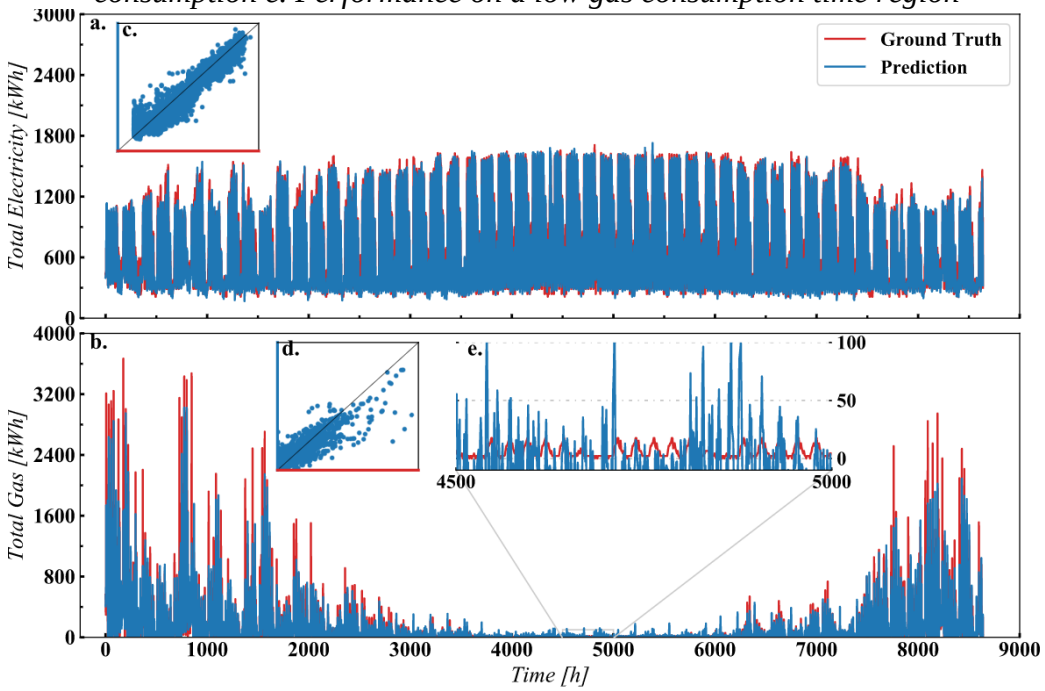

Figure 2-10. Performance of CNN model a. Performance on total electricity consumption b. Performance on total gas consumption c. Scatter plot of ground truth and prediction for electricity d. Scatter plot of ground truth and prediction for gas consumption e. Performance on a low gas consumption time region

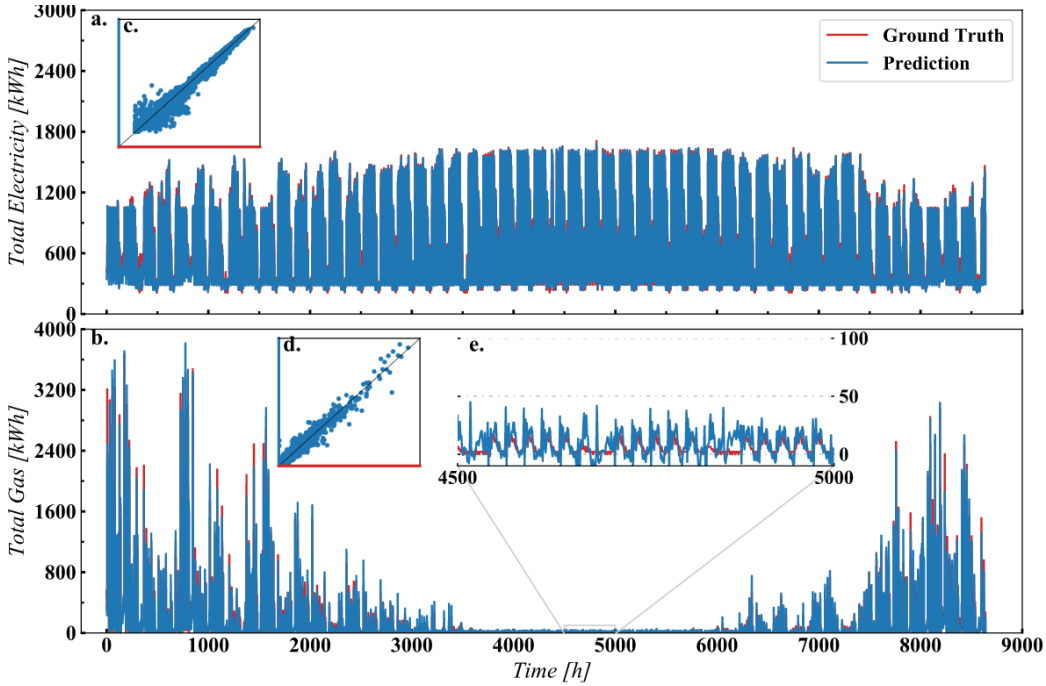

Figure 2-11. Performance of LSTM model a. Performance on total electricity consumption b. Performance on total gas consumption c. Scatter plot of ground truth and prediction for electricity d. Scatter plot of ground truth and prediction for gas consumption e. Performance on a low gas consumption time region

Contrarily, the predicted value LSTM by model approaches the ground truth more closely. Moreover, the performance during summertime is also better than the ANN model and CNN model.

2.6.1.2 Best and worst performance comparison

To visualize the performance on a broader range of buildings, these three models' best and worst performance in terms of CVRMSE values is plotted and compared in Figures and Figure 2-13, respectively. The time series plot can be found in supplementary section from reference (Gao et al. 2021). The best performance for the LSTM model has an obvious advantage over ANN and CNN models for both electricity and gas. However, the worst performance of the LSTM model does not show much dominance compared with CNN and ANN models, especially for gas prediction. Whereas this is

only the prediction performance from a single building and single energy type. An overall performance evaluation should be considered to compare ANN and two DL models comprehensively.

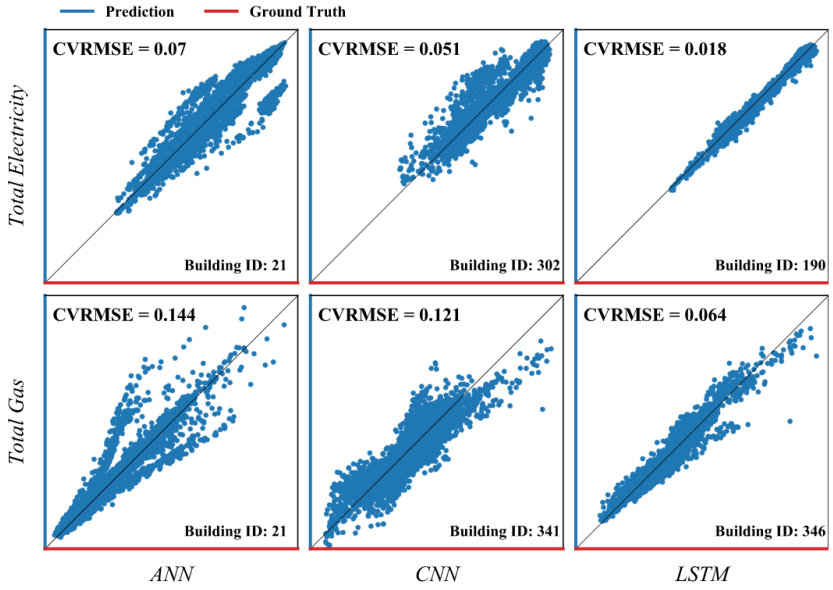

Figure 2-12. Best performance comparison on total gas and electricity

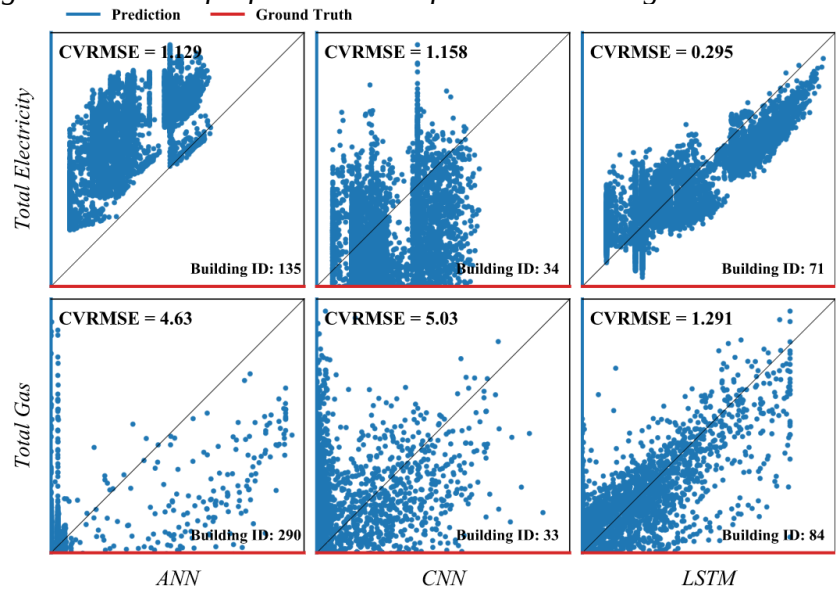

Figure 2-13. Worst performance comparison on total gas and electricity

2.6.2 Overall performance comparison

As discussed in the data analysis section, energy consumption has wide ranges and scales for different sectors of various building types. It is essential to look at the prediction performance from both absolution and relative error viewpoints for multiple energy types on all building types. Consequently, the error of every task in each building file was calculated and grouped by building types in the same statistical way, as shown in section 2.1. Basic CNN and LSTM models were compared first for these two metrics. The model input step is 24 hours in these two basic models, and the neural network follows the structures shown in Table 2-6 and Table 2-7.

2.6.2.1 MAE distribution

The absolute error distributions of ANN models for the testing set are summarized in Figure 2-14. For convenience, the MAE in three buildings groups were consistently discussed from the left side figure of small-scale consumption to medium-scale consumption and the right-side figure of large-scale consumption. From the bottom figure of total energy consumption to sectional gas consumption and last sectional electricity consumption in different tasks.

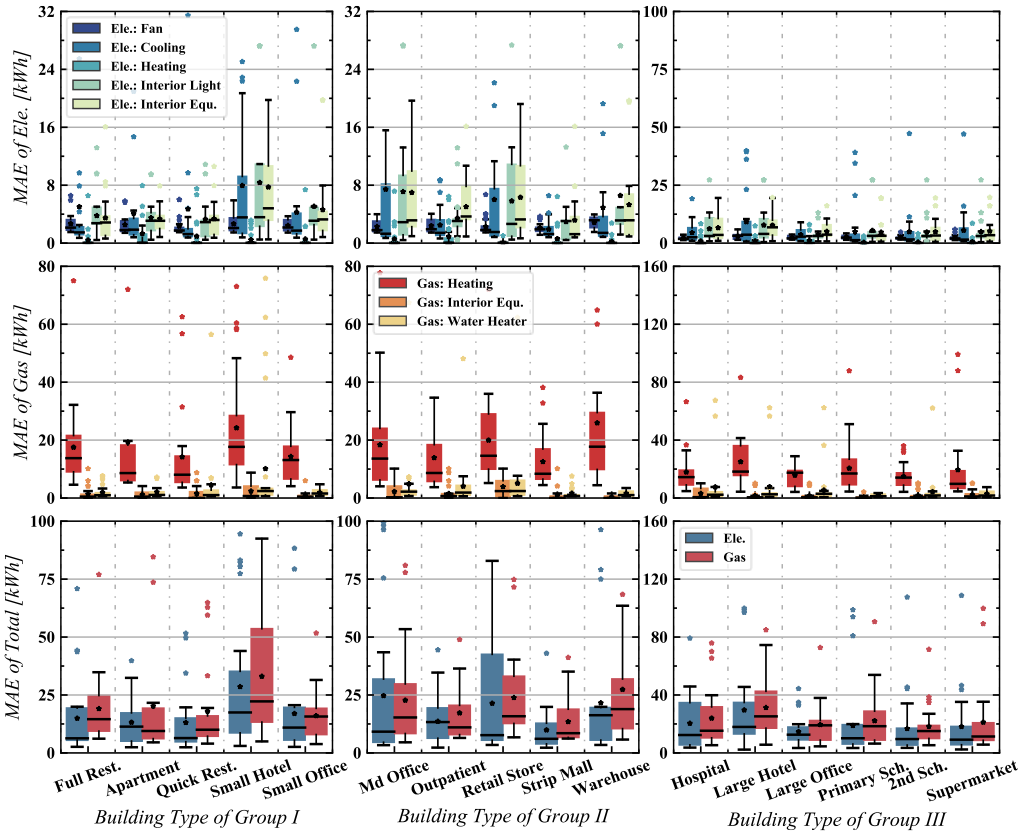

Figure 2-14. MAE of multiple energy types on different types of buildings by ANN model

For Groups I, II, and III, the errors have even wider total and sectional energy consumption ranges for all building types. Specifically, the errors in predicting total electricity and gas concentrate on a wide range between 5 kWh and 100 kWh. The error value is relatively larger for small hotels, retail stores, and warehouses. The errors for predicting sectional gas consumption in the three groups are more related to gas heating, considering the range and scale of total gas and sectional gas-related prediction. As for sectional electricity consumption, the error range is also evenly distributed for all building types, with a range between 2 kWh and 25 kWh. However, various outliers are observed for electricity prediction. The ANN model has better performance on large consumption scale buildings in different building types.

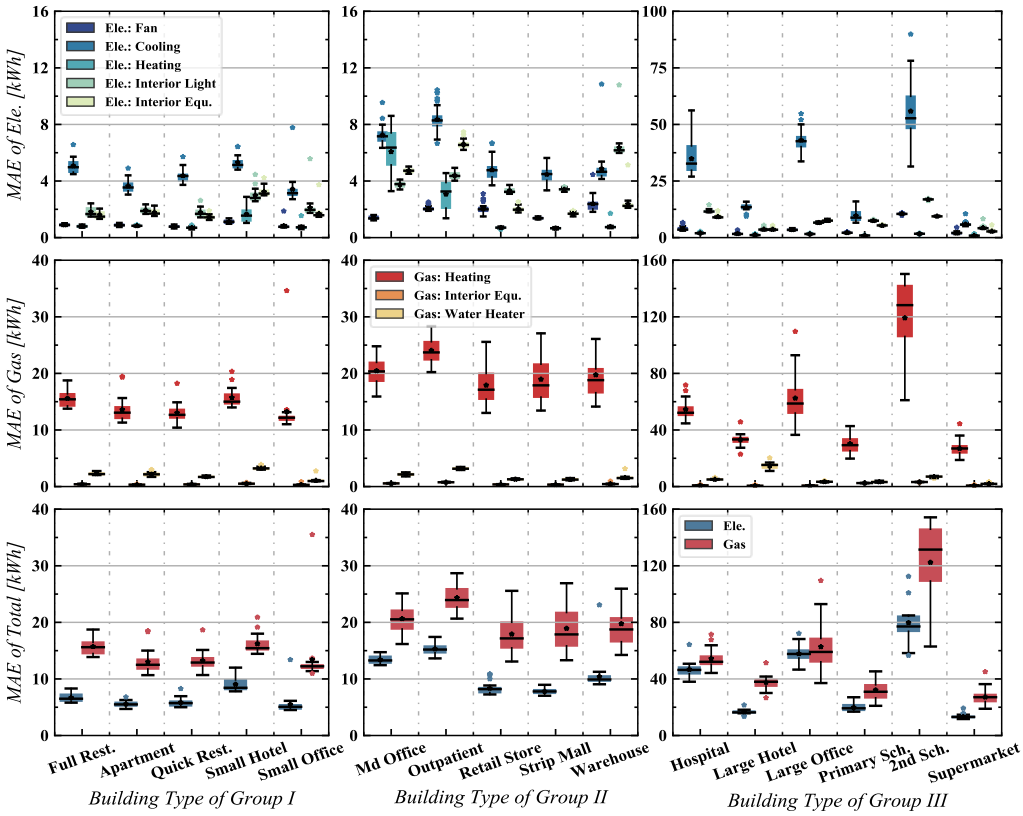

Figure 2-15. MAE of multiple energy types on different types of buildings by CNN model

The absolute error distributions of CNN and LSTM models for the testing set are summarized in Figure 2-15 and Figure 2-16. In G*roup I*, the total electricity prediction errors for all building types concentrate on a range between 5 kWh and 10 kWh. The total gas prediction error ranges from 10 kWh to 21 kWh, except for one outlier from the small office reaching 35 kWh. The errors for predicting sectional gas consumption in this group are bounded to gas heating. This is similar to the errors in predicting total gas consumption. The errors in predicting other sections are almost equal to zero. The prediction errors of electricity for fans, heating load, lights, and other interior equipment are below 4 kWh. However, the prediction error of electric cooling ranges between 4 kWh and 10 kWh. The outlier values of sectional electricity are relatively larger for small offices. In *Group II*, the error of total electricity consumption is around

10 kWh. And the error in predicting total gas consumption ranges from 25 kWh to 29 kWh. The errors in predicting sectional gas consumptions remain similar to that of total gas. The errors in predicting sectional electricity are all below 12 kWh. The errors in predicting electric heating are equivalent or larger than that of predicting electric cooling for median office and retail stores. Interior equipment and lights are predicted with higher errors in this group. In *Group III*, secondary school has the largest error value for total gas, electricity, and all sectional energy prediction tasks. More specifically, the errors for total gas and total electricity range from 60 kWh to 110 kWh and from 60 kWh to 150 kWh, respectively. Moreover, the errors for gas heating and electric cooling are notably high.

As for the LSTM model, in *Group I*, the absolute errors for the total electricity prediction are below 5 kWh. Significantly, the error of predicting total electricity in a quick restaurant is about 2 kWh. The errors in predicting total gas are below 10 kWh except for the full restaurant. For sectional gas consumption prediction, the errors in predicting gas heating and water heater are below 10 kWh and 5 kWh, respectively. The errors in predicting interior equipment are almost zero for all buildings in this group. The sectional electricity has error ranges below or equal to 4 kWh. And errors in predicting electric cooling are still the largest among all sectional electricity consumptions. For *Group II*, there are a few outliers of error for total gas prediction from warehouses reaching 21 kWh, whereas errors for other building types are below 20 kWh. The errors in predicting total electricity for retail stores, strip malls, and warehouses are below 8 kWh. In contrast, it reaches 10 kWh for median offices and outpatient buildings. For sectional energy prediction, the error distribution for gas

heating is like that of total gas. Errors of other sectional gas predictions are almost zero in this group. The errors in predicting sectional electricity remain below 7 kWh. The errors for predicting electric cooling rank first for all types of building in this group. For Group III, the secondary school still has a relatively high error value in total gas prediction. The error in predicting total electricity for all buildings in this group ranges from 8 kWh to 35 kWh. The error in predicting sectional gas ranges from 10 kWh to 45 kWh. The errors in predicting sectional electricity are as low as 10 kWh except for errors in electric cooling in hospitals, large offices, and secondary school buildings.

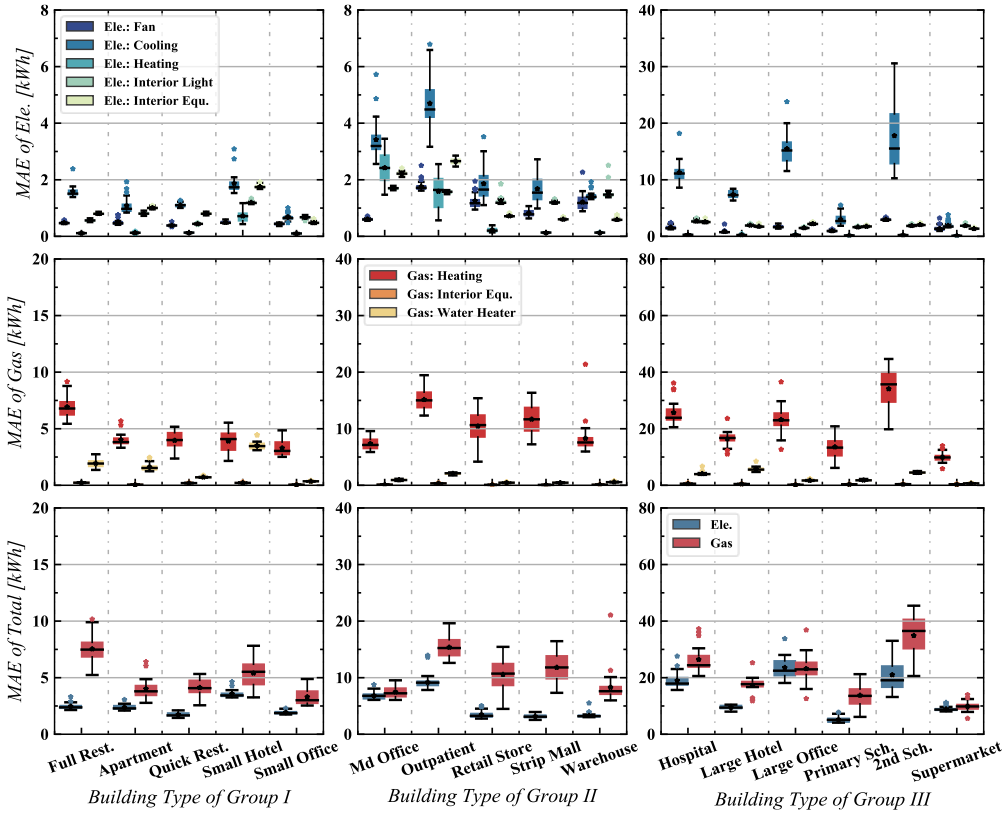

Figure 2-16. MAE of multiple energy types on different types of buildings by the LSTM model

2.6.2.2 CVRMSE distribution

Since the scales of energy consumption differ significantly for different building types and energy types, it is necessary to discuss the relative error in building types and

multiple energy types aspects. Figure 2-17 shows CVRMSE for the CNN and LSTM models, respectively. The number in the figure represents for the CVRMSE value, the darker color means a larger value. The hospital has the CVRMSE values ranging from 3% to 29% in the building type dimension for the CNN model. Small office shows this index ranges from 77% to over 100%. In multi vectors dimension, the relative error of predicting gas consumption by interior equipment ranges from 3% to 66%. The relative errors in predicting electric cooling and gas heating are almost 100%. As for the LSTM model, in the building type dimension, the relative error of hospitals ranges from 3% to 26%. And the relative error of small offices ranges from 33% to over 100%. In the multiple energy types dimension, the CVRMSE of total electricity consumption ranges from 3% to 45%, whereas it ranges from 1% to 100% for total gas consumption. When summing up all the prediction tasks in different building types and energy types, 60% of the tasks predicted by LSTM satisfy ASHRAE Guideline 14 with a CVRMSE under 30%.

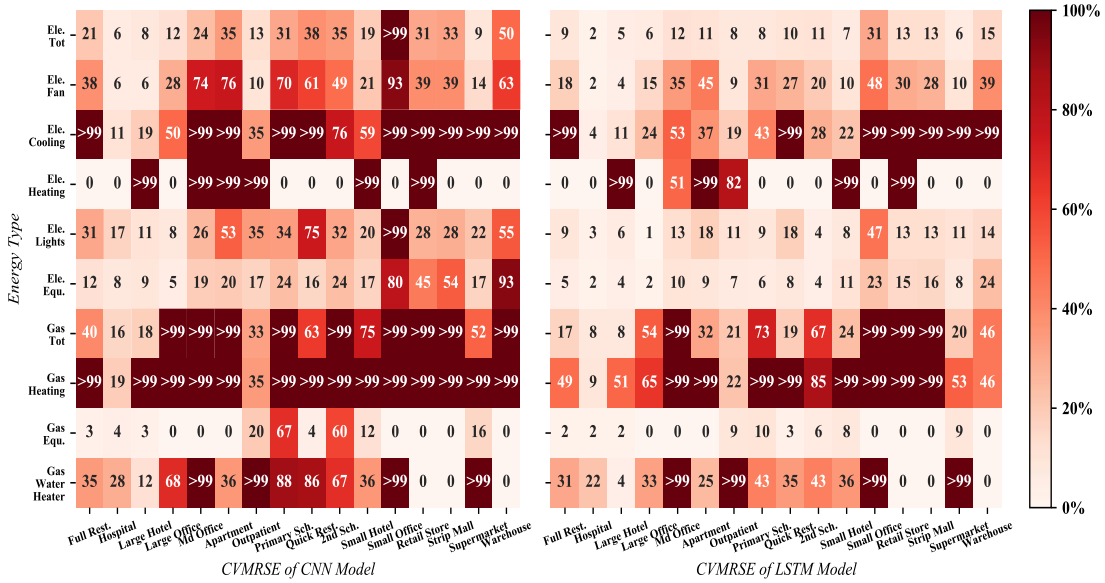

Figure 2-17. Overview of the accuracy comparison in terms of CVRMSE

2.6.3 Performance analysis

The two DL models have better performance over the ANN model, especially for overall performance on all types of buildings. Comparing the best and worst performance of total electricity and gas prediction, the LSTM model has an obvious advantage over the other two models. The ANN model has a larger but even error distribution on all the building types than the two DL models. The two DL models have the potential of providing multiple energy predictions for different building types. The performance of LSTM is relatively better than the CNN model, no matter from the MAE or CVRMSE point of view. In terms of the absolute error, both models perform better in predicting total electricity consumption compared with that total gas consumption. For predicting sectional energy consumption, the performance of predicting gas heating and electric cooling has a relatively higher MAE value. The CNN model performs well in predicting gas consumption for interior equipment in terms of relative error. Hospital is the best building category in multi vectors prediction. The LSTM model performs well in predicting total electricity, electricity for lighting, electricity for interior equipment, and gas for interior equipment. It also performs best in the hospital category. The worst performance concentrates on predicting total gas, sectional gas, and electric cooling consumption. As for building types, the performances in different building types have a strong relationship with the consumption characteristics of each building type. The unsatisfied performance is closely related to those building types that are heavily influenced by different seasons, such as offices, apartments, schools, and strip malls.

In terms of the data characteristics, three factors pose the challenge for accurately predicting energy consumption in a single model. First, different commercial buildings show different scales for total energy consumption. Second, in the energy type dimension, different types of energy have distinct scales. Last, any types of energy have distinctive fluctuation patterns in a year. There are many low-value data for season-related consumption, such as gas-related energy consumption and the electricity for cooling demand. These inconsistent characteristics make multiple energy predictions on some types of buildings unsatisfied. For instance, the small office has a minimum energy consumption in terms of the energy scale. Large hotels and hospitals have few outlier points of electric cooling.

Comparing these two DL models, the CNN model was designed to tackle spatially related tasks like computer vision problems. However, time-related information cannot be passed down if no special operation is implemented for the convolution process. In this CNN model, the 1D kernel was implemented, therefore, it works as a multiplier filter to move in the time serial direction. The kernel stride process is a vector that moves over the input data, performs the dot product with the sub-region of input data, and gets the output as the vector of dot products. The patterns in a short period would be recorded and passed to the following channels of layers. Besides, timestamps can be added as additional input features for indicating fluctuation patterns of energy consumption. However, the CNN model still lacks accuracy and requires fine-tuning or more complex structures. As for the LSTM model, the cell and hidden layer, together with gated operators, can be used to load previous information. Then the temporal information can be inherently passed to the following time steps. Therefore, the LSTM

model outperforms the CNN model when a similar layer structure is constructed. Specifically, for the same building type and task type, the value of absolute error for the LSTM model is half of the CNN model and improves relative error for most tasks and building types. Considering multiple energy types on all types of buildings, the LSTM model predicts 50.7% of the tasks with CVRMSE lower than 20%. And 22.8% of the tasks with CVRMSE between 20% and 50%.

On the other hand, it is also observed that some of the CVRMSE values exceed 30%. Besides the factors associated with the distinguished scales of different building types and different energy types, the energy consumption scale is also different for the same building type and energy type at various locations (cities) due to the thermal settings and meteorology characterizations. Moreover, there are only ten types of buildings for each location. This feature variety is relatively small compared with the continuous meteorology information as input. Last, since the energy consumption profile has a wide range of different energy types and building types, A slight deviation of prediction on small-scale consumption buildings, such as small offices or apartments, would make the relative error large. These made some of the predictions unsatisfied.

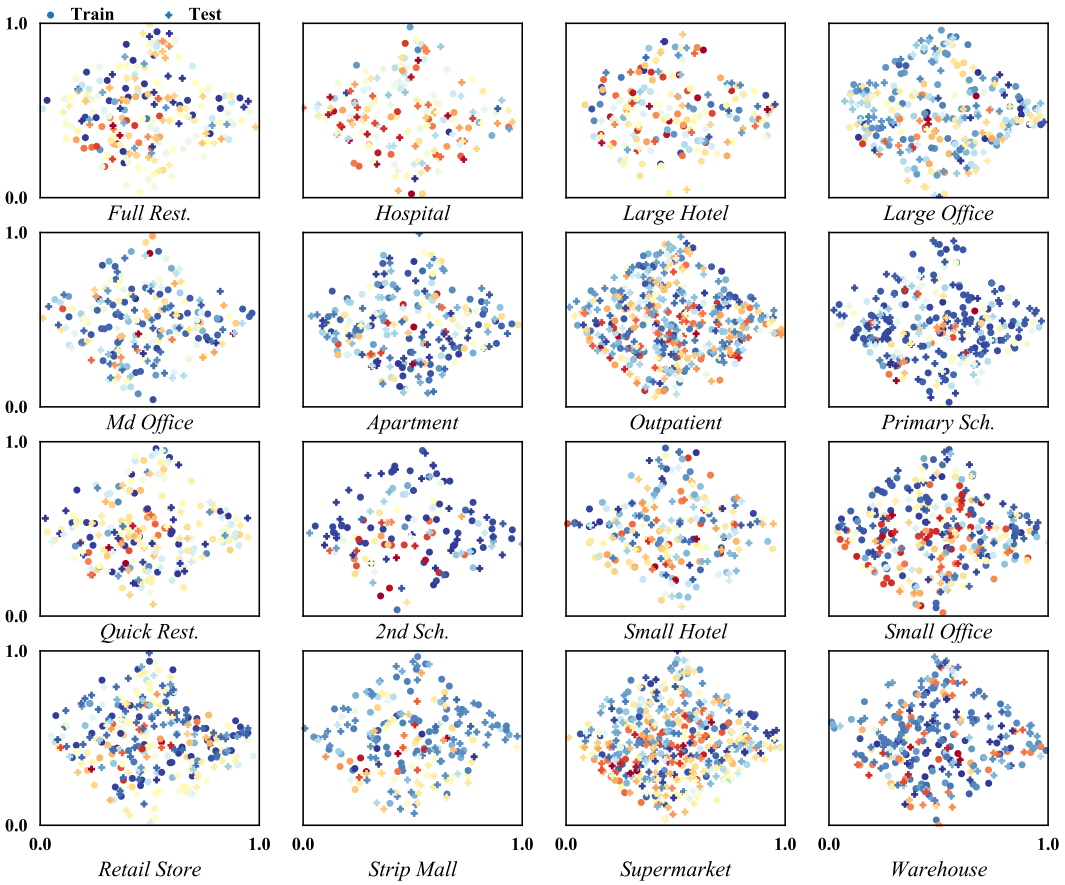

Figure 2-18. Dimension reduction of original input features before training

To further illustrate why the LSTM model has various ranges of errors on different building types, principal component analysis (PCA) dimensionality reduction is performed on the original data and the internal output of DL before pattern concatenation. The dimension of the output feature for original features and after training is two (Price et al. 2006). The results are shown in Figure 2-18 and Figure 2-19. It is observed that the original parts are very chaotic. However, the feature distribution after training shows a strong pattern.

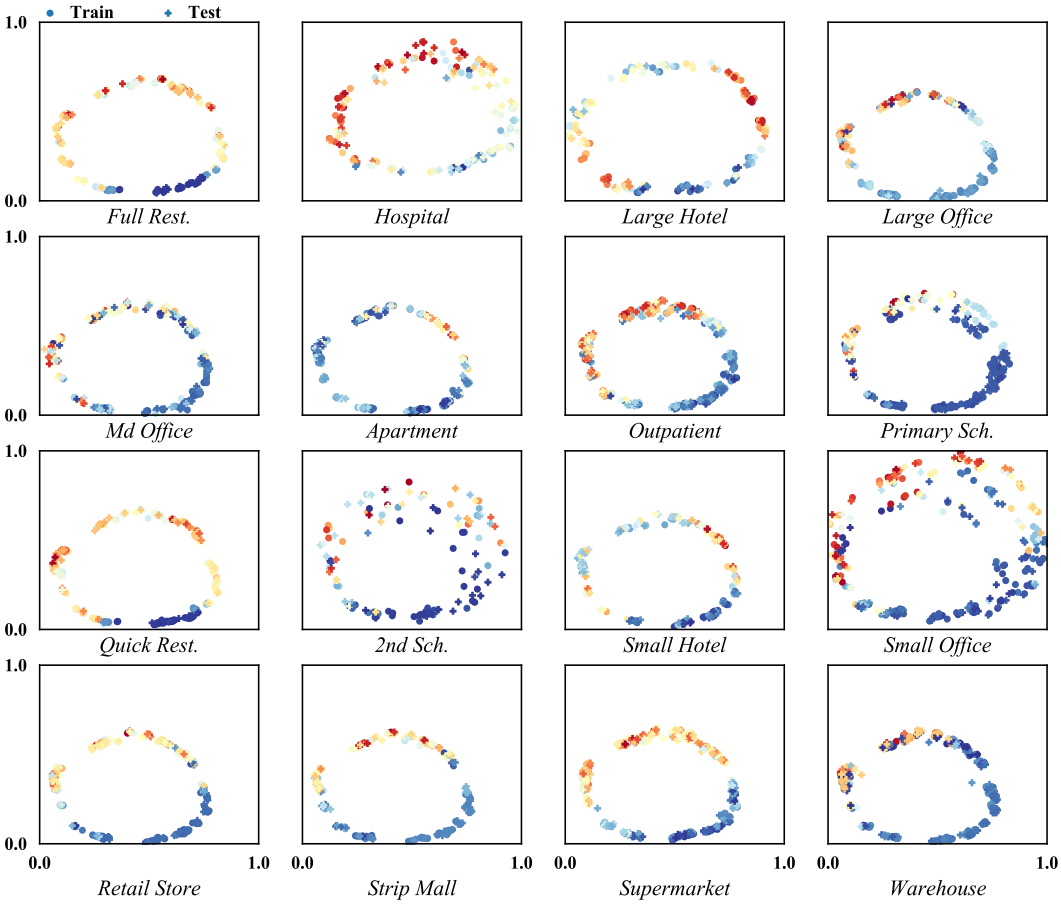

Figure 2-19. Dimension reduction of internal layer features after training

Furthermore, the training and testing datasets overlap, indicating that overfitting is restricted after training. Look back to the error plot in Figure 2-16 and Figure 2-17, where the higher error percentage appears in building type of small office and 2nd school. The relatively weak patterns in these two buildings indicate that the encoding cannot distinguish some of the building types well and might miss other building types. There should be more ample embedding method that can involve more information on building types in the model. This should be explored in the future.

2.6 Hyperparameter influence

This section investigated three critical factors: input steps, neural network layer number, and loss function. To demonstrate their influence on the performance of prediction accuracy and training efficiency, the baseline models are 2-layers and 24-steps trained CNN and LSTM models. The relative change evaluated the performances of CNN and LSTM models with different parameters compared with the corresponding baseline model. For dedicating to the parameters' influence, only the tasks of predicting total energy consumption were picked and presenting these two tasks' performance changes on all building types.

2.7.1 Influence of input step and DL layers

Figure 2-20 and Figure 2-21 show the performance changes on these two tasks by using CNN and LSTM models. A larger input step would not necessarily indicate better performance for the CNN model. For both tasks on most building types, a 36 hours input is the best, and then 48 hours input. A 12-hour input has a similar performance as the 24-hour input. One-layer CNN degrades the performance by around 25%. The enhancements are approximately 25% and 50%, respectively, when three and four layers of NN structure are engaged. The performance is unchanged when the input step changes for predicting gas consumption in apartments and warehouses. It also remains intact for predicting electricity consumption in strip malls and supermarkets. And some building types are not sensitive to layer number. For instance, the task performance does not change even the layer increases when predicting gas consumption in apartments and warehouses or predicting electricity consumption in supermarkets.

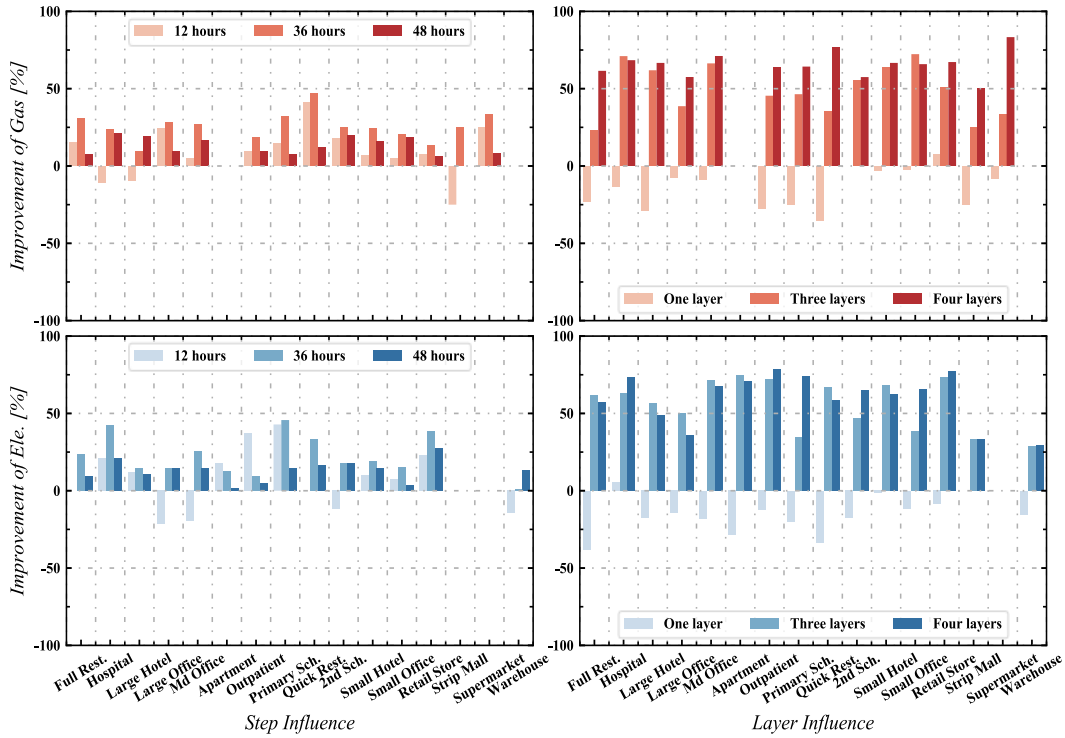

Figure 2-20. Hyperparameters influence on CNN model

As for the LSTM model, the step increase would mostly decrease the prediction performance compared with the 24-step model, except for a slight increase for a large office, middle office, and strip mall. The degradation rates for both tasks of predicting total gas and electricity are larger than 100% on a retail store under 12 or 48 steps mode. And the increase in layer number would slightly increase the performance of most of the buildings. However, the depletion percentage for full restaurant, middle office, apartment, and outpatient would be larger than 100% when choosing a one-layer structure. The task performance of predicting gas for a hospital, outpatient, and secondary school degrades slightly compared with a three-layer structure model. The task performance of predicting electricity for apartments, fast food restaurants, and strip malls degrades over 50% when choosing a three-layer structure. The degradation of outpatient is above 200% for a one-layer model. Like CNN, some building types are

less sensitive to the change of input steps, such as both tasks for full restaurants, gas prediction tasks for apartments and warehouses, and electricity prediction tasks for supermarkets. Other building types are less sensitive to layer change, such as gas prediction tasks on apartments and warehouses and electricity prediction tasks on large hotels, supermarkets, and warehouses.

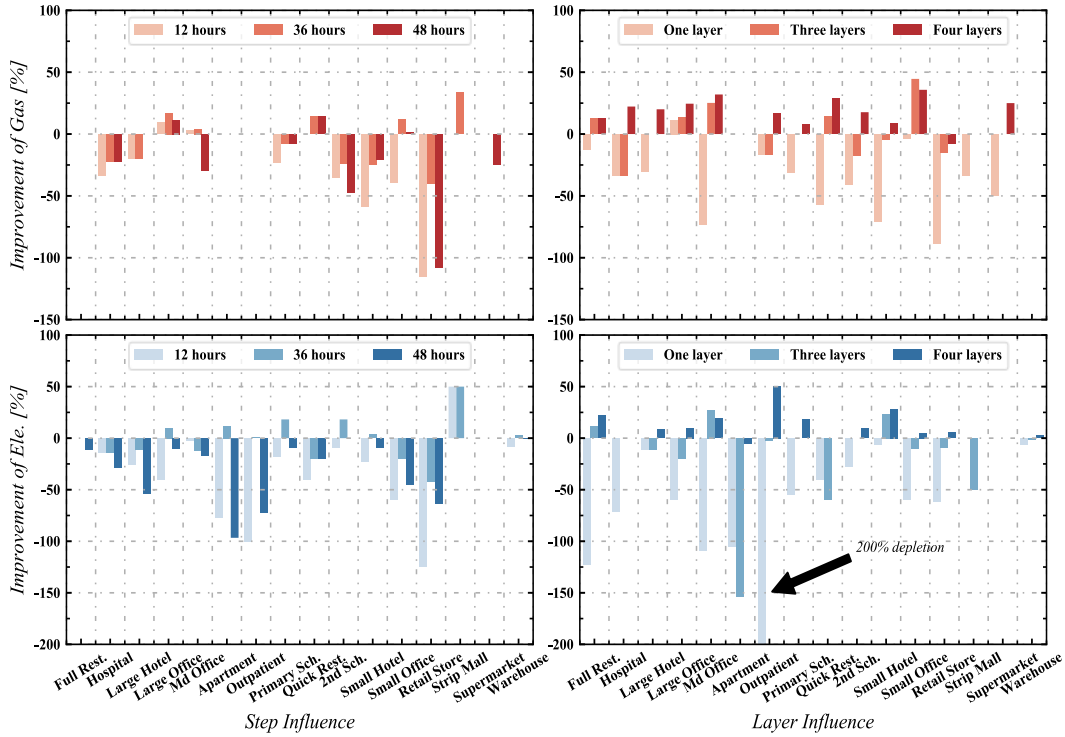

Figure 2-21. Hyperparameters influence on LSTM model

The improvement in predicting accuracy is directly related to how structure of neural network is constructed , hyperparameters and lose functions were chosen. For the CNN model, the influence of steps on prediction accuracy is not merely monotone. Since the time serial characteristics are embedded manually, the kernel shape is essential to the performance of the CNN model. Longer input steps might require a refined convolution kernel in different layers. Moreover, the time step influence is limited when compared with layer influence. A larger number of layers for the CNN model would significantly

boost the model performance since more patterns in the temporal dimension can be extracted for a deeper CNN structure. As for the LSTM model, the larger input step from 24 hours to 36 hours improves accuracy for gas and electricity in most building types. However, when decreasing to 12 hours or increasing to 48 hours, the deterioration is apparent. The relatively shorter input step results in incomplete recognition of the periodic pattern. In this LSTM design, the prediction heavily relies on the output of the last time step.

For this reason, a longer input step might slightly undermine the effect from the very beginning. Attention mechanisms could be added to the LSTM network to tackle the strongly coupling issue of long-time input. Regarding the layer influence of LSTM on the accuracy, a larger layer number might not bring many benefits for most buildings. However, the one-layer LSTM model degraded much because the spatial features are hard to be extracted from this structure.

2.7.2 Loss function influence

All ten tasks are considered when switching to the influence of a loss function. The CNN and LSTM models adopt a 2-layer and 24-step structure. Figure 2-22 and Figure 2-23 show CVRMSE changes compared with the corresponding baseline DL model. The performance of the CNN model with the Huber loss function has little improvement over a conventional MSE loss function. There is slight degradation for the LSTM model compared with a traditional loss function. The prediction errors are still worse for predicting electric cooling, electric heating, gas heating, and water on

almost all building types. The multiple energy performance of the small office is also the worst of all building types.

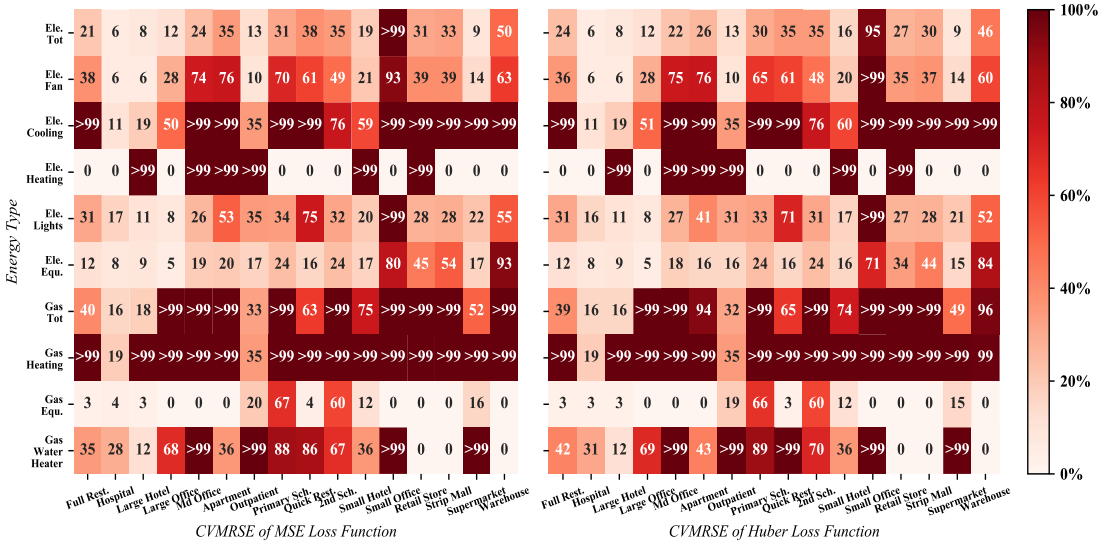

Figure 2-22. Loss function influence on CNN model

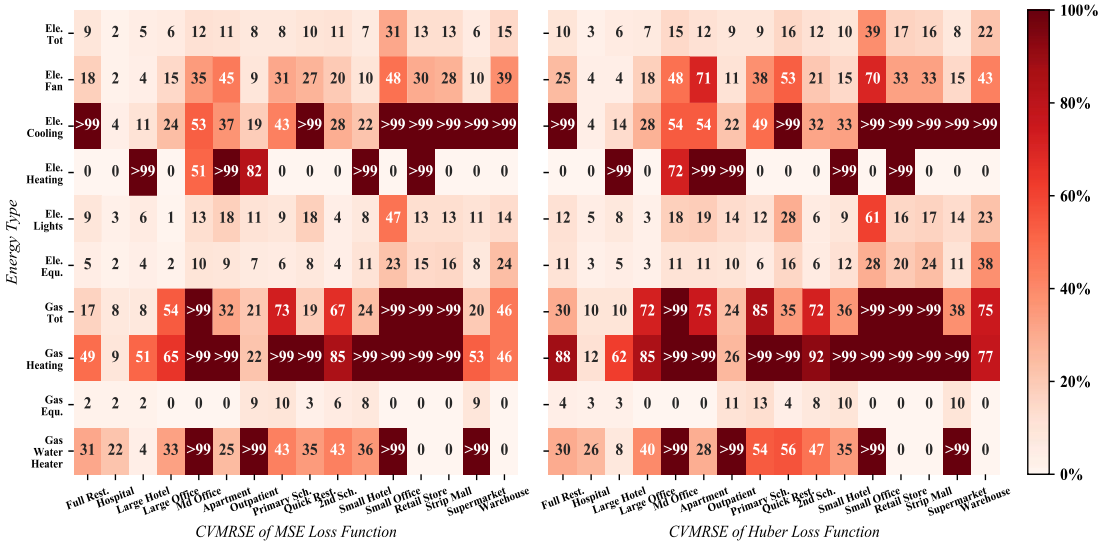

Figure 2-23. Loss function influence on LSTM model

The steps, layers, and hyperparameter in the Huber loss function should be further studied. Systematic optimization work should be carried out to improve the prediction performance by the DL method. Meanwhile, the unsatisfied performance in the tasks of predicting gas-related consumption should be further investigated. In particular, the multiple energy types prediction models need to strengthen the uneven distribution of

gas in a whole year range. Second, the building information is compressed in a one-hot encoded vector. Other feature engineering techniques that can differentiate sufficient building characteristics should be compared for multiple building types of prediction.

2.7.3 Training speed

Previous sections evaluated the accuracy of the DL model for multiple energy predictions on different building types. The tradeoff between accuracy and computational efficiency emerges when determining which model should be implemented. Figure 2-24 shows the computational efficiency for different layer numbers, input steps, and loss functions in total time and time consumption per iteration. First, the total computational time is constant for both CNN and LSTM models under the 12- and 24-step input model structure. Moreover, it increases slightly for LSTM after 24 steps. But a sharp increase is observed with a time step increase from 24 to 48 for the CNN model. The computational time per iteration keeps relatively stable for the LSTM model. However, it decreases by 0.25 seconds from 24 steps to 36 steps and 0.2 seconds from 36 steps to 48 steps for the CNN model. As regards the layer's influence, the total time and the computational time per iteration increase exponentially with layer number increases for the CNN model. However, these values show a linear increase for the LSTM model. The Huber loss function significantly influences the model convergence for both DL models. It decreases by 33% of the total computational time for the LSTM model while increasing by 15% of the total computational time for the CNN model.

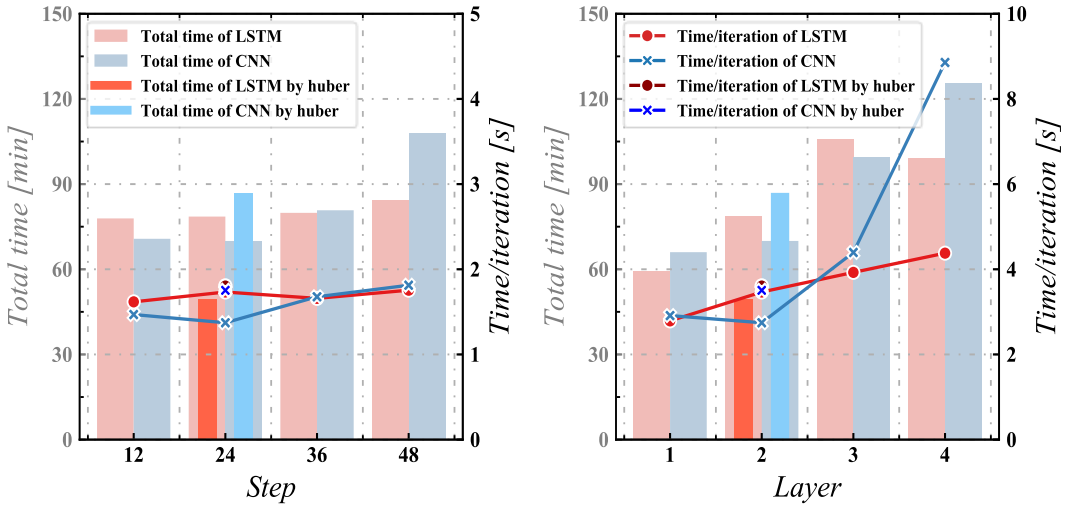

Figure 2-24. Overview of the computational efficiency comparison

The time consumption for training CNN and LSTM models differs when changing input step and layer numbers. Overall, the influence of the step number is less than that of the layer number. The total time consumption is stable for different input steps of the LSTM model, while it linearly increases for the CNN model. And the total time consumption linearly increases when increasing the layer number of the LSTM model, while it exponentially increases for the CNN model. Time consumption per iteration is also more related to the layer number and shows similar patterns as total time consumption. The time consumption fits with the DL model's scale when recalling the parameter number changes with the input step and layer number. The Huber loss function does not influence the performance from an accuracy perspective and even longer the training time for the CNN model. However, it accelerates the convergence of the LSTM model. The Huber loss function pays less attention to outliers in building energy consumption than the L2 loss function. The outliers are clear for this multiple energy prediction on multiple building types, especially in predicting gas-related

energy consumption. Therefore, taking advantage of the Huber loss function would benefit convergence.

2.7 Summary

Energy saving in buildings is critical for increasing total energy efficiency and is significant to maintaining a sustainable society. With growing attention to accurate, rapid, and robust implementation approaches, ML, especially DL, has garnered broad attention in building energy prediction tasks. However, current research works are most concerned about the consumption of single energy types in single building types. They seldom deal with the multiple energy types prediction on multiple building types. This multiple energy prediction is vital when considering the robustness of the prediction model on various types of buildings. It requires comprehensive guidance for the operation of multiple sectors of buildings.

Multiple energy types prediction fills the gap of detailed guidance for the energy supply systems and yields a robust ML model on multiple types of buildings. Non-DL models and two DL models for multiple energy prediction on multiple building types were constructed. The original dataset comprises 16 building types in 936 cities across the US. Each building in one city contains ten types of energy from different sectors. A random 75% of data files were chosen as the training set. The main conclusions can be drawn as follows:

ANN is the best non-DL model among KNN, DT, XGBoost, GPR, and ANN and was chosen as the baseline for DL models. Compared with DL models, ANN has a more even and larger error distribution on all building performance. LSTM model

outperforms the ANN and CNN model from MAE and CVRMSE perspectives. Among all the tasks, the performance of the task for predicting total gas and electricity consumption is better than that of the task for predicting sectional energy consumption such as fan, cooling, heating, and other interior equipment. Besides, the electricity-related prediction is better than gas-related ones.

LSTM has good performance in hospitals and large hotels, satisfying performance in outpatients, secondary schools, apartments, and warehouses, while unsatisfied performance in offices, retail stores, strip malls, and supermarkets. When considering multiple energy types on all types of buildings using the LSTM model, 50.7% of the tasks have CVRMSE lower than 20% and 22.8% of the tasks have CVRMSE between 20% and 50%. It is also worth mentioning that 19.3% of tasks have CVRMSE over 100%, which requires substantial improvement. When summing up all the prediction tasks in different building types and energy types, 60% of the tasks predicted by LSTM satisfy ASHRAE Guideline 14 with a CVRMSE under 30%.

Hyperparameters, such as layer numbers ranging from 1 to 4, input step ranges from 12 to 48 hours, the loss function is chosen from L2 norm, and Huber in the DL model was investigated. The performance of the CNN model is sensitive to the layer number rather than the input step. The convolution kernel should be carefully designed for a specific input step. A two-layer structure is good enough for the LSTM model. Larger layer numbers would not improve the performance much. The LSTM model is more stable under different input steps and layer numbers considering training speed. However, the training time of the CNN model would exponentially increase with layer

increase, which matches the parameter number increase. The Huber loss function help with the training speed for the LSTM model but not the prediction accuracy.

Overall, the performances of the LSTM model on most of the building types are better than the CNN model. The LSTM model is more suitable for this multiple energy prediction. Whereas several limitations should be considered and investigated more in the future. First, attention should be paid to the unsatisfied performance of gas-related tasks. In this regard, the multiple energy types prediction models need to strengthen the uneven distribution of gas in a whole year range, especially in the summertime. Second, the features reduction through PCA shows that the encoder part of DL has limited ability to recognize some types of building patterns, such as 2nd school and small office. Currently, the building information is encoded by one-hot technique, which makes some building type is hard to be differentiated from others. Therefore, other feature engineering techniques or embedding methods that can differentiate sufficient building characteristics should be compared for predicting multiple types of buildings. In addition, the DL-based model is evaluated by simulating multiple energy types on multiple building types. Testing the DL model on a real building consumption dataset would be more valuable. Since it is hard to obtain delicate granularity data for multiple energy types on multiple building types simultaneously, it might be more practical to apply this model with two types of energy on the same type but different buildings. Last, systematic optimization work is required to enhance the overall performance without losing computational efficiency.

3 Optimum Design of S-CO_2 Power System

The optimization design for the energy system concentrates on two aspects, as shown in Figure 1-8. The configuration (also called synthesis in the chemical process area) implies the entire set of functional components like compressor, turbine, heat exchangers, etc., and how they are connected. The performance variable aspect involves the specification of every component that operates at a nominal condition, includes pressure ratio of compressor, effectiveness of heat exchanger, split ratio of splitter, etc. Various studies focus on performance variable optimization based on expertise and experience in the thermal engineering area (Iglesias Garcia et al. 2017). The performance variables are often continuous for a given configuration, like capacity and pressure ratio at the design stage (Yang et al. 2021). Overall, the system design optimization concerns about how the components are connected to each other and how to determine the performance variables of every component to obtain a desire objective.

The topic of configuration optimization has been extensively investigated in chemical process engineering (Mencarelli et al. 2020). However, little work has deeply researched the configuration optimization problem of thermal energy systems. At the same time, researchers have developed various advanced thermodynamic configurations. This development might greatly promote the system performance (L. Wang et al. 2019), such as multiple complex and advanced supercritical carbon dioxide (S-CO_2) power generation configurations summarized in (Crespi et al. 2017). Specifically, the variables in configuration optimization involve discrete variables, making it more complicated when exploring the best system configuration and design.

With more advanced configurations developed, it is essentially impossible to try every candidate. Under most conditions, researchers limit the feasible region to predefined configurations based on basic and mature ones. Therefore, the feasible region is shrunken by the prescreening process and it is easier to be trapped by a local minimum of system configuration design when relying on researchers' experience. Therefore, exploring feasible and robust approaches that reproduce researchers' mind maps is necessary to find the optimum thermal system configuration.

The system evaluation process in configuration optimization can be determined by steady-state (Garcia et al. 2016) or transient state analysis (Xu Zhang et al. 2018), which corresponds to the design and operation level. The system design optimization problem is discussed under steady-state conditions. Therefore, the governing equation for every component is time independent. The operation level optimization is not considered in this chapter. Figure 3-1 shows a Brayton cycle that consists of four components. The performance of this system can be evaluated after solving for the state variables (enthalpy/temperature and pressure) at each connection pipe (x_1-x_4). The governing equations for each component are nonlinear functions, which map the input variables to the output variables.

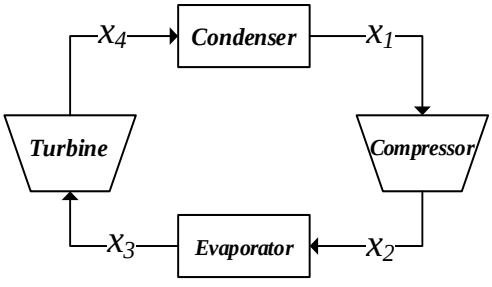

Figure 3-1. Brayton cycle

Since the feasible region of configuration is discrete and mixed with continuous variables in design space, the following subsection first explicitly explains the volume change of feasible region with specific rule constraints. The rest of the subsection is organized by the methodology used to conduct a configuration optimization process, including the whole framework and details of the configuration optimization framework. Then simple and complex case studies are carried out to validate the optimization framework through the S-CO_2 power generation system. Last, the influence of an initial configuration on the effectiveness and convergence of the framework is explored.

3.1 Feasible region of configuration optimization

As mentioned earlier, two folds of variables should be optimized for design optimization, including configuration variables of system and the variables that determine components' performance in a configuration. The general but formal expression of a configuration optimization problem can be represented in Equation 3-1, where X_c and X_p are variables that represent the configuration feasible region of the system level and performance feasible region of components level, respectively; h_j and g_k are the functions that restrict the design variables (configuration and component performance) to feasible region. It is worth mentioning that explicitly expressing constraints towards configuration is difficult. Instead, these constraints may be implemented through configuration generation rules. f is the function that evaluates the interested metrics, such as thermal efficiency.

The number of configurations should be limited for practical usage and there would be many ineffective designs if no rule is applied for connection between components. Therefore, it is essential to confine the feasible region with appropriate constraints for configuration design. Except for the range limitations of variables, the implicit constraints adopted by researchers can be approximately categorized based on different types of knowledge levels: graphic, thermodynamic, and experience/expertise aspects, as shown in Figure 3-2. Here, graphic knowledge means the strongly connected (SC) property of a thermodynamic cycle. The strongly connected means any vertex in a graph can be reached by any other vertex in a directed graph. Mathematically, a graph is strongly connected if it contains a directed path from u to v and a directed path from v to u for every pair of vertices u, v. This rule can also be interpreted as mass conservation for thermodynamic cycles. The feasible region shrinks when more knowledge is applied. Therefore, the question becomes: is it possible to develop appropriate configuration generating rules in this optimization problem without relying on the engineers' or researchers' experience? These generating rules implement all graphic and thermodynamic knowledge requirements to keep as many potential candidates of configuration design as possible.

$$\begin{cases} \min\limits_{X_c, X_d} f(X_c, X_p) \\ s.t.\ \ h_j(X_c, X_p) = 0; j = 1\ to\ J \\ g_k(X_c, X_p) \le 0; k = 1\ to\ K \end{cases} \quad \text{3-1}$$

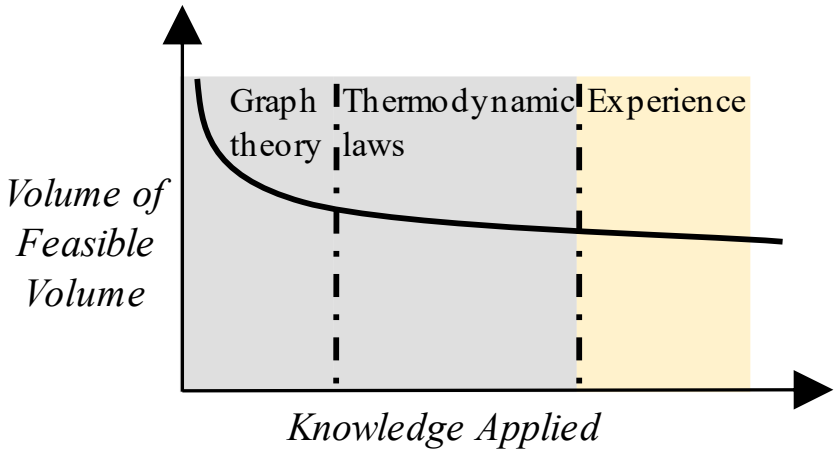

Figure 3-2. Feasible region shrinks as knowledge applied

The candidate configuration number can be calculated by the permutation of the connection among components with or without these rules. Graphic or thermodynamic laws can vastly shrink the number of candidate configurations. The details of calculating the volume of feasible region can be redirected to the appendix section. Figure 3-3 shows that the number of configurations increases when the number of components increases with and without graphic constraints. The results show that the number of configurations satisfying the SC property is much less than the full permutation (indicated with 'all') without considering SC property. This high volume implements exhaustive searching impossible, especially for cases with a larger component number. Here, the split means a component that divides one stream of working fluid into two or more streams in a thermodynamic system. It is noted that the candidate number explodes exponentially when the component number increases, and the magnitude increases three levels with one more split. Considering the feasible region of component performance variable (X_d) is built based on the configuration (X_c), the overall feasible region (X_c, X_d) would be even larger.

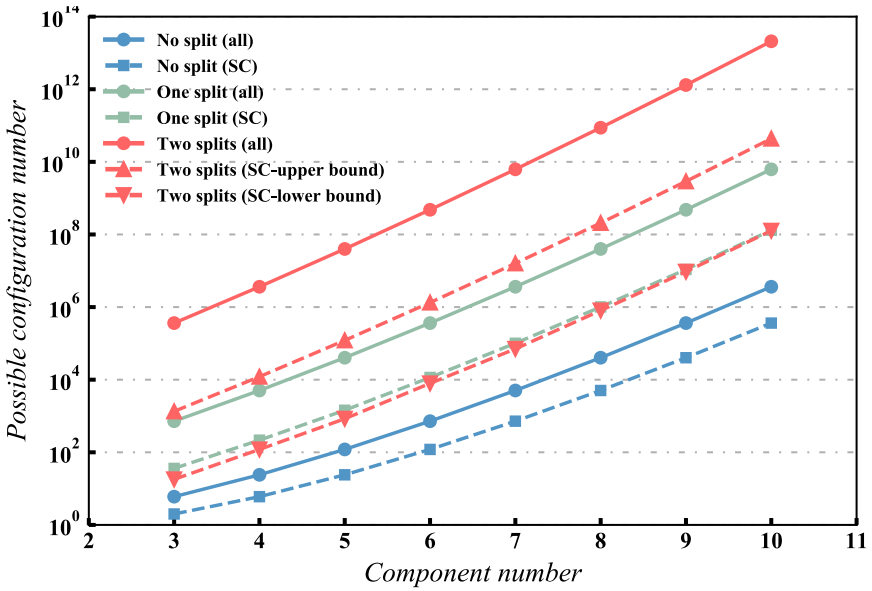

Figure 3-3. Total configuration number increases with component number

Even SC is applied to narrow down the feasible region. The diminished feasible region is still too large (over ten million for only nine components). Therefore, thermodynamic constraints should be applied to decrease the volume further. However, applying too much experience from researchers, such as cascading different configurations, might screen out potential candidates. This thesis only applies SC property and basic thermodynamic knowledge (1st and 2nd laws) for restricting and exploring the feasible region, even though exact algorithms, such as branch and cut, can be used to solve integer programming problems. These algorithms cannot implement the SC and thermodynamic rules to screen out candidates. Therefore, the exact algorithms will spend much time evaluating configuration that does not obey SC and thermodynamic laws. And assigning objective values for these violated configurations will be a problem if one wants to measure the quality of the design. In addition, a second-level optimization will still be needed to optimize the component performance variables, which will slow down the optimization process further. Therefore, a meta-heuristic

method -- simulated annealing-based configuration optimization framework is developed to simultaneously find the best configuration and performance variable setting for the S-CO_2 power generation system. Two case studies based upon limited component numbers for simple and complex configurations are investigated to prove this framework's effectiveness. In the next section, a formal expression of the objective and constraints that describe the optimization problem is carried out.

3.2 Graph representation of thermal system configuration

Modified from the structural graph for chemical process networks proposed in (Emmerich, Grötzner, and Schütz 2001), a thermodynamic graph (***T-graph***) is designed that can fully represent all configurations' information, including topology, components' governing equations, and variables. The completed definition can be described as follows:

$$G \coloneqq (\Omega, V, V_0, \omega^v, C^{in}, C^{out}, E, \pi^c, f^\pi) \tag{3-2}$$

$$\begin{cases} C^{in} \coloneqq \{(j,v) | v \in V \wedge j \in \{1,2,\dots,n^{in}(v)\}\} \\ C^{out} \coloneqq \{(v,i) | v \in V \wedge i \in \{1,2,\dots,n^{out}(v)\}\} \\ E \subseteq \{(c^{out}, c^{in}) | c^{out} \in C^{out}, c^{in} \in C^{in}\} \\ \pi^c : C \mapsto \mathbb{R}^3 \\ f^\pi \mapsto f^\pi(\pi^c(C^{in})) = \pi^c(C^{out}) \end{cases} \tag{3-3}$$

T-graph is defined as a *tuple* in Equation 3-2 and Equation 3-3, where Ω is a finite set of all thermodynamic component types, including compressors, turbines, and heat exchangers, etc.; V is a set of vertices in a directed graph; $V_0 \subset V$ is the subset of fixed vertices of V; $\omega^v$is a function that assigns element in set Ω to each vertex in set V; $n^{in}, n^{out}: V \mapsto \mathbb{N}_0$ are the functions that describe the indegree and outdegree of vertices in set V; C^{in}, C^{out}: are the sets of inlet and out connectors; E is a set of edges; π^c is the

function that assigns the thermodynamic properties (mass, pressure, and temperature) to every connector; f^{π} are the governing equations that assign outlet properties to the inlet properties of every vertex in set V.

It should be noted that $V_0 \coloneqq \emptyset$ when there is no restriction on specific components and the configuration of a thermodynamic cycle is a closed loop. By adopting this definition, the *T-graph* of a recompression cycle (Figure 3-4(a)) can be mathematically expressed as in the Appendix, and Figure 3-4 (b) expresses the connection in terms of the graph theory aspects.

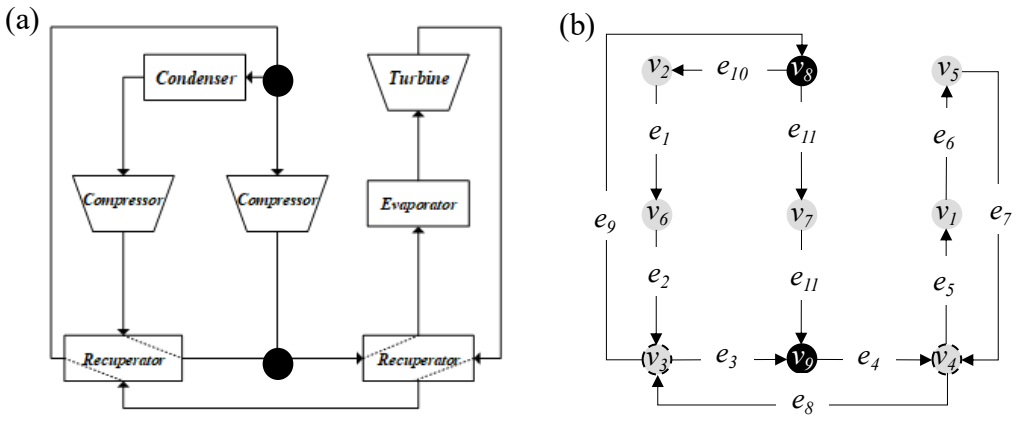

Figure 3-4. T-graph of recompression cycle: (a) original cycle; (b) T-graph representation

3.3 Configuration optimization framework

The configuration optimization framework consists of a thermodynamic graph (T-Graph), configuration solver, and optimizer. The overview of this framework is depicted in Figure 3-5. The configuration solver is responsible for solving nonlinear equations representing thermodynamic cycles in this framework. The T-Graph part bridges the gaps between the thermodynamic cycle and configuration representation.

Possible mutations of original configurations can be codified and implemented on a T-graph. Lastly, the optimizer tries to find the optimum design through iteration.

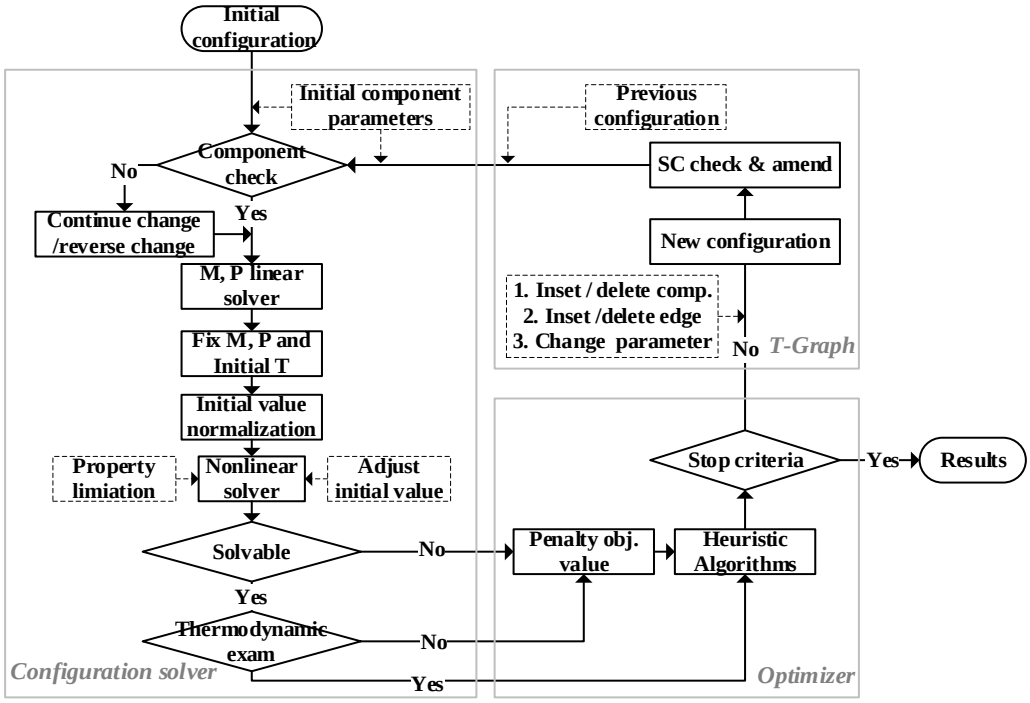

Figure 3-5. Configuration optimization framework

3.3.1 Configuration solver

A configuration solver should solve any given configuration and evaluate the system performance. The primary task is to provide sufficient governing equations representing a T-Graph and find an appropriate method to solve these nonlinear equations. Figure 3-6 shows the complete solution process of a given configuration and its parameter setting. It should be noticed that there is component check after designing a new system configuration. There might be insufficient component in a circuit of a cycle, such as compressor/turbine, this should be amended by inserting this missing type of component. And when designing a new system configuration through insert or performance variable (pressure ratio/expansion ratio) change, the pressure

ratio/expansion ratio might be less than 1, this turns a compressor to a turbine or a turbine to a compressor. The detail about these amendments will be illustrated in Section 3.3.3.2 and Section 3.3.3.3. The M, P here means mass and pressure, these two sets of variables are solved first and independently (from enthalpy) due to simplified model.

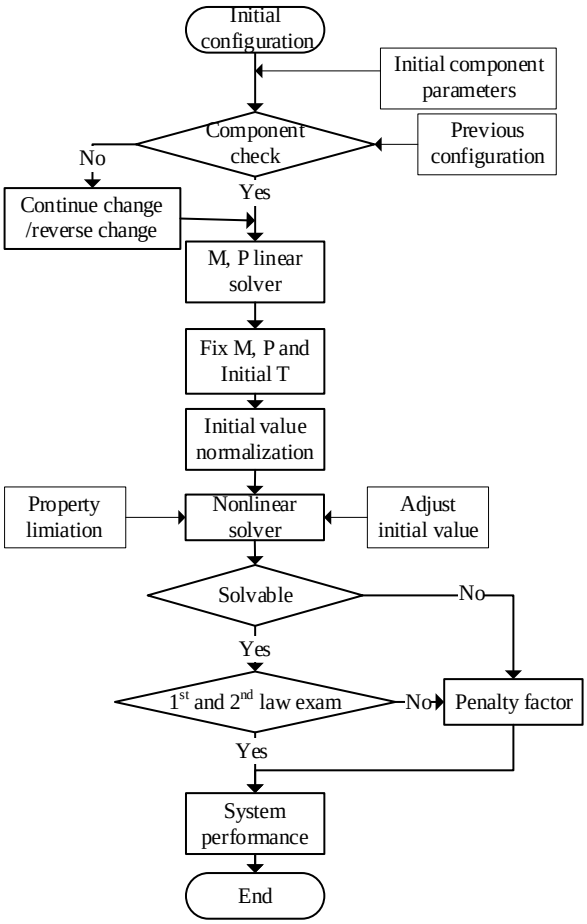

Figure 3-6. Configuration solver

The governing equations that map the input and output (component) are critical to solving the thermodynamic cycles. As defined in the T-graph, the total number of unknown variables imported by these different components can be calculated by indegree and outdegree of nodes, as expressed in Equation 3-4.

$$N_{tot}^{var} = 3 * (\sum n^{in}(v) + \sum n^{out}(v))/2, v \in \Omega \quad 3\text{-}4$$

An equivalent number of equations plus parameters should be provided to perform a steady-state evaluation of a T-graph. Taking S-CO_2 power generation as a case study, only limited component types were considered in T-graph, such as the component types of a recompression cycle in the last section. Each component except split/ merge has one indegree and one outdegree. Each split/merge has two/one outdegree and one/two indegree. Then, the total number of variables is expressed in Equation 3-5.

$$N_{tot}^{var} = 3 * (\sum N_{HX_{sr}} + \sum N_{HX_{sk}} + \sum N_{Turb} + \sum N_{Comp} + 2 * \sum N_{Re} + 3 * \sum N_{Split}) \quad 3\text{-}5$$

Heat transfer coefficient and pressure drop equations are used for the heat exchanger model, while compressors and turbines adopt a ten-coefficient model. Simplified component models are assumed to avoid struggling with convergence issues and focus on optimization. The equations that govern every component are explicitly expressed in Equation 3-6 through Equation 3-11. Merge and split are passive components that solely obey mass, momentum, and energy conservations laws. In the solution process, the Python-based package CoolProp (Bell et al. 2014) is used for property search represented by functions f_{PH}^{T}, f_{PT}^{H}, f_{PS}^{H}, f_{PH}^{S}. There are variables that need to be determined for solving a configuration governed by these equations. These variables are deemed constant when solving a configuration in this optimization work. They can also be optimized through the simulated annealing optimizer. These variables include effectiveness or efficiency (η), pressure ratio, expansion ratio (pr), mass split ratio (mr).

$$\begin{cases} m_{out}^{i_{ss}} = m_{in}^{i_{ss}};\ P_{out}^{i_{ss}} = P_{in}^{i_{ss}} \\ H_{out}^{i_{ss}} = f_{PT}^{H}\left(P_{out}^{i_{ss}}, T_{out}^{i_{ss}}\right) \\ \text{Sink: } T_{out}^{i_{ss}} = T_{in}^{i_{ss}} - \left(T_{in}^{i_{ss}} - T_{ss}^{i_{ss}}\right) * \eta^{i_{ss}} \\ \text{Source: } T_{out}^{i_{ss}} = T_{in}^{i_{ss}} + \left(T_{ss}^{i_{ss}} - T_{in}^{i_{ss}}\right) * \eta^{i_{ss}} \\ T_{in} = f_{PH}^{T}\left(P_{in}^{i_{ss}}, H_{in}^{i_{ss}}\right) \end{cases} \quad \text{for } i_{ss} \in \Omega(\text{Source, Sink}) \tag{3-6}$$

$$\begin{cases} \begin{cases} m_{hot,out}^{i_r} = m_{hot,in}^{i_r};\ P_{hot,out}^{i_r} = P_{hot,in}^{i_r} \\ H_{hot,out}^{i_r} = f_{PT}^{H}\left(P_{hot,out}^{i_r}, T_{hot,out}^{i_r}\right) \\ T_{hot,out}^{i_r} = T_{hot,in}^{i_r} - \left(T_{hot,in}^{i_r} - T_{cold,in}^{i_r}\right) * \eta^{i_r} \\ T_{in}^{i_r} = f_{PH}^{T}\left(P_{in}^{i_r}, H_{in}^{i_r}\right) \end{cases} \\ \begin{cases} m_{cold,out}^{i_r} = m_{cold,in}^{i_r};\ P_{cold,out}^{i_r} = P_{cold,in}^{i_r} \\ H_{cold,out}^{i_r} = (H_{hot,out}^{i_r} - H_{hot,in}^{i_r}) * m_{hot,in}^{i_r} / m_{cold,in}^{i_r} - H_{cold,in}^{i_r} \end{cases} \end{cases} \quad \text{for } i_r \in \Omega(\text{Recuperator}) \tag{3-7}$$

$$\begin{cases} m_{out}^{i_{ct}} = m_{in}^{i_{ct}};\ P_{out}^{i_{ct}} = P_{in}^{i_{ct}} * pr^{i_{ct}} \\ \text{Compressor: } H_{out}^{i_{ct}} = H_{in}^{i_{ct}} + \left(H_{iS,out}^{i_{ct}} - H_{in}^{i_{ct}}\right)/\eta^{i_{ct}} \\ \text{Turbine: } H_{out}^{i_{ct}} = H_{in}^{i_{ct}} + \left(H_{iS,out}^{i_{ct}} - H_{in}^{i_{ct}}\right) * \eta^{i_{ct}} \\ H_{iS,out}^{i_{ct}} = f_{PS}^{H}\left(P_{out}^{i_{ct}}, S_{in}^{i_{ct}}\right) \\ S_{in}^{i_{ct}} = f_{PH}^{S}\left(P_{in}^{i_{ct}}, H_{in}^{i_{ct}}\right) \end{cases} \quad \text{for } i_{ct} \in \Omega(\text{Compressor/Turbine}) \tag{3-8}$$

$$\begin{cases} m_{out}^{i_m} = m_{in_1}^{i_m} + m_{in_2}^{i_m} \\ P_{out}^{i_m} = P_{in_1}^{i_m};\ P_{out}^{i_m} = P_{in_2}^{i_m} \\ H_{out}^{i_m} = (H_{in_1}^{i_m} * m_{in_1}^{i_m} + H_{in_2}^{i_m} * m_{in_2}^{i_m}) / m_{out}^{i_m} \end{cases} \quad \text{for } i_m \in \Omega(\text{Merge}) \tag{3-9}$$

$$\begin{cases} m_{out_1}^{i_s} + m_{out_2}^{i_s} = m_{in}^{i_s}, m_{out_1}^{i_s} / m_{out_2}^{i_s} = mr \\ P_{out_1}^{i_s} = P_{in}^{i_s};\ P_{out_2}^{i_s} = P_{in}^{i_s} \\ H_{out_1}^{i_s} = H_{in}^{i_s};\ H_{out_2}^{i_s} = H_{in}^{i_s} \end{cases} \quad \text{for } i_s \in \Omega(\text{Split}) \tag{3-10}$$

$$m_{right}^{i_e} = m_{left}^{i_e};\ P_{right}^{i_e} = P_{left}^{i_e};\ H_{right}^{i_e} = H_{left}^{i_e}, \quad \text{for } i_e \in \Omega(\text{Edge}) \tag{3-11}$$

Until now, the total number of variables in these governing equations equals what is shown in Equation 3-12. Considering that split and merge are in pairs, it seems the number of equations equals the number of unknowns. However, the mass balance

equations are not independent, which means the matrix constituted by mass equations is rank-deficient. It is the same for pressure balance equations when no pressure drop in heat exchangers is assumed. Therefore, two additional equations are needed to make configurations solvable. By setting the minimum pressure in the system and fixing the mass flow rate of coolant at one heat sink, the equations required for solving any thermodynamic cycles are complete.

$$N_{tot}^{equ} = 3 * \sum N_{HX_{sr}} + 3 * \sum N_{HX_{sk}} + 3 * \sum N_{Turb} + 3 * \sum N_{Comp} + 6 * \sum N_{Re} + \quad 4 * \sum N_{Split} + 5 * \sum N_{Split} \tag{3-12}$$

As discussed before, the pressure drops in a heat exchanger and pipeline are not considered. And based on the equations listed in Equation 3-6 ~ Equation 3-11, the mass and pressure can be separated from enthalpy (temperature) if the pressure ratio and split ratio are given ahead. Therefore, a separated solver that solves mass and pressure first and then enthalpy (temperature) is designed. A linear equation solver from NumPy (Harris et al. 2020) is used for solving mass and pressure. A nonlinear multivariate equation solver that uses MINPACK's 'hybrd' and 'hybrj' routines from SciPy (Virtanen et al. 2020) is implemented for solving enthalpy (temperature).

Moreover, multiple techniques have been embedded in the solver to boost the success rate of solving thermodynamic cycles: 1) This configuration solver is embedded in the optimization process. The configuration change during optimization is limited, and the new configuration is within the neighbor of the previous design. In other words, the mutations designed in the optimization algorithm only bring minor change for the original design. The inlet and outlet state should be changed not much. Therefore, the new system design is similar with previous design before mutation and the states of

working fluid are close to the previous ones. Therefore, the previous iteration of configuration information (performance variables) can be used for the current iteration of the configuration's performance variables for fast convergence. In other words, the thermodynamic information from the previous configuration is used as the initial guess value for the new T-graph. 2) Based on the material limitation of the turbine and other components, the property boundary for any configurations is set (e.g., maximum pressure is 25 MPa, the maximum temperature is 500°C); 3) adjusting the guess value approach (Winkler, Aute, and Radermacher 2008) is implemented after the failure of the first trial. Overall, if maximum efficiency is the objective, the objective can be expressed as Equation 3-13. It should be noticed that the two sets of variables are directly reflected in the objective function, however, the two sets of variables, including configuration and performance variables, are concluded in constraints. The objective can only be calculated after connection (E) and performance variable (θ) are decided.

$$\begin{cases} E,\theta = \text{argmax}\dfrac{\sum_{c_i,c_i'} \pi^c(c_i') - \pi^c(c_i)\text{, for } c_i \in C^{out}, c_i' \in C^{in} | u(c_i) = u(c_i') \wedge \omega^v(u(c_i)) = \text{ Turb}}{\sum_{c_j,c_j'} \pi^c(c_j') - \pi^c(c_j)\text{, for } c_j \in C^{out}, c_j' \in C^{in} | u(c_j) = u(c_j') \wedge \omega^v\left(u(c_j)\right) = \text{ Sou}} \\ s.t.\ \ G(V,E,\theta) \in \text{strongly connected} \\ N_i^{low} \le D(i) \le N_i^{up};\ \text{for } i \in \Omega \\ \theta_i^{low} \le \theta_i \le \theta_i^{up}, \text{for } i \in \Omega \end{cases} \quad 3\text{-}13$$

Finally, for recuperators in S-CO2 systems, the pinch point may exist in the middle of the heat exchange due to inappropriate design. Pinch point temperature is examined after solving the configuration, and a minimum value for recuperators is set to be 5 K (Dostal, Hejzlar, and Driscoll 2006, 2) to eliminate this issue. A penalty factor is placed on the objective value if the thermodynamic laws, including 1st and 2nd laws, are violated. This violation might be inappropriate performance variables that result in energy imbalance, negative power cycle.

3.3.2 Constraints of configuration

3.3.2.1 Graph property constraints

A graph (configuration) is strongly connected (SC) if every vertex (component) is reachable from every other vertex (component). This definition can be interpreted as mass conservation and steady-state condition corresponding to the energy conversion system. Therefore, the graph constituted by the configuration of the energy conversion system should be an SC graph no matter how researchers intend to design the configuration of a thermal system. The natural question is how to guarantee the newly generated configuration is an SC graph. The T-graph is further simplified to examine the SC property.

The recuperator in configuration is decoupled as a heating and cooling part to simplify the graph structure. Then, the T-Graph in Figure 3-4 (b) can be disassemble to Figure 3-7 (a), where only two-edge vertices ($n^{in}(v) = 1, n^{out}(v) = 1$) and three-edge vertices ($n^{in}(v) + n^{out}(v) = 3$) exist. Then, the decoupled T-Graph is disassembled by two-edge vertices and three-edged vertices. It is easy to find that the deletion or insertion two-edge vertices will not influence the flow direction or the stream of working fluid. Therefore, two-edge vertices do not influence the SC property, no matter how many of these vertices are embedded in the graph. Last, a disassemble T-graph like Figure 3-7 (b) can represent the original graph without losing the SC property. With the above configuration representation, and analysis of the decoupled T-graph, an SC graph can be manually permuted with two steps: first, make a connection within only splits/merges (three-edge vertices) of a system without

considering other components (two-edge vertices), second, insert other components (two-edge vertices). This process will guarantee the SC property and any configuration can be reached.

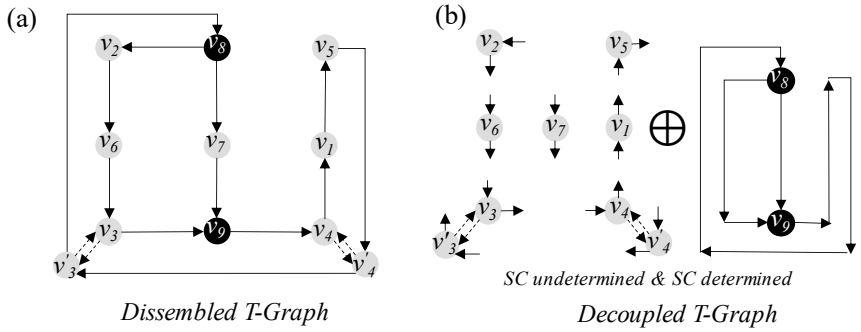

Figure 3-7. Simplification of T-Graph: a) dissemble version of T-graph; b) decoupled T-graph

3.3.2.2 Thermodynamic property constraints

From a thermodynamic viewpoint, the cycle in the T-Graph should comply with mass, momentum, and energy conservations. It might be determined whether thermodynamic laws are violated when the solver cannot converge. However, manually screening them would be trivial and waste many computational resources. Therefore, a few general and necessary conditions should be satisfied for any configuration if one wants these conservation rules to be satisfied. T-S diagram can be used to explain the thermodynamic rules, 1st and 2nd law. First, the energy conservation for closed loop can be interpreted as closed loop in T-S diagram (1st law). Second, the compressor or turbine must have irreversible compression or expansion process, which means entropy increase across these two components. Take the S-CO_2 cycle as an example, as shown in Figure 3-8. There must be incline compression or expansion process. To fulfill a closed loop in T-S diagram, 1) there must be a compressor and turbine in a loop for circulating working fluid, and 2) a nominal sink is required to complete the

thermodynamic loop. Here, the 'sink' might not be a heat exchanger. Any method that results in a temperature decrease can be viewed as a sink. Such requirements in each loop of thermodynamic cycle should be satisfied to fulfill the thermodynamic rules when implementing mutation in optimization. It should be noted that only these three components in a configuration result in a negative efficiency due to net power input. This type of configuration might be a refrigeration system and would be phased out by the optimization algorithm eventually. With these two minimum requirements, energy conservation is satisfied. The other two conditions are automatically satisfied due to SC property. Therefore, the influenced loop in the configuration during new configuration generating process is examined. The supplementary components are inserted into the loop when configuration (connection) changes.

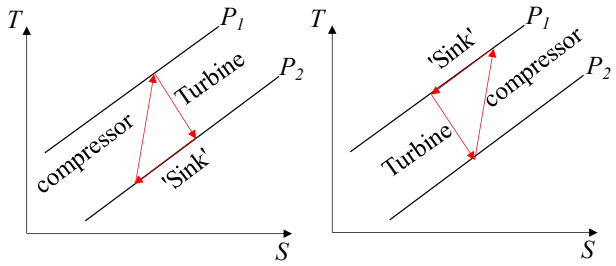

Figure 3-8. The necessary condition of loops in T-Graph

3.3.3 Configuration optimizer

3.3.3.1 Simulated annealing algorithm

Since the feasible region mixes discrete and continuous variables, metaheuristic algorithms would be preferred for optimum search. There are other algorithms that can be used to solve mixed discrete and continuous variables, such as branch and cut based methods. These algorithms are hard to implement the strongly connected and thermodynamic rules to screen out candidates. Therefore, the exact algorithms will spend much time evaluating configuration that does not obey SC and thermodynamic

laws. And assigning objective values for these violated configurations will be a problem if one wants to measure the quality of the design. In addition, a second-level optimization will still be needed to optimize the component performance variables, which will slow down the optimization process further. A simulated annealing algorithm is implemented due to robustness and effectiveness (Nikolaev and Jacobson 2010). Kirkpatrick et al. developed this algorithm in 1983 based on the metal annealing process (Kirkpatrick, Gelatt, and Vecchi 1983). It is a probabilistic technique for approximating the global optimum of a given function. It is good at exploring the solution space. It has good characteristics of finding the global optimum even though it often converges very slowly in practice due to the lack of crossover operator in the algorithm.

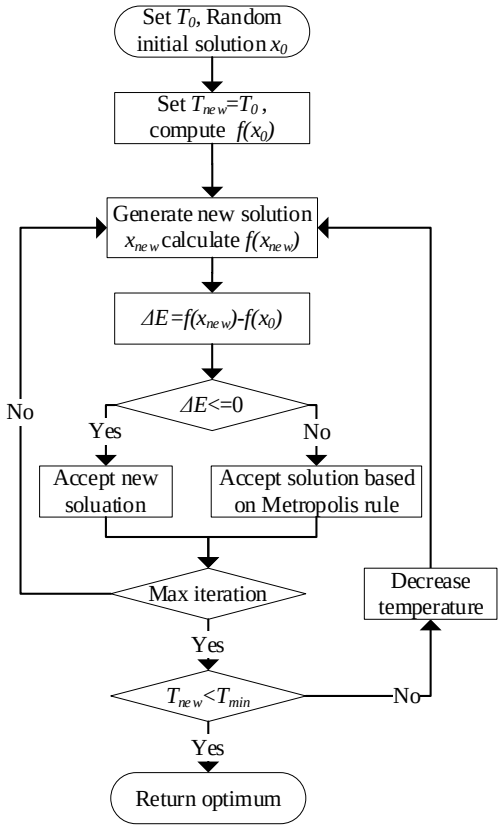

Figure 3-9. Simulated annealing algorithm

Figure 3-9 shows the optimization process of a simulated annealing algorithm. The algorithm first starts from an initial temperature (T_0), candidate solution (x_0) and computes the initial objective value ($f(x_0)$). Then a random disturbance is added for generating a new solution (x_{new}) in the neighbor of current one, and the energy difference ($f(x_0) - f(x_{new})$) between the new solution and the original solution is obtained. The new solution is adopted as the current solution for a minimization problem if the difference is less than 0. On the contrary, a probability proportional to the energy difference is used to determine whether to accept or reject the new solution. A lower temperature indicates a smaller probability of acceptance, and a greater difference in objective value means a smaller probability of acceptance. The probability of acceptance is calculated based on Equation 3-14. Here ΔE is the energy difference between the new solution and the original solution, and T_{new} is current temperature. The temperature gradually decreases with the number of iterations (t), which can be expressed in Equation 3-15. As a result, the probability of obtaining a poor solution is small. The loop continues until the temperature is lower than the tolerance value (T_{min}). Before the temperature is lowered to the minimum, a better solution is found through multiple iterations at the current temperature.

$$p = exp(-\Delta E / T_{new}) \tag{3-14}$$

$$\begin{cases} \text{Fast: } T_{new} = T_0/(1+t) \\ \text{Cauchy: } T_{new} = T_0 * \eta \\ \text{Boltzmann: } T_{new} = T_0/log(1+t) \end{cases} \tag{3-15}$$

3.3.3.2 Mutation of configuration

The new configuration update is the core part of the whole optimization process. The maximized range of the feasible region is maintained by applying specific updating

rules. In other words, the updated configuration should be any possible configurations that do not violate the constraints at the topological and thermodynamic levels. All the possible updates from the mutation perspective can be categorized as performance variables changes of component, insertion, or deletion components in the T-Graph. Since performance variables changes are straightforward and do not impact topological properties, only the deletion and insertion of components in the T-Graph needs to be discussed.

As described in the last subsection, various disturbances generate a new configuration, such as insertion/deletion of three-edge or two-edge vertex. This subsection will discuss how constraints influence the generation of new designs from both topological and thermodynamic aspects. It has been explained in Section 3.3.2.1 that the two-edge vertices would not impact the SC property. Therefore, any deletion or insertion of two-edge vertices into the T-Graph will not be constrained. The only need is to find a method that does not violate the SC property for insertion or deletion of three edge vertex. It should be noted that the slightest change should be made for a T-Graph if multiple procedures can be executed when practicing new configuration generation.

It is apparent that split and merge (three-edge vertices) appear in pairs, and the insertion of three-edged vertices will not influence the SC property either. The exceptional one that required special care is the deletion of three-edged vertices. Deleting the same type of three-edged vertex will not be considered since it will result incomplete system configuration. Moreover, there must be a direct edge between a pair of split and merge (without other splits or merges within this edge) for a strongly connected T-Graph with

at least one pair of split and merge. Considering a general case shown in Figure 3-10. For the condition of split pointing to merge, since any new configuration remains an SC graph, the connectivity between vertex (v_1) and vertex (v_4) determine the new configuration as I or II. The new configuration should be the least changed type I (without crossover) if there is more than one path from the vertex (v_1) to vertex (v_4). Otherwise, the new configuration should be a crossover type II. For the condition of merge pointing to split, the direct paths between vertices before merge (v_1, v_2) and vertices after split (v_3, v_4) should be found and matched. The new configuration should be a crossover type III. Moreover, induction can be proved by any configurations that can reach any configurations through multiple insertions or deletions. Until now, I can freely make any deletion or insertion to generate any new configuration from the graphical perspective, even though the number of splits or merges is limited in practice.

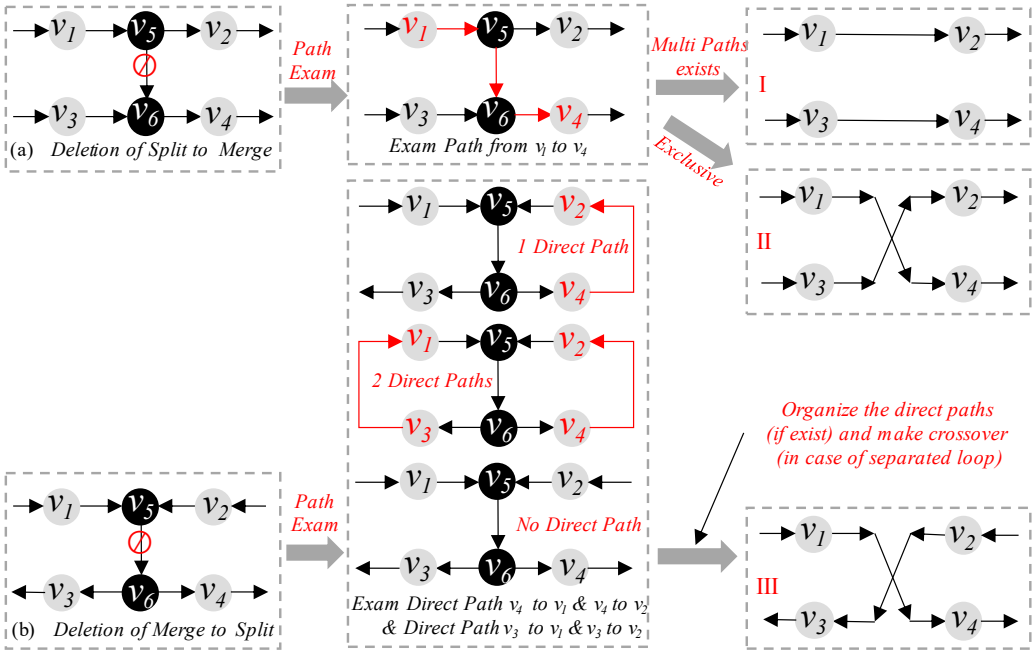

Figure 3-10. Deletion for pair of three-edged vertices: (a) deletion for pair of split-merge; (b) deletion for pair of merge-split

Figure 3-11 is an example of one-step insertion and deletion of an edge to a bottom composite S-CO_2 power cycle (Crespi et al. 2017). There are various insertions and deletions for different types of components. Based on the edge number and component information of a bottom composite S-CO_2 cycle, the total number of new configurations is listed in Table 3-1. However, some insertions or deletions require additional components which satisfy thermodynamic constraints.

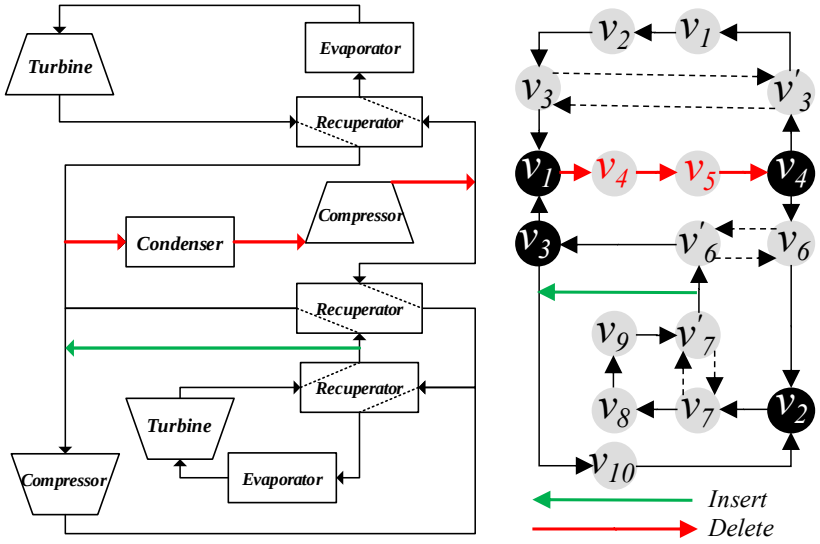

Figure 3-11. Example of edge insertion or deletion on bottom composite cycl-e

Table 3-1. New configuration number based on a bottom composite cycle

Component	Insertion	Deletion
Evaporator	14	2
Condenser	14	1
Compressor	14	2
Turbine	14	2
Recuperator	364	3
Split and Merge	364	6

3.3.3.3 Mutation of performance variables

Contrary to an explicated designed configuration, it is not straightforward to determine which pressure component(s) should be changed after an initial pressure ratio change

from the T-Graph perspective, especially for two splits/merges configurations. The same amount of pressure change is assumed for computational efficiency when manipulating the pressure/expansion ratio. For instance, two compressors have a pressure ratio (PR) of 2, and one turbine has an expansion ratio (ER) of 4 in a loop. If the ER increases from 4 to 6, then only one of the compressors increases the PR from 2 to 3. There are many choices to change other pressure components to make pressure balance when initializing a change for one pressure component. For instance, there is a system configuration shown as Figure 3 (other components that do not influence pressure are excluded). The pressure ratio or expansion ratio of components are listed in bracket in the figure.

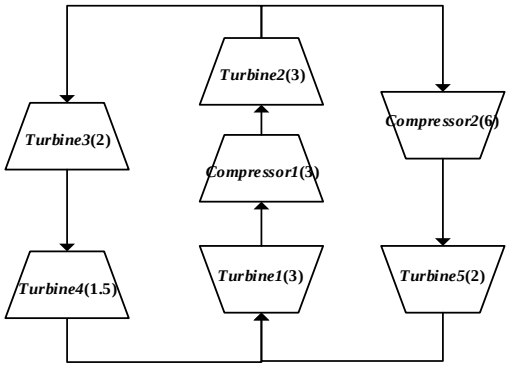

Figure 3-12. Configuration without non-pressure components

When change the pressure ratio of compressor1 from 3 to 6, we can have following choice: 1) change expansion ratio of turbine1/turbine2 from 3 to 6; or 2) change expansion ratio of turbine5 and turbine3 from 2 to 4; or 3) change pressure ratio of compressor2 from 6 to 3 and turbine3 from 2 to 4, etc. There are many choices we can make the pressure balance. The first choice only needs to change one component whereas the second and third choices need to change two components. The first change option will be preferred considering minimized change is brought to the system design.

A composite configuration from Figure 3-13 is used as an example to illustrate this problem to clarify the challenging aspects. First, the pressure components and loops are identified in Table 3-2. Any loops should contain an even number of changed pressure components based on the pressure/expansion ratio change rule mentioned above. And the minimum number of changes should be achieved simultaneously to make the change as small as possible. In that case, the possibility of excluding candidate design is low. Based on these criteria, the pressure ratio of V10 should be changed if vertice V9 is changed, while V5, V9, and V11 should change if vertice V2 is altered. A bi-level integer linear programming problem determines which pressure component(s) should be changed.

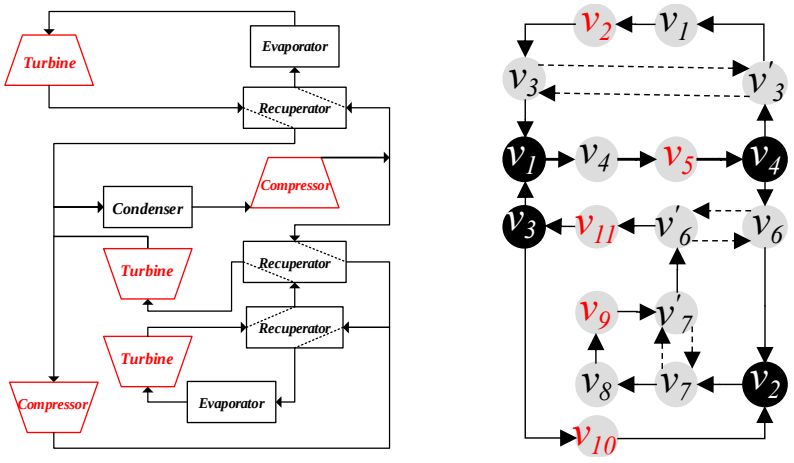

Figure 3-13. Composite configuration and T-Graph representation

Table 3-2. Pressure components exist in loops of a composite configuration

Pressure component	Loop 1	Loop 2	Loop 3
V_2	Yes	No	No
V_5	Yes	Yes	No
V_9	No	Yes	Yes
V_{10}	No	No	Yes
V_{11}	No	Yes	Yes

The formal and general expression of the integer programming problem that determines the changed components can be represented by Equation 3-16:

$$\begin{cases} \min \sum_{L_i} \left(c_{L_i} + t_{L_i}\right), \ for\ L_i \in T - Graph \\ s.t. \sum_{L_i} (c_{L_i} + t_{L_i} - 2 * N_{L_i}) = 0, for L_i \in T - Graph \\ \text{int}\left[\frac{1}{2}\sum_{L_i} \left(\bar{c}_{L_i} + \bar{t}_{L_i}\right)\right] + 1 \leq N_{L_i},\ \bar{c}, \bar{t}\text{: initial change} \\ c_{L_i}, t_{L_i} \in \text{Binary}, N_{L_i} \in \text{integer} \\ c_{L_i}, t_{L_i} = \begin{cases} 1, \text{if PR/ER changed} \\ 0, \text{if PR/ER not changed} \end{cases} \end{cases} \quad 3\text{-}16$$

Where $c_{L_i}$and t_{L_i} are binary variables representing whether to make the pressure ratio change for compressor and turbine in a loop L_i. The objective of the above integer programming is to minimize the change of the whole configuration due to the change of initial PR/ER change of compressors or turbines. The constraint in the problem can be interpreted as the number of changes in one loop should be even. It is necessary to determine the following change type (increase/decrease) based on the original change type in the changed components. The mathematical expression for the change type can be represented in Equation 3-17, where c'_{L_i} and t'_{L_i} are integral variables (1 for increase or -1 for decrease) represent pressure components determined to change in Equation 3-16.

$$\begin{cases} \sum_{L_i} \left(c'_{L_i} - t'_{L_i}\right) = 0, c'_{L_i}, t'_{L_i}\text{: changed components} \\ c'_{L_i}, t'_{L_i} = \begin{cases} 1, \text{if PR/ER increases} \\ -1, \text{if PR/ER decreases} \end{cases} \end{cases} \quad 3\text{-}17$$

Then, the changed components from the first level will be transferred into the second level of integral programming problem as Equation 3-18 shows. The component

change type (increase/decrease) is determined by solving this integer optimization problem.

$$\begin{cases} \max \sum_{L_i} \left(c'_{L_i} + t'_{L_i}\right), \ for\ L_i \in T - Graph \\ s.t. \sum_{L_i} (c'_{L_i} - t'_{L_i}) = 0, for L_i \in T - Graph \\ c'_{L_i}, t'_{L_i} = \begin{cases} 1, \text{if PR/ER increases} \\ -1, \text{if PR/ER decreases} \end{cases} \end{cases} \quad 3\text{-}18$$

3.3.3.4 Codification of mutation

For any configuration, the topological information is a T-Graph. Any insertion or deletion will import additional state variables that determine the pressure, mass and enthalpy of fluid. New nodes and edges carry these new variables, and they need to be codified into the original design. Figure 3-14 shows how insertion and deletion influence the T-Graph of an initial configuration. The edge inserted will move backward, and every insertion imports one vertex and one additional edge. The edge connected to the deleted vertex outlet will be removed together with the vertex. The inlet of the deleted vertex will be kept and directly connected with the forwarding vertex. Deletion or insertion will be easily codified with this consistent and modularized deletion or insertion process.

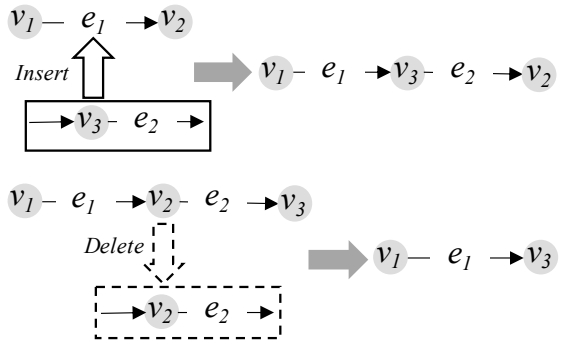

Figure 3-14. Codification of insertion and deletion on T-Graph

3.4 Implementation of configuration optimization

3.4.1 Parameter settings

The whole optimization platform is built in a Python environment with the help of the property library from CoolProp (Bell et al. 2014) and a nonlinear solver package from SciPy (Virtanen et al. 2020). Two sample cases under limited component number constraints to are examined validate the proposed optimization framework. Necessary parameters are required before conducting the evaluation. Based on the design reference and limitation from (Dostal, Hejzlar, and Driscoll 2006). The maximum temperature and minimum temperatures in the S-CO_2 power system are set to 850 K and 320 K. The maximum and minimum pressures in the S-CO_2 power system are 32 MPa and 8 MPa. The rest of the performance variables of every component are listed in Table 3-3, including the range and the initial value of these variables in governing Equation 3-6 and Equation 3-11.

Table 3-3. Performance variable of components

Component name	Performance variable	Range	Initial value
Heater	Source temperature, $T_{sr}[K]$	850 (fixed)	850
	Approach temperature, T_{ap} $[K]$	[10, 100]	50
Cooler	Sink temperature, T_{sk} $[K]$	310 (fixed)	310
	Approach temperature, $T_{ap}[K]$	[-10, -50]	-30
Compressor	Pressure ratio, pr[-]	[1.1, 4.0]	1.2
	Isentropic efficiency, η [%]	[10, 90]	80
Turbine	Expansion ratio, er[-]	[1.1, 4.0]	1.2
	Isentropic efficiency, η[%]	[10, 90]	80
Recuperator	Effectiveness, η[%]	[10, 90]	80
Split	Split ratio, sr[-]	[0.05, 20]	4

The parameter setting of the SA algorithm is listed in Table 3-4. Since SA is a probability-based optimization algorithm, multiple variations can be implemented for

an existing configuration. Designing the probability of different mutation type is one of the key parts of the optimization process. The mutation happens within the deletion or insertion of components and changes the components' performance variables. The probability should obey Equation 3-19. In this study, the detailed probability distribution can be viewed in Figure 3-15. It should be noted that when changing performance variable of split/merge, there is only one choice – split ratio (eta). Suppose the minimum or the maximum number of each component type is satisfied. In that case, the probability of deletion or insertion should be 0, respectively.

Table 3-4. Parameter setting of the SA algorithm

Terminated temperature (T_{min})	0.1
Initial temperature (T_0)	50
Internal iteration time (k)	100
Penalty factor	299

$$\begin{cases} \sum_i p_i = 1\,, \quad i \in \{\text{insert, delete, change}\} \\ \sum_v p_i^v = 1, \quad v \in \{\text{HX, Comp, Turb, Recu, Split/Merge}\} \\ p^v_{delete} = 0, n(v) < n(v)_{min} \; and \; p^v_{insert} = 0, n(v) > n(v)_{max} \\ \sum_v p^{v_j}_{change} = 1, v \in \{\text{HX, Comp, Turb, Recu, Split/Merge}\}, j \in \{eta, pr, er\} \end{cases} \quad 3\text{-}19$$

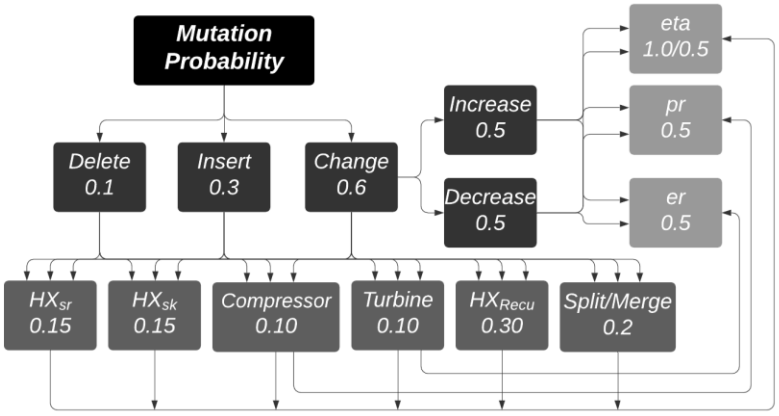

Figure 3-15. Probability distribution of mutation types

Table 3-5. Component number limitation for two case studies

Component	Maximum # for no split/merge	Maximum # for one split/merge	Minimum #	Initial #
Evaporator	1	1	1	1
Condenser	1	1	1	1
Compressor	1	2	1	1
Turbine	1	1	1	1
Recuperator	1	1	0	0
Split	0	1	0	0

The total number of each component type is restricted, which is displayed in Table 3-5, to investigate the capability of the thermodynamic solver. The initial configuration is a four-component-based system that includes a compressor, turbine, evaporator, and condenser. Two case studies that correspond to no split/merge condition and one split/merge condition are carried out to prove this optimization framework's capability.

3.4.2 Validations of configuration optimizer

There is no split/merge in the configuration in the first case study. The component types include one more recuperator. The three types of mutation, including deletion, insertion, and performance variables change, can be implemented under the limitation of maximum and minimum component numbers. It is viable to permutate all possible configurations under the limited searching domain and do an exhaustive search for all the configurations. Therefore, this case study's purpose is mainly to demonstrate the capability of the configuration solver. Since the number of feasible configurations is limited, the optimization result from the SA-based optimizer can be verified by the result of an exhaustive search among all the feasible configurations.

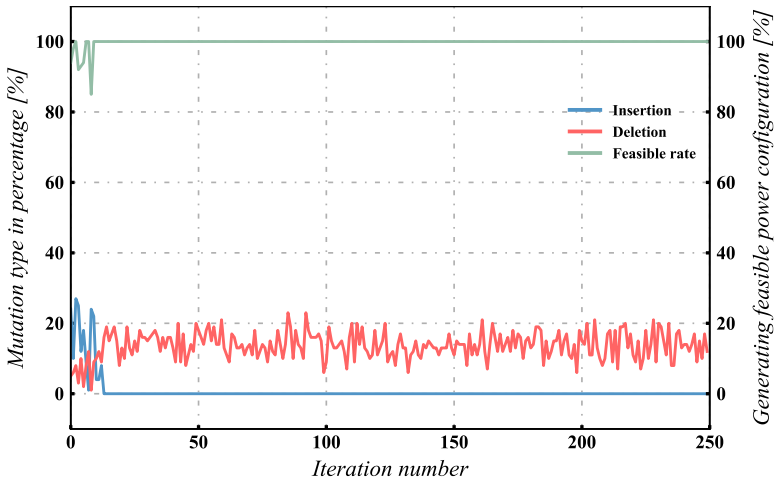

Figure 3-16. Rate of feasible configuration and implemented mutation

Figure 3-16 shows the generation rate of feasible configurations for every temperature drop of SA and the mutation type implemented by the SA algorithm in a single execution. The configuration solver has a high convergence rate for all configurations in this low volume of the feasible region. Here, the type of design that violates thermodynamic laws and assign penalty factor is called infeasible design. The rest of design is feasible and divided by total design will produce rate of feasible configuration The feasible rate shows below 100% at the beginning optimization process due to the inappropriate performance variables setting of components resulting in the second law violation condition. The deletion type is as low as 5% at the beginning since the initial configuration already reaches the minimum number restriction. After iteration 15, the optimizer mainly implements the performance variables and deletion mutation since the objective value reaches a relative value and reaches the maximum number restriction for this simple case study. The rest of the iterations can be viewed as purely tuning of performance variables.

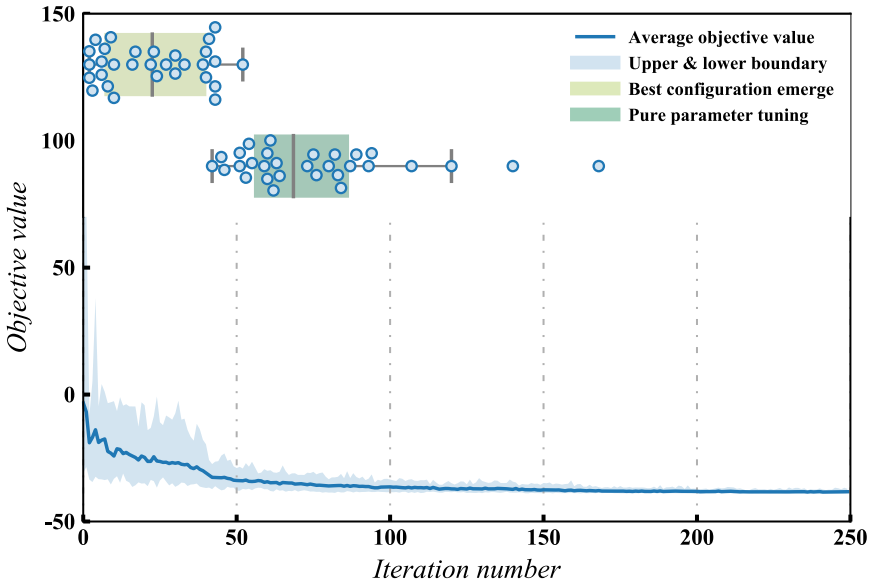

Figure 3-17. Converge process of configuration without split/merge over multiple times

Considering the stochastic characteristic of the SA algorithm, 30 times of independent execution of the optimization framework has been carried out to investigate the optimizer's stability. Figure 3-17 shows the convergence process of the objective value and the earliest occurrence of the final configuration. The maximum and minimum value of the objective value in every iteration is shown as a cloud. In contrast, the mean objective value in each iteration is plotted as a solid line in the middle. At the beginning of the iteration, the performance variables might be set inappropriately to yield an unsolvable or negative efficiency. Therefore, the objective value would be a large positive value. The narrowing down of the area and iteration validate this optimization platform's stability. The scatter points in the graph highlighted the emerging best configuration and purely performance variables tuning in each framework's execution. A statistical summary of the emerging and tuning information is also depicted as a box plot. With no split/merge restriction, the best configuration emerges primarily before iteration #50. The performance variables tuning happened before iteration #100.

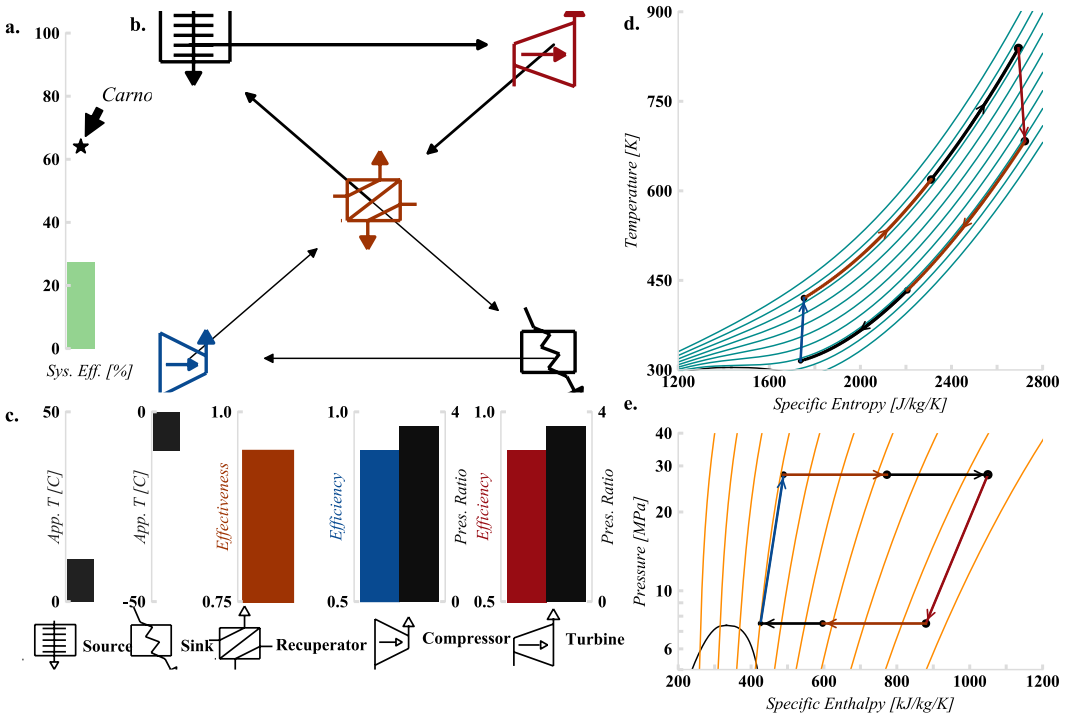

Figure 3-18. Best configuration without split/merge: (a) System efficiency, (b) System configuration, (c) Performance variables of every component, (d) T-s diagram of system, (e) P-h diagram of system

Figure 3-18 shows the result of the optimization. The best system efficiency reaches 39%, while the Carnot efficiency (the theoretical maximum efficiency one can get when the heat engine is operating between two temperatures) shown in subfigure (a) is 63%. The final configuration is depicted in subfigure (b), the additional recuperator is inserted at outlet of turbine and compressor. The optimized performance variables of every component in the system as the bar plot in subfigure (c) with maximum and minimum value set in Table 3-3, The T-s diagram shows isobaric line of heat exchange process in subfigure (d) and P-h diagram shows the isentropic line in subfigure (e). The The configuration is referred to as the recuperated S-CO_2 cycle. It is observed from subfigure (c) that the approach temperature of the sink/source reaches the lower bound of settings, and the effectiveness of the recuperator reaches the upper bound of settings.

The efficiency of the compressor and the turbine reaches the upper boundary of variable range settings, while the pressure ratio and expansion ratio do not reach the upper boundary of settings.

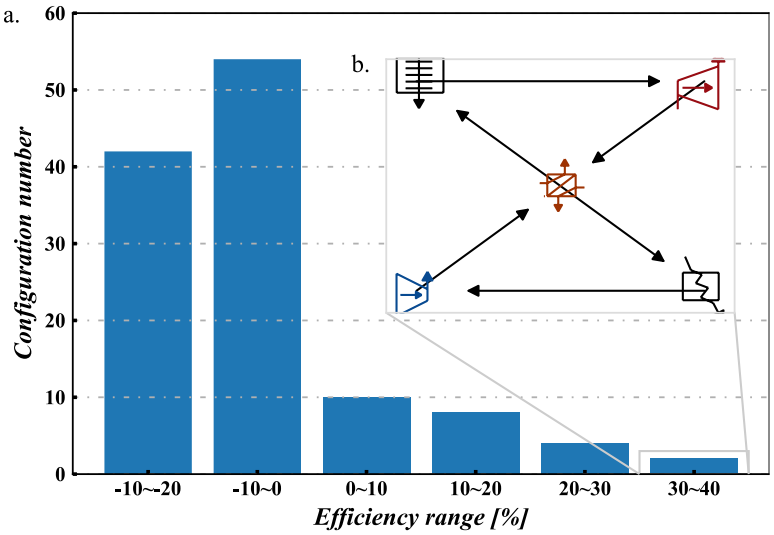

Figure 3-19. Exhaustive search for configuration without split/merge: (a) Statistical histogram of efficiency on feasible region, (b) Best configuration through the exhaustive search

To further prove the effectiveness of the whole framework, an exhaustive search was implemented in this simple case study. Results are shown in Figure 3-19 that the total number of configurations was 120 for a system with six components without a split. Therefore, I permutated all the 120 configurations constituting of these six components, executed only the performance variables tuning, and obtained the efficiency of all the configurations. The statistical results of all configurations are summarized in Figure 3-19. A negative efficiency indicates that the system consumes more electricity than it generates. Furthermore, the number of configurations decreases with the increase in efficiency level. The best configuration is a recuperated system that achieves 39% of thermal efficiency, which is the same as the one optimized by the framework.

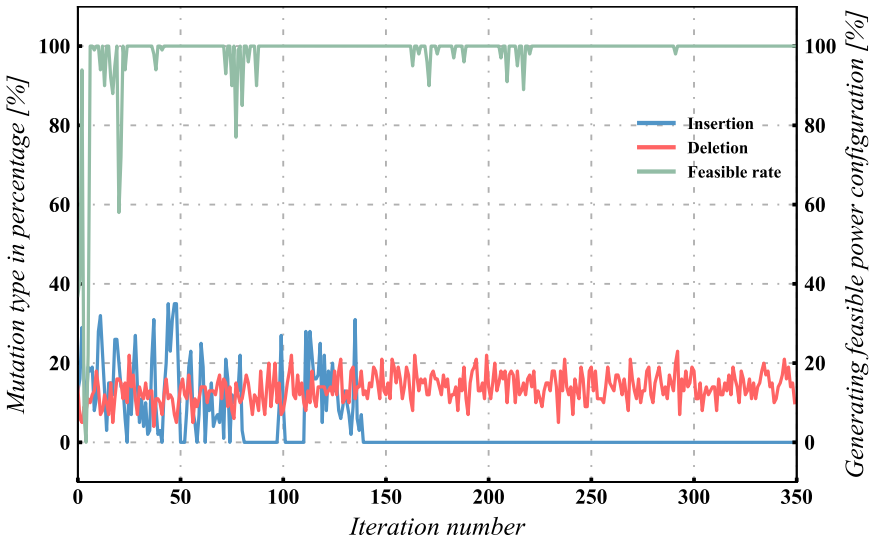

Figure 3-20. The generation rate of feasible configuration and accumulated configurations

The maximum number of different components is expanded with one split/merge to validate the framework on a more complex feasible region, as shown in Table 3-5. The generation rate of feasible configuration and the mutation type percentage change and iteration can be viewed in Figure 3-20. The generation rate of feasible configurations is lower than the limitation without split/merge due to more complex mutation types and more considerable diversity. However, the overall rate of feasible configuration is still over 97%. Similar to the simple case study, the mutation type insertion stops when the iteration number is over 140, which indicates the system reaches the maximum number limitation of every component and starts performance variables tunning purely.

The optimization process's stability is validated by 30 times of independent execution of the optimization framework and depicted as the cloud region in Figure 3-21. Compared with the simple case, the convergence process is slower but still reveals the stability of this optimization platform after 350 iterations. Equivalently, the statistical information was highlighted as box plots for the emergence of a best configuration and

purely performance variables tuning in each execution. With one split/merge restriction, the best configuration emerges mostly before iteration #75. Furthermore, the performance variables tuning happened before iteration #175. These two metrics suggest that it is harder to find optimality when the feasible region expands.

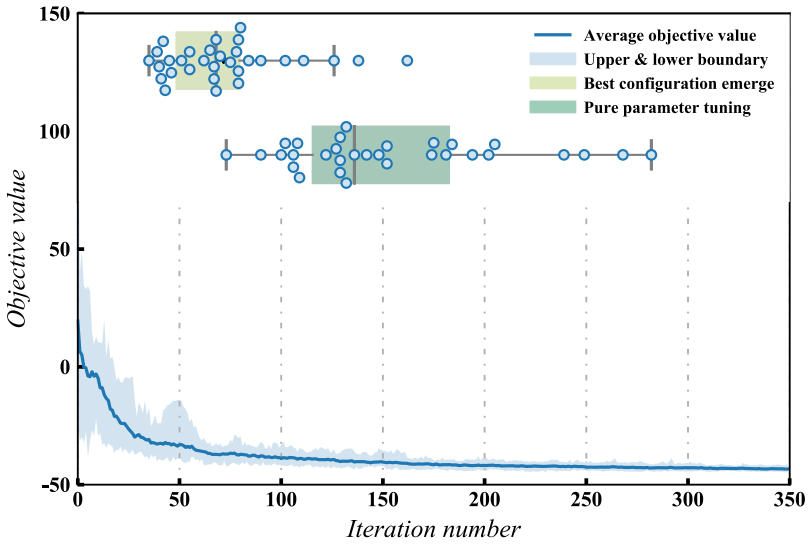

Figure 3-21. Convergence process of configuration with one split/merge over multiple times

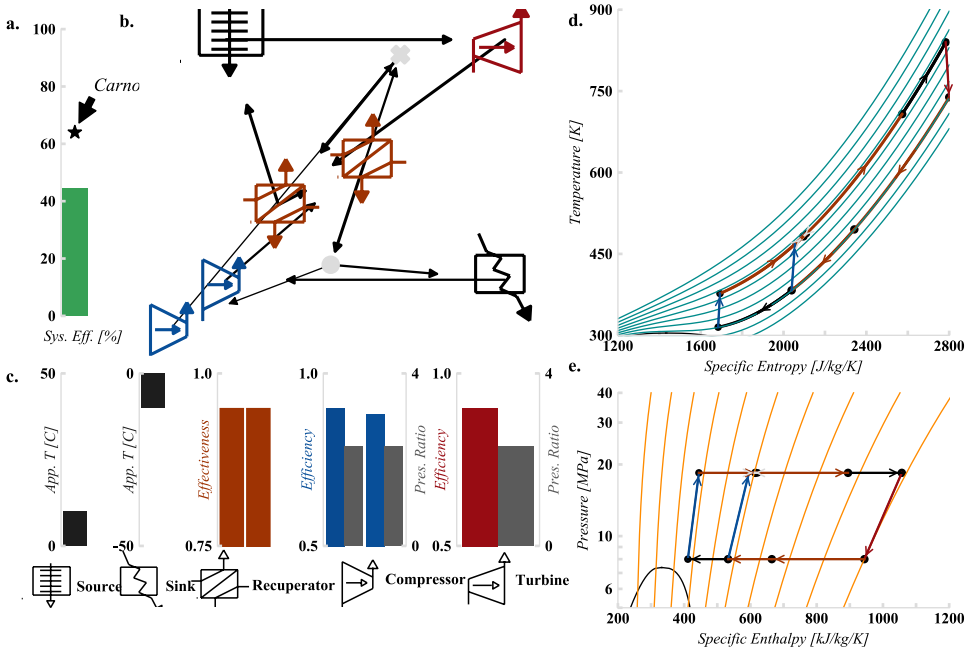

Figure 3-22. Best configuration with one split/merge: (a) System efficiency, (b) System configuration, (c) Performance variables of every component, (d) T-s diagram of system, (e) P-h diagram of system

Figure 3-22 shows the result of a single execution of the optimization process for one split/merge case study. The configuration converges to a recompression cycle with an overall efficiency of 44%. Even though exhaust searching on the case study is impossible, as discussed in Figure 3-19, the recompression cycle has been reported as the most efficient among various advanced configurations. The system design is limited by component number constraints for different application areas based on the state-of-the-art S-CO_2 power generation system. It has been applied in multiple areas, such as nuclear power generation (Dostal, Hejzlar, and Driscoll 2006; Ahn et al. 2015), solar power (Turchi et al. 2012; Neises and Turchi 2014), molten carbonate fuel cell system (Bae et al. 2014), and numerous research works, including literature reviews (Crespi et al. 2017; M.-J. Li et al. 2017), book chapters (Brun, Friedman, and Dennis 2017) and scientific report of National Laboratory (S. Wright, Thomas, and Gary 2011). The approach temperature of the heat source/sink reaches the minimum value, and the effectiveness of the two recuperators reaches the maximum value. The efficiency of the compressor and the turbine approach the upper boundary while the pressure ratio and expansion ratio are optimized to an appropriate value. An animation of the converging process for this case study can be viewed in (Gao 2020). Moreover, it is noted that the mass flow rate and other heat exchanger performance variables are tuned to make the outlet of the compressor and recuperator to the same temperature. Therefore, the irreversible loss would be minimized.

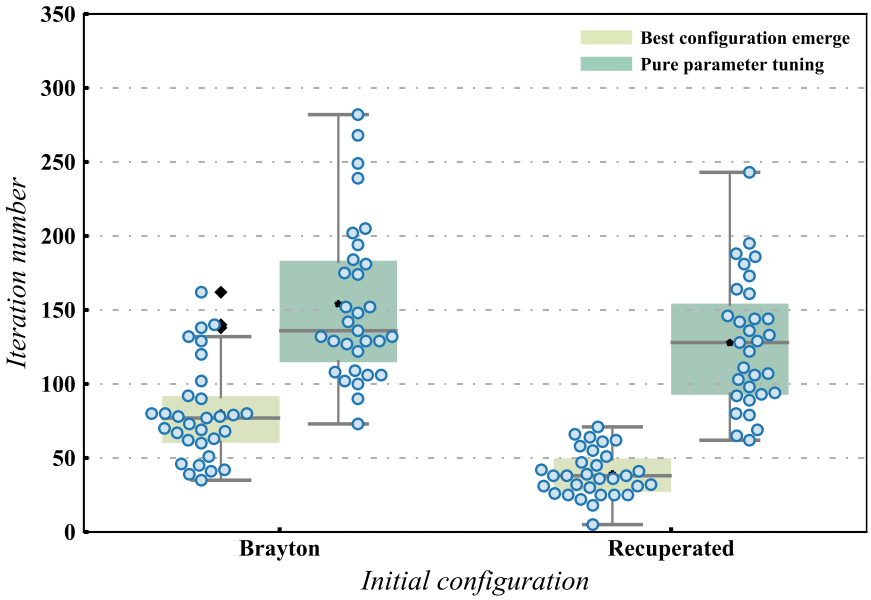

Figure 3-23. Influence of initial configuration

The initial configuration should impact the convergence of the optimization process. To investigate this effect, I compared optimization performance from a Brayton cycle and recuperated cycle with independent 30 executions for each of them. The statistical results are plotted in Figure 3-23. It is seen that the best configuration emerges earlier from an advanced design (recuperated cycle) than the basic configuration (Brayton cycle). Moreover, the fine-tuning process has limited improvement after switching to an advanced configuration. However, initiating with an advanced configuration helps the emergence of optimum configuration. Therefore, it is better to start from the existing optimized configuration for a more advanced configuration. Besides, the computational efficiency would be boosted if the optimizer could start from less advanced configurations and start from them.

3.4.3 Optimized configurations

In the first category, there is no split/merge in the configuration. Figure 3-24 shows the final optimized configuration based on this category. The system contains three

reheating, three recuperating, and two compression processes. It should be noticed that Figure 3-24 (b) plots the decoupled connections as discussed in Figure 3-7, where recuperators are separately plotted as dependent heating and cooling parts. And following figures for the configurations plot are all based on decoupled connection. The minimum pressure in the system is 7.5 MPa, and optimized thermal efficiency reaches 44.6% for this no split/merge configuration. Since there is no split/merge, the optimization of this configuration aims to align the relative location of compressors, turbines, and heat exchangers. Based on the result, the optimized configuration tends to use as many reheating processes as possible to elevate the average absorbing temperature, thus promoting theoretical efficiency. At the same time, all three recuperator processes are utilized to recover as much heat as possible from the outlet of turbines. Unlike the conventional recuperating process, the third recuperator recovers cooling capacity between two compression processes to lower the cooler's inlet temperature. Thus, the basic recuperated configuration decreases the average heat-dissipating temperature and increases theoretical efficiency. The animation of the optimization process (Gao 2020) shows the convergence process towards recompression system and pure performance parameter tuning.

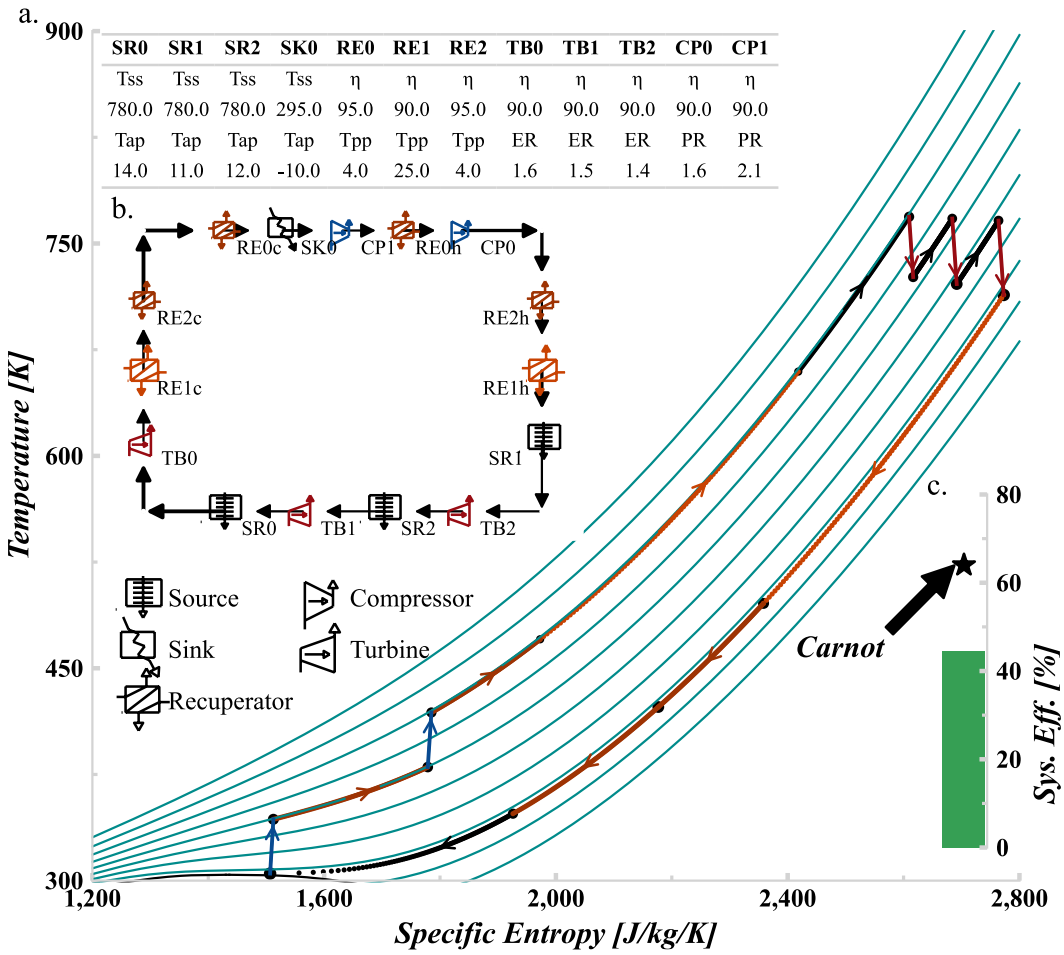

Figure 3-24. Best configuration without split/merge: (a) T-s diagram of system, (b) System configuration and performance variables of every component, (c) System efficiency

The optimization result is based on the configuration with one pair of split/merge (s_1) is shown in Figure 3-22. The minimum pressure in the system is around 8.0 MPa, and the optimized configuration converges to a reheated recompression cycle with an overall efficiency of 48%. Same as the optimized result for configuration s_0, in this optimized configuration, multiple reheating processes are utilized to improve further the average heating temperature for boosting efficiency. Recompression configuration is evolved out as the basic structure of this optimization result. This recompression process has been reported to highly improve efficiency among various advanced

configurations for different application areas, such as nuclear power generation (Dostal, Hejzlar, and Driscoll 2006; Ahn et al. 2015), solar power (Turchi et al. 2012; Neises and Turchi 2014), and molten carbonate fuel cell systems (Bae et al. 2014). The optimized split ratio of the splitter is around 1.7 to match the high-temperature and low-temperature recuperators. The compressor has no internal heating process since the heat capacity is well matched by unequaled mass flow from high pressure and the low-pressure side. This configuration cannot lower the average heat-dissipating temperature while maintaining relatively high thermal efficiency.

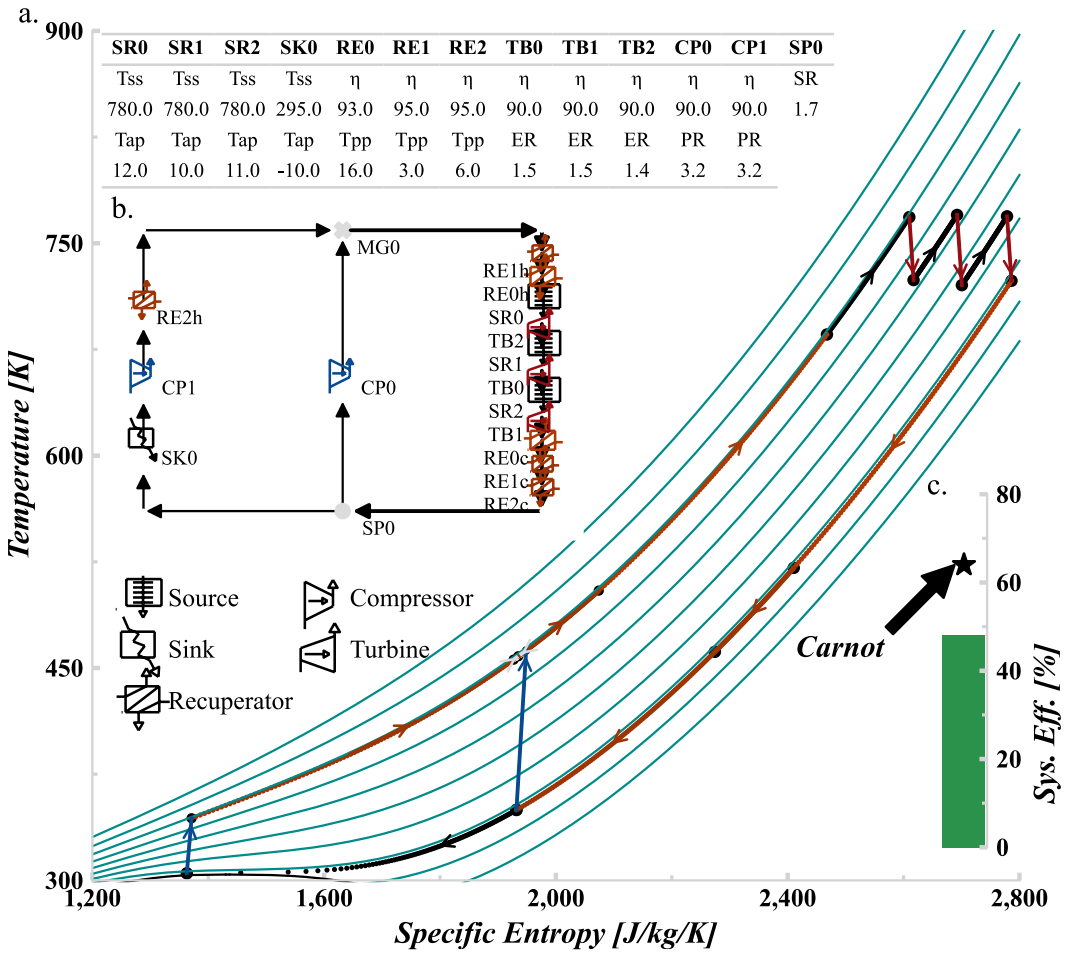

Figure 3-25. Best configuration with one split/merge: (a) T-s diagram of system, (b) System configuration and performance variables of every component, (c) System efficiency

Since there are multiple configurations for the same topological structure with two splits/merges, only the most efficient configuration among all the configurations with two splits/merges is discussed in this context. The most efficient configuration is shown in Figure 3-26 among all the configurations with two splits.

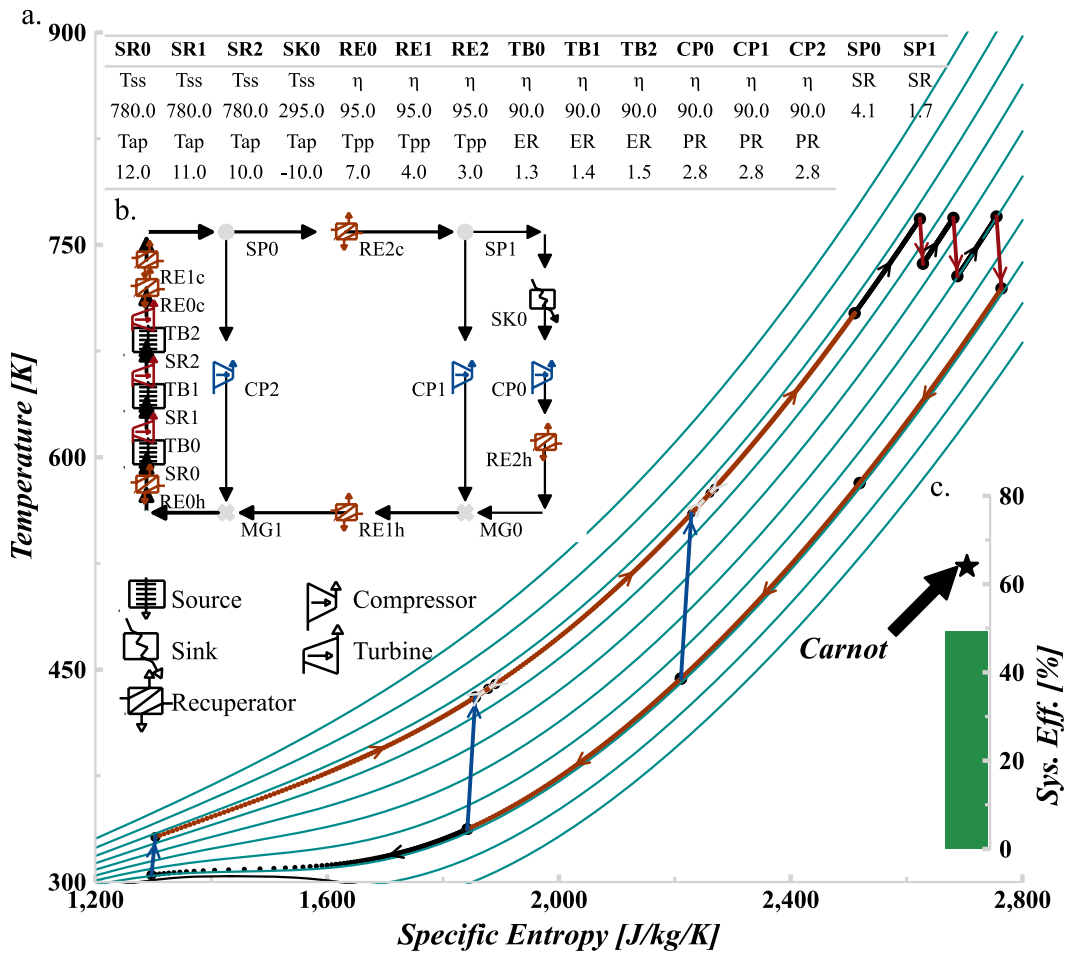

Figure 3-26. Best configuration with two splits/merges: (a) T-s diagram of system, (b) System configuration and performance variables of every component, (c) System efficiency

Three turbines, three compressors, three heaters, and three recuperators. The lowest pressure in the system is 8.1 MPa, and the optimum thermal efficiency reaches 49.1%. Compared with recompression-based configuration, the system further improves efficiency by 2.3% with one more split from the lower pressure side. And it is the same

with configuration with one split; there is no internal heating process for the compressor due to enough recovery process from recuperators. The thermal efficiency boost is relatively small compared with the elevation of the system complex, which might import unstable issues with condition change. Therefore, it is worthwhile to conduct a robustness analysis for all the candidate configurations.

3.4.4 Robustness analysis

In the above section, the best configurations in three categories (no split, one split, and two splits) are optimized from configurations based on six types of T-graph structures. The best configurations are further evaluated by parametric study. These parameters are those selected in the optimization process, including temperature-related, such as approach temperature, minimum pinch point, efficiency for compressor/turbine, effectiveness for recuperator, mass flow rate split ratio for a splitter. The overview of all varied parameters can be viewed in Table 3-6. Since there are two splits in configuration, the variation of split ratio for optimized configuration is presented separately.

Table 3-6. Parameter range for robustness analysis

Parameter	Range
Pinch point requirement [K]	[3, 15]
Approach temperature [K]	[10, 30]
Recuperator effectiveness [%]	[75, 95]
Pressure component efficiency [%]	[70, 90]
Split ratio change [%]	[-10, 10]

Based on the optimized configuration of each category of splits, I varied the minimum pinch point requirement for all recuperators from 3K to 15K. Figure 3-27 (a) shows the performance of thermal efficiency change along with pinch point requirements. It is

found that the optimized configuration without split has the most stable performance even though the optimum performance is lower than the other two. One of the recuperators in the configuration with no split has a higher value compared to the pinch point of all three recuperators in Figure 3-24 through Figure 3-26. This provides optimization space for the recuperator when varying the pinch point requirement. These three configurations have similar thermal efficiency under extreme conditions, where the recuperators' pinch point is 15K.

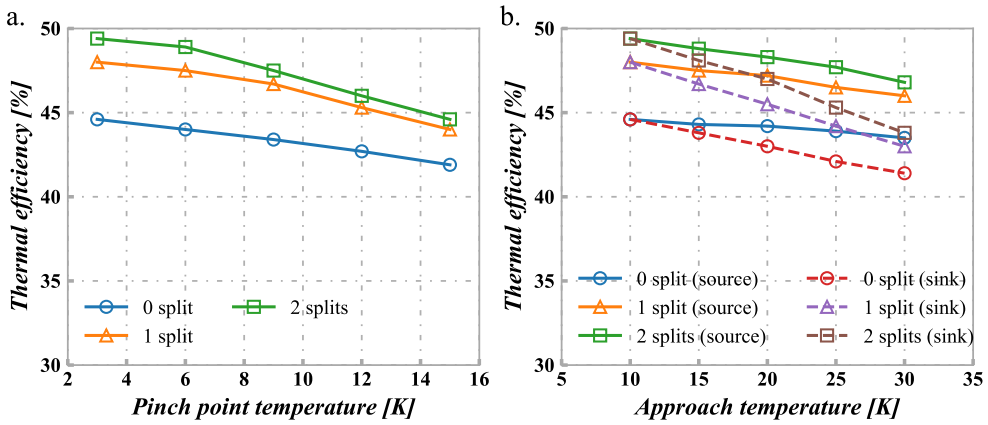

Figure 3-27. Influence of temperature-related parameters on thermal efficiency (a)Minimum pinch point requirement influence; (b) Approach temperature influence

By varying the approach temperature of the heater and cooler, system performance stability concerning the fluctuation of the ambient heat source is investigated. The performance change can be viewed in Figure 3-27 (b). It is seen that approach temperature for heat sink has a larger impact on performance degradation. And three categories of configuration have similar degradation for the approach temperature. The advantage of a two-split configuration is limited compared with one split configuration when the approach temperature increases to 30K.

The effectiveness of recuperators reflects how well the waste heat from the outlet of turbines is recovered. Figure 3-28 (a) shows that the effectiveness of the recuperator

has a large impact on the performance of the configuration with one split, whereas the influences on one split and two splits are limited. The highly efficient performance of the configuration with no split depends more on the recuperators' performance than on the other two configurations. Compared with configuration with split, the influence difference of recuperator effectiveness on performance is negligible. Therefore, the robustness of these two configurations is similar in terms of effectiveness. Figure 3-28 (b) shows the influence of pressure component efficiency on the overall thermal efficiency of systems. The result shows that different configuration categories have different influences on the system performance. The decreasing trend for two-split configurations is more notable than one split and no split configuration. The two-splits configuration has one more compressor to improve system efficiency and is thus more unstable when fluctuating the pressure component efficiency. There is no advantage when efficiency decreases below 75%. And it is observed that the thermal efficiency of a one-split configuration decreases slowly when component efficiency higher than 80% and sharply after that.

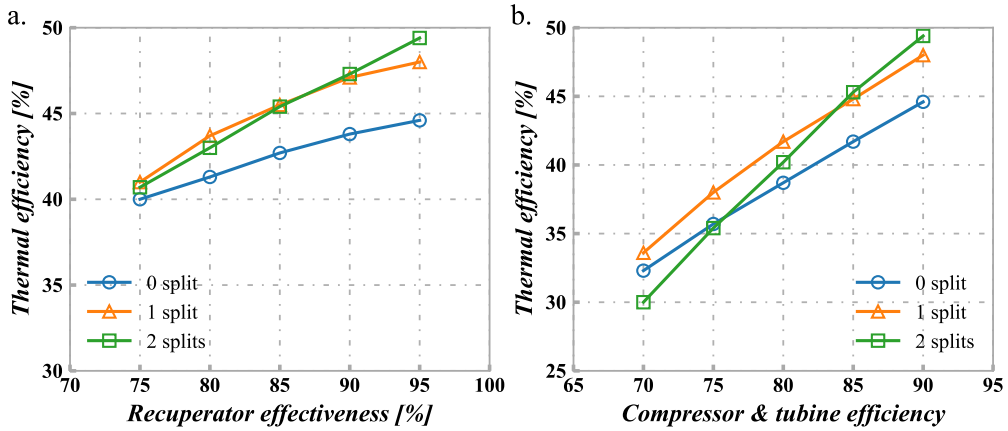

Figure 3-28. Influence of effectiveness and efficiency on thermal efficiency (a) Turbine and compressor efficiency influence (b) Recuperator effectiveness influence

The split ratio is the critical factor contributing to the relatively high efficiency of systems. To investigate the system's robustness under fluctuating split ratio for configurations with one split and two splits, this value of the optimized configurations is changed independently. Figure 3-29 (a) depicts the influence of the split ratio of the optimized configuration with one split with the value change from -10% to 10% of the optimized one. The results show that the efficiency is maintained between 48% and 47% with a changed split ratio. The configuration with two splits has more complicated changes when changed the ratios of two splits. The first split change has more impact compared with the second split since the first split controls the major split ratio. The second split has more effect when the first split decreases from the optimum value. The worst condition occurs when both split ratios fall. Overall, configuration with one split is relatively more stable, and the possibility of a mismatch of two splits ratio might degrade the system performance further. The system with one split is more robust than the system with two. And considering that the thermal efficiency of configuration with two splits is only 2.3% higher than that with one split, it is preferable to design a power system with one split rather than a system with two splits.

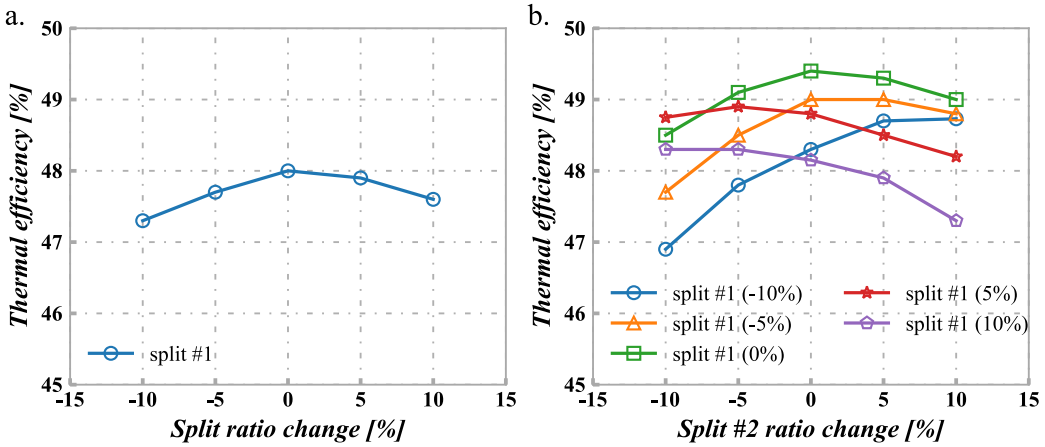

Figure 3-29. Influence of split ratio (a) Influence on configuration with one split (b)) Influence on configuration with two split

3.5 Summary

A configuration optimization framework was developed for optimizing the S-CO_2 power system. In this framework, a general thermodynamic graph was developed to represent any system configurations, including all the information on connections and governing equations. A simulated annealing-based optimizer was used to optimize the configuration. Strongly connected property and thermodynamic laws were implemented to generate new system configurations for the first time. Two types of case studies validated the framework. Several conclusions can be drawn as follows:

The developed optimization framework can effectively solve any newly generated system configuration. Designed mutation operators can be effectively implemented during the optimization process. The overall feasible configurations that obey graph theory and thermodynamic laws take over 99% of all configurations generated.

For the case study with no split/merge in the configuration, the framework can evolve out of the recuperated system with an efficiency of 38%. This optimization result was validated by pure performance variables tuning on 120 independent configurations. The optimization framework can converge to the best configuration. The performance variables of every component tend to approach the upper or lower boundary of settings. For the case study with one split/merge and more component numbers, the optimization framework converges to recompression configuration with an efficiency of 44%, which is the state-of-the-art configuration reported in other publications (Dostal, Hejzlar, and Driscoll 2006; Ahn et al. 2015; Turchi et al. 2012; Neises and Turchi 2014; Bae et al. 2014; M.-J. Li et al. 2017; Brun, Friedman, and Dennis 2017; S. Wright, Thomas, and

Gary 2011). The best configuration emerges slower than the simple case study due to the larger feasible region but tends to purely performance variables tuning at the second half of the iteration. The heat source and sink tend to approach upper and lower boundary conditions. The mass flow rate and the pressure ratio tend to decrease the irreversible loss in high-temperature and low-temperature recuperators. The optimized configuration in a smaller volume of feasible region can help with the earlier emergence of the best configuration and final pure performance variables tuning process for a larger volume of feasible region. Therefore, it would help with computational efficiency if the best configuration can be found for simple case conditions.

The optimized S-CO_2 system reaches a thermal efficiency of 44.6%, 48%, and 49.8% for no split, one split, and two splits when considering more components. This efficiency of the two splits system is 2.3% higher than the state of the art. The robustness of the three configurations is analyzed through a parametric study on pinch point, approach temperature, the effectiveness of recuperators, efficiency of pressure components, and split ratio. The two-split configuration degrades more than the one-split configuration in thermal efficiency. A one-split configuration is recommended, considering a two-split configuration's limited thermal efficiency improvement. And the configuration without a split is applicable when the system is designed under extreme requirements and conditions. With the help of the preliminary selection, optimization, and robustness processes, a broader range of system configurations in this single platform can be analyzed. And similar research, like different working fluids and objectives, can be conducted through this robustness analysis process.

4 Optimum Operation of CCHP

A well-defined CCHP system cannot reach optimum performance without optimum control to fulfill the energy demands. It would be unrealistic to have real-time control for such a complex CCHP system, considering the complicated physical principle of every component in subsystems and the coupling factor of these subsystems. In comparison, offline operation optimization would be a feasible way to investigate the overall performance in a period. First, excluding all the storage subsystems, the subsystems are deemed operating under a CCHP system's transient condition. In other words, every subsystem is assumed to operate in steady-state in a defined period, like an hour or a day. In the meantime, the subsystems are governed by simplified performance maps that summarize their characteristics sufficiently for the operation optimization of a CCHP system without storage subsystems. However, energy storage is a necessary component for a CCHP system to operate steadily and efficiently, namely peak shaving and load shifting. Thereafter, a CCHP system with storage requires more subtle strategies since a trajectory of storages future states should be determined at any time step. Like the shortest path problem, dynamic programming arises for the operation optimization of a CCHP system with storage subsystems. The adjacent time steps can be viewed as adjacent cities. The different state of storage system can be viewed as location in a city. The cost of operation from one state of current time step to another state of next time step can be viewed as path length from one location of one city to another location of adjacent city. A dynamic programming problem can be decomposed into multistage subproblems with the help of the principle of optimality.

Besides the dynamic property, another key issue that needs attention is the uncertainty within the CCHP system. Compared with the deterministic energy conversion subsystems, the energy demands are not always accurate due to inaccurate weather forecasts and occupant schedule fluctuation within buildings. Therefore, it is desirable to incorporate uncertainty into the operation optimization process. Dynamic programming is extended to a stochastic dynamic programming problem for the optimum operation of a CCHP system with storage subsystems with uncertainty. As compared with DP, the difference in SDP is that the solution to each subproblem is an expected value regarding the uncertainty information. There are many other formulations for stochastic programming. Various forms of stochastic optimization have been implemented for energy system operation, most of them are targeted at the electricity market and there is insufficient research work focused on integrated thermal energy subsystems, like a CCHP system. On the other hand, risk level preference is of importance for decision makers when operating an energy system. However, considering the complexity of the risk measurement, non-convexity of feasible regions and inefficient computation, this thesis limited the stochastic programming to risk neutral approach, which means the risk level preference is not considered.

This chapter introduces a general representation of CCHP systems, followed by an uncertain demands analysis, where the sampling method, implementation, and validation are illustrated. Since the SDP problem experiences twofold difficulty, the computation of expectation involves a high-dimensional discretization. The computational complexity grows exponentially concerning the number of stages, including state space, outcome space, and action space. Therefore, in the last part, an

approximation method named stochastic dual dynamic programming is depicted and applied for solving the SDP problem of CCHP with uncertainty.

4.1 CCHP system representation

A system and energy flow representation should be carried out to make the operation optimization process more general for a combined system with various designs. As illustrated in Figure 4-1, the CCHP system can be simplified as the nodes and edges in a direct graph, where electricity, cooling, and heating energy flows exist within energy storage, demands, and sources, such as renewable energy and fossil fuel. The energy flow network graph can be expressed as Equation 4-1 and Equation 4-2. *E* and *H* on the right-hand side of the equations represent electricity and heating generated by components. The superscript *t* represents time (a stage in operation viewpoint).

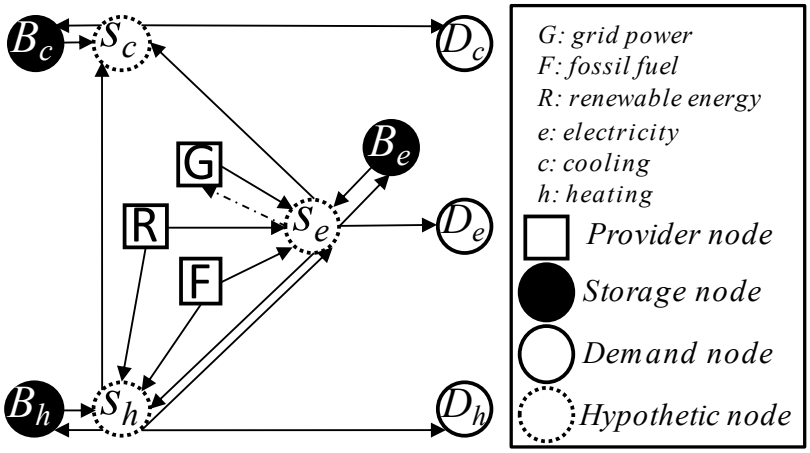

Figure 4-1. Abstract graph of CCHP energy flow

$$V := \{n_G, n_R, n_F, n_{S_c}, n_{S_h}, n_{S_e}, n_{D_c}, n_{D_h}, n_{D_e}, n_{B_c}, n_{B_h}, n_{B_e}\} \quad 4\text{-}1$$

$$E:=\left\{\begin{array}{c}[n_G,n_{S_e}],(n_R,n_{S_e}),(n_R,n_{S_h}),(n_F,n_{S_e}),(n_F,n_{S_h}),\\ [n_{S_e},n_G],[n_{S_e},n_{B_e}],[n_{S_e},n_{D_e}],(n_{S_e},n_{S_c}),(n_{S_e},n_{S_h}),\\ [n_{S_h},n_{B_h}],[n_{S_h},n_{D_h}],(n_{S_h},n_{S_e})\\ [n_{S_c},n_{B_c}],[n_{S_c},n_{D_c}]\end{array}\right\} \quad 4\text{-}2$$

The edge between two vertices represents the energy transfer between two formats or energy supply. Edges expressed as (*, *) indicate energy conversion subsystems on this edge, whereas only a loss factor is applied on edges described as [*, *]. Any subsystems in a CCHP system should be aligned on at least one of the edges (*, *). Therefore, the following equation should be satisfied for any subsystems in the CCHP system.

$$s \in (*,*), \text{for } s \in \Omega_{subsystem}^{all} \quad 4\text{-}3$$

The energy conversion process of every component can be expressed as Equation 4-4 based on the performance of subsystems. Every additional subsystem can find the proper position in the graph and be added to the energy balance equation.

$$\left\{\begin{array}{c}F_s^t\eta_s(\beta_s^t\alpha_s^t,a^t)=Cp_s\beta_s^t\alpha_s^t, s\in(n_F,*)\\ F_s^t\eta_s'(\beta_s^t\alpha_s^t,a^t)=H_s^t, s\in(n_F,n_{S_e})\cap\ s\in(n_F,n_{S_h})\\ H_s^t\eta_s(\beta_s^t\alpha_s^t,a^t)=Cp_s\beta_s^t\alpha_s^t, s\in(n_{S_h},*)\\ E_s^t\eta_s(\beta_s^t\alpha_s^t,a^t)=Cp_s\beta_s^t\alpha_s^t, s\in(n_{S_e},*)\end{array}\right. \quad 4\text{-}4$$

Where$F_s^t$is input fossil fuel for subsystem, $\eta_s$indicates the efficiency or COP profile function of subsystems, β_s^t represents the partial load of a subsystem, Cp_s is the capacity of subsystem, η_s' is the waste heat generation ratio from prime mover, H_s^t is recovered heat. It is noteworthy that, in general, there are ramp limitations for the partial load of subsystems. It is essential to import a binary variable α to indicate the on-off signal of a subsystem, and the partial load ratio β_s^t is replaced by the true partial load value $\beta_s^t\alpha_s^t$. a^t represents other factors that influence the subsystem efficiency.

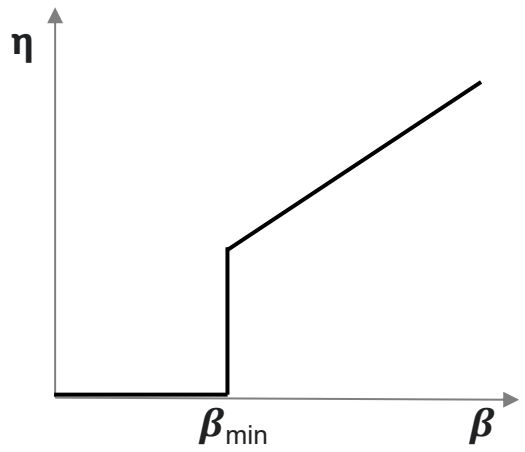

Figure 4-2. Ramp limitation of subsystem

Continuously, the energy conservation equation of the CCHP system that provides energy for demand sides at every time step can be expressed as Equation 4-5 based on the hypothetic nodes. The hypothetic nodes are used for energy balance calculations. There are three types of hypothetic nodes, heating, cooling, and electricity. The energy flow in and out of these hypothetic nodes should be equal. Therefore, it is easy to list the mathematical equation for the energy flow within the network. Here, it is easy to observe that every step involves the previous step status for storage subsystems. Thus, the current time step influences the CCHP system's future, which is the characteristic of dynamic programming.

$$E_{grid}^{t} - \sum_{s_1} E_{s_1}^{t} + \sum_{s_2} Cp_{s_2}\beta_s^t\alpha_s^t + \sum_{s_3} Cp_{s_3}\left(\beta_{s_3}^t - \beta_{s_3}^{t-1}\right) \geq E_{demd}^{t}, \quad \text{for } s_1 \in \left(n_{S_e},*\right), s_2 \in \left(*, n_{S_e}\right), s_3 \in \Omega_{elec}^{strg} \tag{4-5}$$

$$\sum_{s_1} H_{s_1}^{t} - \sum_{s_2} H_{s_2}^{t} + \sum_{s_3} Cp_{s_3}\beta_s^t\alpha_s^t + \sum_{s_4} Cp_{s_4}\left(\beta_{s_4}^t - \beta_{s_4}^{t-1}\right) \geq H_{demd}^{t}, \text{for } s_1 \in \left(n_F, n_{S_e}\right) \cap \left(n_F, n_{S_h}\right), s_2 \in \left(n_{S_h},*\right), s_3 \in \left(*, n_{S_h}\right) \backslash \left(n_F, n_{S_h}\right), s_4 \in \Omega_{heat}^{strg} \tag{4-6}$$

$$\sum_{s_1} Cp_{s_1}\beta_s^t\alpha_s^t + \sum_{s_2} Cp_{s_2}\left(\beta_{s_2}^t - \beta_{s_2}^{t-1}\right) \geq C_{demd}^t, \text{for } s_1 \in \left(*, n_{S_c}\right), s_2 \in \Omega_{cool}^{strg} \quad 4\text{-}7$$

The overall operation process cost is treated as the objective of the optimization problem. In the CCHP system, only natural gas and grid power need to be purchased. Therefore, the total cost of operation can be viewed as follows:

$$cost = \sum_t^T \left(c_{elec}^t E_{grid}^t + c_{fuel}^t \sum_{s_1} F_{s_1}^t\right), \text{for } s_1 \in (n_F, *) \quad 4\text{-}8$$

A conventional CCHP system that contains multiple subsystems, as illustrated in Figure 1-3, can be summarized in Equation 4-9. The electricity demand (ED) is satisfied by a prime mover (PM), which is a gas turbine (GT) in this system, an Organic Rankine Cycle (ORC), which recovers exhaust heat from the gas turbine to generate electricity, and a grid and electricity storage system (ESS). As for the heating demand (HD), the absorption heat pump (ABH) recovers exhaust heat from the gas turbine to generate more heat for space heating. The boiler (BL) and electric heating system (EH) consumes natural gas and electricity separately to provide heating capacity. Cooling demand (CD) is satisfied by absorption cooling (ABC) and/or vapor compression system (VCC). Heat and cooling storages (CSS, HSS) are used to adjust the energy demand and supply balance.

$$\Omega_{subsystem}^{all} := \{MT, ABH, ORC, BL, EH, ABC, VCC, CSS, HSS, ESS\} \quad 4\text{-}9$$

Based on the energy balance of the whole system, the constraints and objective can be expressed as Equation 4-10 through Equation 4-12. Moreover, the object can be depicted in Equation 4-13. The big M method (Bazaraa, Jarvis, and Sherali 2009) is used to dissociate the partial load ratio (β) and on-off signal (α) and eliminate the

nonlinearity brought by ramp limitation, as shown in Figure 4-2. This dissociation is expressed in Equation 4-11, where $\bar{\beta}$ and $\underline{\beta}$ represent upper and lower limit of the partial load, respectively, and $\overline{\beta}$ should be always equal to 1. Then the nonlinear property of actual partial load is transferred to linear constraints. Moreover, the natural gas consumed by GT and BL contributes to the objective of the total cost.

$$\begin{cases} F_{MT}^t \eta_{MT} = Cp_{MT} \gamma_{MT}^t \\ F_{MT}^t \eta'_{MT} = H_{MT}^t \\ F_{BL}^t \eta_{BL} = Cp_{BL} \gamma_{BL}^t \\ H_{ABH}^t \eta_{ABH} = Cp_{ABH} \gamma_{ABH}^t \\ H_{ORC}^t \eta_{ORC} = Cp_{ORC} \gamma_{ORC}^t \\ H_{ABC}^t \eta_{ABC} = Cp_{ABC} \gamma_{ABC}^t \\ E_{EH}^t \eta_{EH} = Cp_{EH} \gamma_{EH}^t \\ E_{VCC}^t \eta_{VCC} = Cp_{VCC} \gamma_{VCC}^t \end{cases} \tag{4-10}$$

$$\begin{cases} \underline{\beta_s^t} \alpha_s^t \le \gamma_s^t \le \alpha_s^t \bar{\beta}_s^t \\ \beta_s^t - (1 - \alpha_s^t)\bar{\beta}_s^t \le \gamma_s^t \le \beta_s^t - (1 - \alpha_s^t)\underline{\beta_s^t}, \text{for } s \in \Omega_{subsystem}^{all} \\ \gamma_s^t \le \beta_s^t + (1 - \alpha_s^t)\bar{\beta}_s^t \end{cases} \tag{4-11}$$

$$\begin{cases} E_{grid}^t - E_{VCC}^t - E_{EH}^t + Cp_{ORC} \gamma_{ORC}^t + Cp_{MT} \gamma_{MT}^t + Cp_{ST_e}\left(\beta_{ST_e}^t - \beta_{ST_e}^{t-1}\right) \ge E_{demd}^t \\ H_{MT} - H_{ABH}^t - H_{ORC}^t - H_{ABC}^t + Cp_{BL} \gamma_{BL}^t + Cp_{EH} \gamma_{EH}^t + Cp_{ST_h}\left(\beta_{ST_h}^t - \beta_{ST_h}^{t-1}\right) \ge H_{demd}^t \\ Cp_{ABC} \gamma_{ABC}^t + Cp_{VCC} \gamma_{VCC}^t + Cp_{ST_e}\left(\beta_{ST_c}^t - \beta_{ST_c}^{t-1}\right) \ge C_{demd}^t \end{cases} \tag{4-12}$$

$$cost = \sum_t^T \left[c_{elec}^t E_{grid}^t + c_{fuel}^t (F_{MT}^t + F_{BL}^t) \right] \tag{4-13}$$

The subsystem performance and the unit price of energy resources can be found in Table 4-1.

Table 4-1. Subsystem performance and energy resource prices

Parameter	Value
Microturbine efficiency [%]	25
Vapor compression chiller COP [-]	3
Absorption chiller COP	0.7
Boiler efficiency [%]	82
Electric heater [%]	95
Supercritical heat recovery cycle [%]	15
Electricity rate [$/kWh]	0.112
Natural gas rate [$/MBtu]	6.741

4.2 Operation without uncertainty

There are several operation strategies for a CCHP system operation without considering uncertainty within the system. The complexity of the operation and computation increases from a straightforward prescheduled strategy to an optimization-based one. The following sections will illustrate the above strategy in detail and compare the performance between each other.

4.2.1 Prescheduled strategy

The decision-making in a prescheduled strategy depends on limited information, such as time or the comparison between demands and performance of a primary mover. The time-dependent strategy is simple, all or part of the subsystems in CCHP operate based on the time of the day. This operation strategy might render a cost-effective system if there is a big difference in time of use rates at different times and energy demand is relatively stable within corresponding time sections. Since building energy demands fluctuate daily, the system operation should be flexible enough for this requirement. The second prescheduled method has a deeper understanding of building energy

demands and subsystem performance. The first demand-based scheduling strategy follows thermal/electricity/hybrid load (FTL/FEL/FHL) (P.J. Mago and Chamra 2009).

In summary, the following strategies can be referred to in Figure 1-9 In the following demand strategies, the waste heat from the primary mover in the CCHP system is all used for thermally driven cooling or heating subsystem, limiting the implementation of new technology. Therefore, the waste heat was first transferred through a supercritical power cycle to generate additional electricity and then used for thermal demand. The modified load-following strategies are summarized as Equations 4-14 ~ 4-16. η_{MT_SC} indicates the overall thermal efficiency from cascaded microturbine and waste recovery (supercritical CO_2 cycle) electricity generation system, which is expressed in 4-17.

$$cost^t_{\mathrm{FEL}} = c^t_{fuel}\left[\frac{E^t_{demd}}{\eta_{MT}} + \frac{1}{\eta_{BL}}\,\mathrm{ReLU}\left(\frac{H^t_{demd}}{\eta_{ABH}} + \frac{C^t_{demd}}{\eta_{ABC}} - \frac{E^t_{demd}}{\eta_{MT\_SC}}\left(1-\eta_{MT\_SC}\right)\right)\right] \quad 4\text{-}14$$

$$cost^t_{\mathrm{FTL}} = c^t_{fuel}\left(\frac{H^t_{demd}}{\eta_{ABH}} + \frac{C^t_{demd}}{\eta_{ABC}}\right)\frac{\eta_{MT\_SC}}{\left(1-\eta_{MT\_SC}\right)} + c^t_{elec}\left[\mathrm{ReLU}\left(\left(\frac{H^t_{demd}}{\eta_{ABH}} + \frac{C^t_{demd}}{\eta_{ABC}}\right)\frac{\eta_{MT\_SC}}{\left(1-\eta_{MT\_SC}\right)} - E^t_{demd}\right)\right] \quad 4\text{-}15$$

$$cost^t_{\mathrm{FHL}} = c^t_{fuel}\left[\min\left(\frac{E^t_{demd}}{\eta_{MT\_SC}}, \frac{H^t_{demd}}{1-\eta_{MT\_SC}}\right)\right] + c^t_{elec}\left[\mathrm{ReLU}\left(E^t_{demd} - \left(\frac{H^t_{demd}}{\eta_{ABH}} + \frac{C^t_{demd}}{\eta_{ABC}}\right)\frac{\eta_{MT\_SC}}{\left(1-\eta_{MT\_SC}\right)}\right)\right] + c^t_{fuel}\left[\mathrm{ReLU}\left(\frac{E^t_{demd}}{\eta_{MT\_SC}}\left(1-\eta_{MT\_SC}\right) - \frac{H^t_{demd}}{\eta_{ABH}} - \frac{C^t_{demd}}{\eta_{ABC}}\right)\right] \quad 4\text{-}16$$

$$\eta_{MT\_SC} = \eta_{MT} + \eta_{SC} - \eta_{MT}\eta_{SC} \quad \text{4-17}$$

Building energy demand samples are extracted from the machine learning model to investigate the performance of the following demands' operation strategies. The well-trained machine learning model obtained in Chapter 2 is used to generate building energy demands. The historical weather information is associated with College Park, Maryland, and the building type is a hospital with a floor area of 22,422 m^2. Three types of building energy are considered. The sampled cooling, heating, and electricity demands are shown in Figure 4-3. These demand-following operation strategies are implemented in this building based on the energy demands. The operation process should be noticed in a way that subsystems have unlimited capacity. This operation result can be used as a limitation while it is unrealistic in real-world applications.

However, it can be used to determine optimum capacities when the unit price of the capital cost is available for each subsystem. Although the capacity design is necessary for the system operation, it yields a bilevel optimization process and complicates the whole operation process. Moreover, there is a lack of credible correlation for the capital cost of all subsystems, which makes the design optimization work challenging. Therefore, in this thesis, unlimited capacities are assigned to subsystems. The result of the load-following strategies is used as a baseline for simple operation (load-following) strategy under extreme conditions. In the optimization-based operation strategies, the capacity is predefined with reasonable values for analysis.

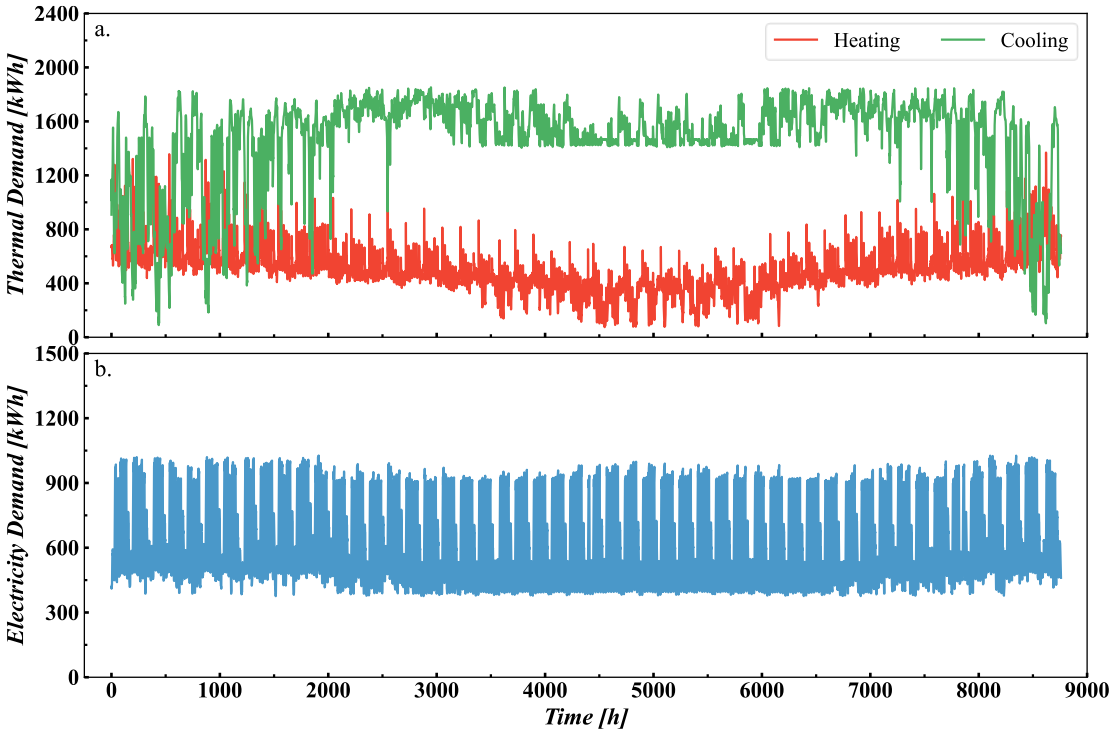

Figure 4-3. Energy demand sampling from machine learning model at College Park, Maryland a) thermal demand; b) electricity demand

Figure 4-4 shows the operation result of different load-following strategies for building energy demands integrated on a monthly basis. It is clearly shown that the FHL is more cost-effective than FTL and FEL. There is always energy wasted under certain scenarios, as shown in Figure 1-7. The superiority of FHL is obvious in the summer season from June to September. The annual operation cost of FHL for this hospital is 25.35 $/m^2. This value serves as a baseline for optimization-based operation strategy.

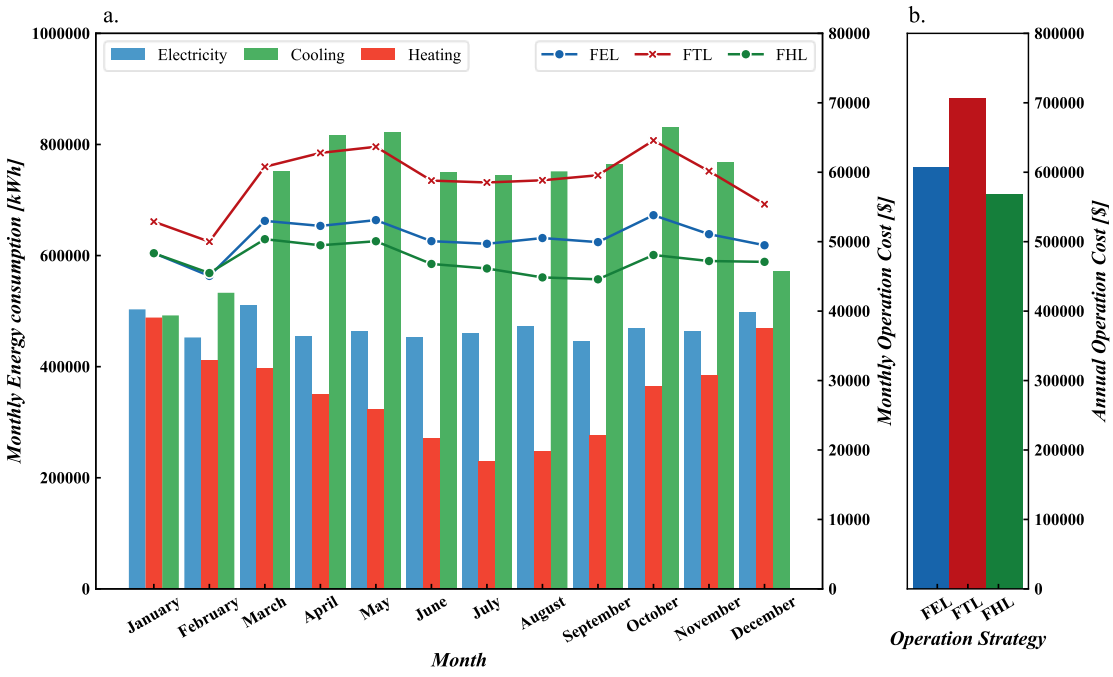

Figure 4-4. Load-following strategies on hospital building energy a) Monthly energy demand and operation cost; b) Annual cost of different operation cost

4.2.2 Optimization-based strategy

As described above, the primary energy is not used in an optimum way for prescheduled operation strategy. It is necessary to investigate the saving potential by implementing an optimization procedure for the system operation. The capacity of subsystems can be viewed in Table 4-2, together with the subsystem's performance in Table 4-1. The operation optimization process is expressed in Equation 4-10 through Equation 4-13. The storage subsystems are not considered at the very beginning. Moreover, the storage systems are assumed to be used up at the end daily for decreasing the computational complexity. Therefore, the optimization-based strategy operates the CCHP system daily and implements for annual analysis.

Consequently, variables representing the partial load of storage are removed from Equation 4-10 through Equation 4-13. The optimization process is then regarded as

static optimization compared with the full version of the operation with storage subsystems. In other words, the optimization process is executed individually without considering future time step, and every time step constitutes an independent optimization problem. Then storage subsystems are added to improve the CCHP system's performance further.

Table 4-2. Capacities of the subsystem in the CCHP system

Subsystems	Capacity [kW]
Microturbine	1,000
Vapor compression chiller	1,200
Absorption chiller	1,200
Boiler	800
Electric heater	800
Supercritical heat recovery cycle	400
Heating storage	600
Cooling storage	600
Electricity storage	600

The performance of the optimization-based operation strategy shows priority over FHL from a cost perspective. A utilization ratio τ and a change ratio φ for different subsystems are developed to investigate the performance of subsystems. These two indicators show how well the subsystems (not including storage) are involved and how much the subsystems (including storage) are fluctuating in supplying energy. They are expressed in Equation 4-18. The utilization ratio is defined by integrating the partial load ratio monthly. The indicator function represents the fluctuation ratio, where the time step counts when the partial load ratio changes over 50% for adjacent time.

$$\tau = \frac{\sum_{t=1}^{T} \gamma_s^t}{T} \text{ for } s \in \left(\Omega_{subsystem}^{all} \backslash \Omega_{hsubsystem}^{strg}\right)$$

$$\varphi = \frac{\sum_{t=1}^{T} 1_{|\gamma_s^t - \gamma_s^{t-1}|>0.5}}{T} \text{ for } s \in \left(\Omega_{subsystem}^{all}\right) \quad \text{4-18}$$

Figure 4-5 shows that optimization-based operation strategies outperform prescheduled ones for most months. The CCHP system with storage surpasses one without storage. It also shows the performance from the whole year basis. It is observed that optimization-based operation beats FHL operation optimization by 3.2% and 14% for a system without and with storage, respectively. The saving potential can be improved by the more detail design of subsystems considering the predefined capacity of the subsystem. The cooling storage operates with a high utilization ratio compared with heating and electric storage, especially in summer. The electricity storage operates with a relatively stable utilization ratio due to the relatively stable electricity demand.

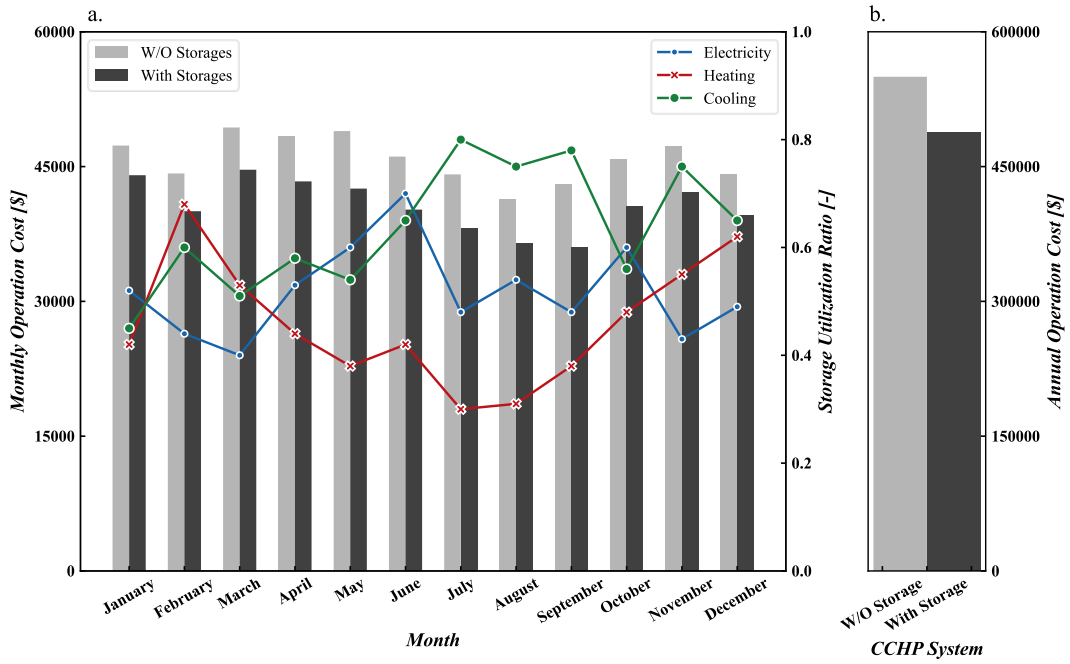

Figure 4-5. Optimization-based strategy for building energy

Then the utilization ratios between the system with and without storage from Figure 4-6 are compared. It is found that the integration of storage subsystems results in a lower utilization ratio of the microturbine and supercritical CO_2 cycle. In contrast, the

utilization ratio of electric heater, boiler, and vapor compression chiller changes little, and the corresponding value of absorption cooling chiller (thermally driven cooling) increases. Accordingly, for the CCHP system's design with storage, the capacity of the primary mover and waste heat recovery subsystem could be decreased to diminish capital cost, and the capacity of thermally driven generation subsystems could be increased to boost the subsystems' performance further.

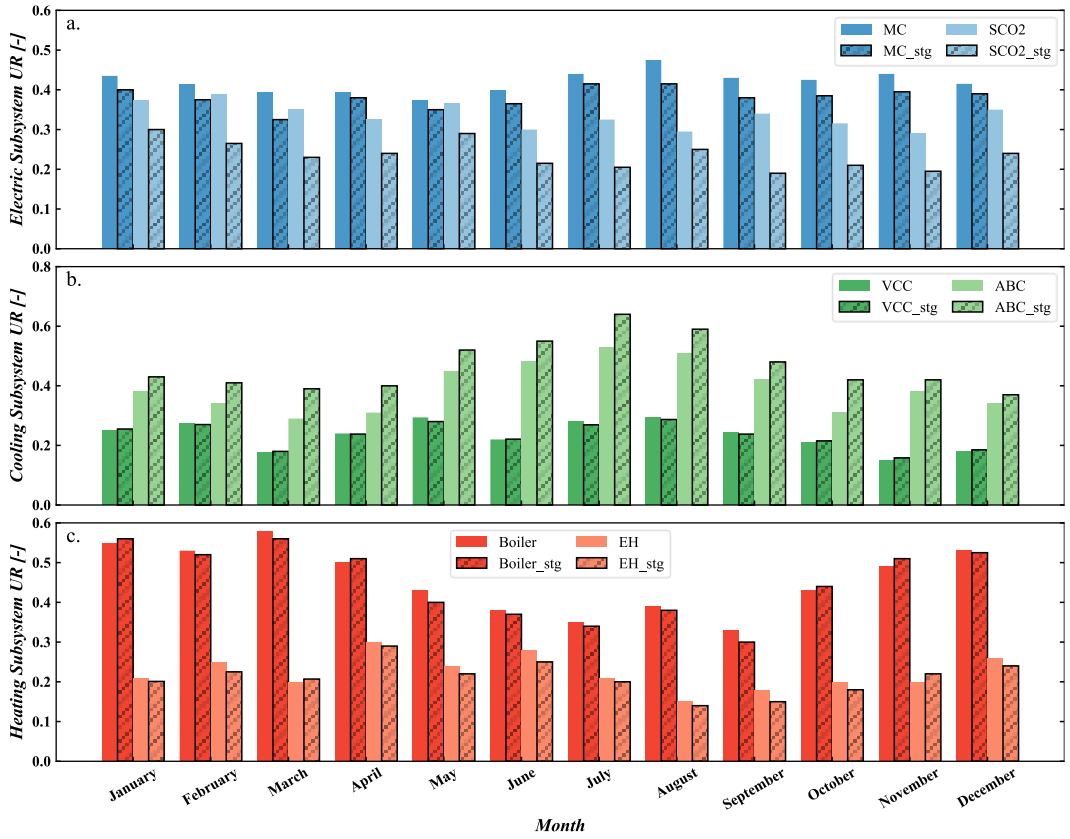

Figure 4-6. The utilization ratio of subsystems with and without storage

From the perspective of the fluctuation ratio shown in Figure 4-7, it can conclude that the storage subsystem can mitigate the subsystem fluctuation degree and render a smoother operation during the energy supplying process stacked up against the CCHP

system without storage. Furthermore, the thermal storages have a higher impact on the fluctuation of these subsystems than electric storage.

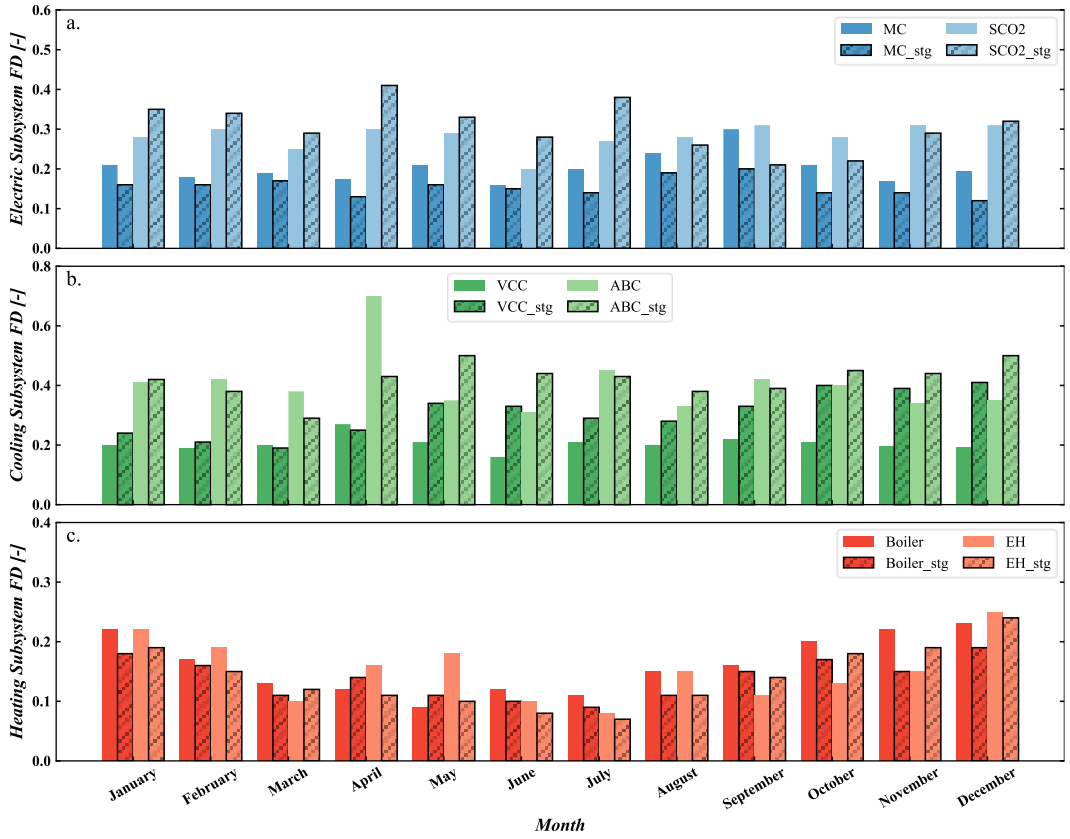

Figure 4-7. Fluctuation of subsystems with and without storage

4.3 Energy demand analysis

In the previous chapter, the machine learning-based prediction model provides hourly energy consumption for multiple building types with multiple energy types, which determines the energy demands without error. However, deterministic demands are unrealistic in real life. Building energy demands are always influenced by building envelope, weather information, and occupant behavior. Thus, uncertainty should be modeled for energy demands and applied to the operation of the CCHP system.

Based on the uncertainty information, there are two approaches to deal with uncertain data in optimization: stochastic optimization and robust optimization. In robust optimization (RO), a certain measure of robustness is established towards the uncertainty. Some of the data in the optimization problem are given in an uncertainty set which a simple linear optimization problem can represent. The uncertainty is shown as an upper or lower limit in the problem. In contrast to robust optimization, stochastic optimization (SO) problems deal with random data that obey a known probability distribution. The uncertainties are usually characterized by some discrete realizations of the uncertain parameters as an approximation to the real probability distribution. It has more information regarding the uncertainty compared with RO. A logical way to tackle the problem in stochastic programming is to make one decision now and minimize the expected costs (or profit) of the consequences of that decision, which is called the recourse model (Birge and Louveaux 2011). Stochastic models have proved their flexibility and usefulness in diverse areas of science and engineering (Shapiro, Dentcheva, and Ruszczyński 2009). Table 4-3 summarizes the difference between these two genera of optimization.

Meanwhile, variability of cost (or profit) is another aspect that needs to be characterized in stochastic optimization except the expectation. The general formulation of combining these two aspects can be viewed in Table 4-3 if a minimization problem is proposed, where β is a weighting factor used to quantify the tradeoff between expected profit (E) and risk aversion (r_w) (Conejo, Carrión, and Morales 2010). In addition, stochastic optimization can be categorized as risk neutral one when $\beta = 0$.

Table 4-3. Comparison of robust optimization and stochastic optimization

	Robust optimization (RO)	Stochastic optimization (SO)
Common	Deal with data that is not completely known	
Assumption	Belong to a certain set	Probability distribution
Math formulation	Min(worst case)	$\min_{x \in \boldsymbol{X}} \mathrm{E}\big(f(x,w)\big) + \beta r_w\big(f(x,w)\big)$
Result	Larger obj value	More optimum
Computation	Relatively simple	Complex

One formulation that used to control the level of the risk is chance constraints. It is a formulation that produces a decision which ensures that a set of constraints will hold with a certain probability. Or the constraints restrict the feasible region so that the confidence level of the solution is high. The optimization models with chance constraints are not guaranteed to be easily solved, because in general their feasible regions are not convex and may be not efficiently computable (Shapiro and Philpott, n.d.). And the authors also show that CVaR approximation is the best convex approximation of a chance constraint. Currently, CVaR measure is the most frequently proposed and used measure in the optimization problems (Artzner et al. 1999). This method has been well implemented for the energy system and electricity market. Xuan et al. applied CVaR for investment planning on equipment capacity of integrated energy systems (IES) and optimized the potential risk loss in operation scenarios along with risk preference. It is concluded that the study on CVaR-based IES planning is still a blank (Xuan et al. 2021). Montoya-Bueno et al. proposed scenario-based stochastic MILP model to invest in renewable generation economically (Montoya-Bueno, Muoz, and Contreras 2015). Javadi et al. proposed a risk-constrained optimization model for the home energy management system with the target of mitigating the electricity bill under uncertain solar PV generation (Javadi et al. 2021). Mu et al. proposed a CVaR-

based risk assessment method for planning schemes of park-level integrated energy systems with the consideration of uncertainties of natural gas price and electricity price (Mu et al. 2022). Hemmati et al. developed a CVaR based risk-constrained two-stage stochastic programming model to make optimal decisions on energy storage and thermal units in a transmission constrained hybrid wind-thermal power system (Hemmati, Saboori, and Saboori 2016).

There are many other risk measures to control the level of the risk in the energy market or energy system for investment, management, and portfolio selection, etc., These measurements include shortfall probability, expected shortage, stochastic dominance. The mathematical formulations that quantify the risk level can be redirected to (Birge and Louveaux 2011). Specifically, Dominguez et al. developed a stochastic dominance based multi-stage investment model considering long-term uncertainties in the decision-making process, including the demand growth and the investment and fuel costs, and short-term variability (Domínguez et al. 2021). Jian et al. applied a risk evaluation method called stochastic dominance (SD) to show the operation of the system in the risk-averse and risk-neutral models (Jian et al. 2022).

various forms of stochastic optimization have been implemented for energy system operation, most of them are targeted at the electricity market and there is insufficient research work focused on integrated thermal energy subsystems, like a CCHP system. On the other hand, risk level preference is of importance for decision makers when operating an energy system. However, considering the complexity of the risk measurement, non-convexity of feasible regions and inefficient computation, this thesis

limited the stochastic programming to risk neutral approach, which means the risk level preference is not considered and a multi-stage recourse model is implemented. An uncertainty sampling method is needed to investigate the CCHP system's optimum operation under stochastic conditions. The extraction of discrete uncertainty information from multiple historical energy consumptions. The following sections introduce uncertainty type, modeling, and sampling method independently for energy demands.

4.3.1 Stochastic demands

For multistage stochastic problems, the stochastic date process is modeled as Equation 4-19, in which the energy demand vector at the first stage $\boldsymbol{X}_1$ is deterministic and $\boldsymbol{X}_2 \dots \boldsymbol{X}_t$ is revealed gradually ($\boldsymbol{X}_t = [E^t_{demd}, C^t_{demd}, H^t_{demd}]$). And there are two types of the stochastic process when revealing future data sequentially. The first is stage-wise independent if the data of an arbitrary future event is independent of all previous time steps. The second one is called Markovian if the conditional probability distribution of an arbitrary future event given the entire past of the process equals the conditional probability of that future event given only one previous time step of the process. The mathematical expression of the independent and Markovian processes can be viewed in Equations 4-20 and 4-21, where Pr is the probability of an event.

$$\boldsymbol{X}_{[t]} \coloneqq (\boldsymbol{X}_1, \dots \boldsymbol{X}_t) \tag{4-19}$$

$$\Pr\left(\boldsymbol{X}_t \middle| \boldsymbol{X}_{[t-1]}\right) = \Pr(\boldsymbol{X}_t) \tag{4-20}$$

$$\Pr\left(\boldsymbol{X}_t \middle| \boldsymbol{X}_{[t-1]}\right) = \Pr(\boldsymbol{X}_t | \boldsymbol{X}_{t-1}) \tag{4-21}$$

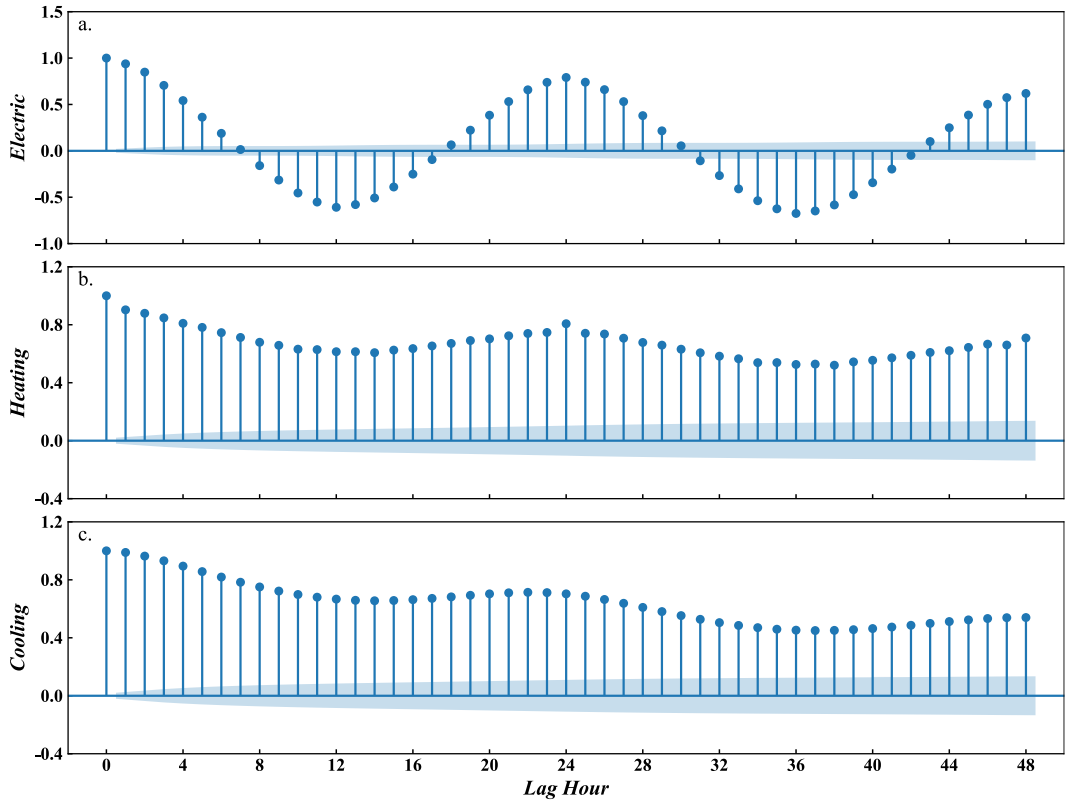

Figure 4-8. Autocorrelation of energy consumption of building

Figure 4-8 shows the autocorrelation of building energy consumption for the sampled hospital, as mentioned in the last subsection. Energy consumption is closely correlated to the previous time step. Therefore, the uncertainty demand cannot be treated as stage-wise independent. On the other hand, the Markovian would be not validated if there are multiple steps lag in the autoregression model since current state would be strongly related to previous n (n>1) steps. The Markov chain model cannot be used to replace multi-step autoregression model to predict future energy consumption. However, the autoregression model is adopted to generate sample data and use the sample data as input of Markov chain approximation model to discretize appropriate state space and probability transaction matrix, which is used later as input of stochastic dynamic programming model. Other regression model could be used to generate sample data,

but this autoregression model is adopted due to its simplicity and well performance in time series data prediction. In other words, the purpose of Markov chain sampling process in the thesis is not 100% accurately capturing the performance of autoregression model, or the Markov chain is not designed to reflect the characteristics of autoregression model.

4.3.2 Autoregression of energy demand

I take advantage of autoregression (AR) to generate samples with uncertainty from the output of the ML model. Briefly, the one-step-ahead prediction from the historical model can be expressed as follows.

$$\boldsymbol{X_t} - \boldsymbol{\mu}_t = \boldsymbol{\phi}(\boldsymbol{X}_{t-1} - \boldsymbol{\mu}_{t-1}) + \epsilon_t \quad \text{4-22}$$

where $\boldsymbol{X}_t$ is the time serial multiple energy vectors, including cooling, heating, and electricity ($\boldsymbol{X}_t = [E^t_{demd}, C^t_{demd}, H^t_{demd}]$) for a CCHP system, μ_t is average hourly energy consumption, and electricity. $\boldsymbol{\phi}_t = \boldsymbol{\phi}_{t+24}$ is three by three coefficient matrices of the three energy demands, ϵ_t is an independent and identically distributed (iid) sequence that has multivariate normal distribution follows $N(0, \boldsymbol{\Sigma}_t)$, $\boldsymbol{\Sigma}_t = \boldsymbol{\Sigma}_{t+24}$. $\boldsymbol{\phi}_t$ and $\boldsymbol{\Sigma}_t$ are calibrated by least squares approach.

4.3.3 Markov chain approximation

Continuing with the last subsection, recalling that the Markov process of energy demands at stage *t* was denoted as $\boldsymbol{X}_{[t]}$. The sampled space of energy demand at stage *t* is Ω_t and the joint distribution of $\boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_t$ that obtained from autoregression process from last subsection is $\boldsymbol{F}$. To approximate the Markov process, Voronoi cells were used to partition sampling space Ω_t into K_t subsets $\{\omega_{tk}, k=1,\ldots,K_t\}$ centered at μ_{tk} from stage

two. Let the center μ_{tk} be the Markov states. U_t denote the partition $\{\omega_{tk}, k=1,\ldots,K_t\}$ at stage t and U denotes the whole partition process $\{U_t, t=1,\ldots,T\}$. The objective of the sampling process is to search for a partition U that minimizes the Euclidean distance from centers $\mu_{tk}$and the Markov process $\boldsymbol{X}_{[t]}$. It can be expressed as follows.

$$\min_{U} \mathbb{E}_{\boldsymbol{X}} \left\{ \sum_{t=2}^{T} \min_{k=1,\ldots,K_t} \|\boldsymbol{X_t} - \mu_{tk}\|_2^2 \right\} \quad \text{4-23}$$

There are three approaches for solving the above problem, sample average approximation, stochastic approximation, and robust stochastic approximation methods. In addition, there is no clear evidence proving that one approach is superior to the others. Therefore, this study uses the stochastic approximation to calculate the state space and transition matrix. S independent sample paths $\{\widehat{\boldsymbol{X}}_1^s, \ldots, \widehat{\boldsymbol{X}}_t^s\}, s = 1, \ldots, S$ are drawn from joint distribution $\boldsymbol{F}$. The process starts by initializing μ_{tk} as μ^0_{tk} for k =1,…, K_t. With a sequence of step sizes (β_s) for s=1, …, S, where S is the number of sampling paths to train the Markov chain, the Markov states are updated recursively by stochastic approximation as follows. To guarantee a local minimum, β_s=1/s for s=1, …, S.

$$\mu_{ti}^s = \begin{cases} \mu_{ti}^{s-1} + \beta_s\left(\widehat{\boldsymbol{X}}_t^s - \mu_{ti}^{s-1}\right), \text{if i} = \arg\min_{k\in\{1,\ldots,K_t\}} \{\left\|\widehat{\boldsymbol{X}}_t^s - \mu_{tk}^{s-1}\right\|_2^2\} \\ \mu_{ti}^{s-1}, \text{otherwise} \end{cases} \quad \text{4-24}$$

Then, the transition matrix can be approximated by their relative frequency as:

$$\Pr_{ij}^t = \frac{\sum_{s=1}^{S} 1_{\widehat{\boldsymbol{X}}_{t-1}^s \in w_{t-1,i}} 1_{\widehat{\boldsymbol{X}}_t^s \in w_{t,j}}}{\sum_{s=1}^{S} 1_{\widehat{\boldsymbol{X}}_{t-1}^s \in w_{t-1,i}}} \quad \text{4-25}$$

where $1_{(\cdot)}$ is an indicator function. Until now, for $t = 2, \ldots, T$, the MCA discretized the stochastic process $\boldsymbol{X}_{[t]}$ by a Markov chain with Markov state spaces $\{s_t | s_t \in S_t\}, t = 1, \ldots T$, and transition matrices $\left\{\Pr^t_{s_{t-1}, s_t} \middle| s_{t-1} \in S_{t-1}, s_t \in S_t\right\}, t = 2, \ldots T$. The uncertainty discretizing process can be viewed in Figure 4-9.

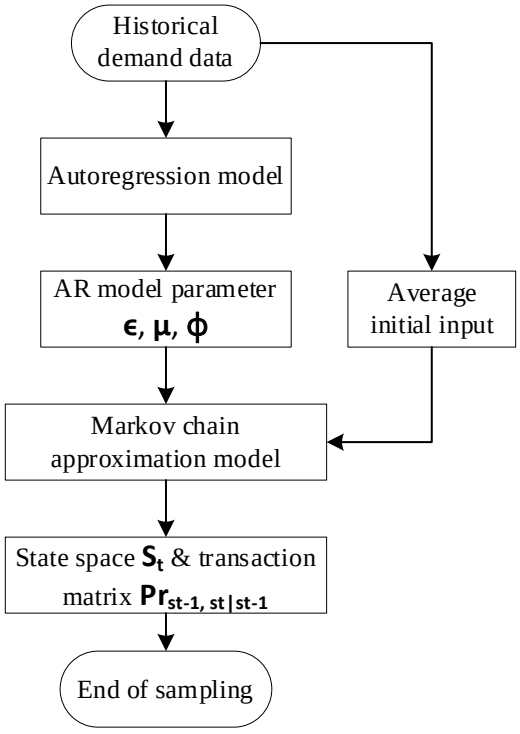

Figure 4-9. State space and transaction matrix sampling process

4.3.4 Sampling validation

A case study is implemented to verify the fidelity of this sampling method for MCA, as shown in Figure 4-9. The dataset is associated with the same hospital energy demands, as shown in Figure 4-3. As stated before, a stochastic approximation approach is used to train a non-homogenous three-dimensional Markov chain. And the initial Markov state is calculated based on the average value of periodic energy demands. One hundred Markov states are set from stage two in this validation process.

A total of 1000 and 10000 iterations are utilized for training the Markov states and the transition matrix independently. The original dataset is viewed first in the density plot of Figure 4-10.

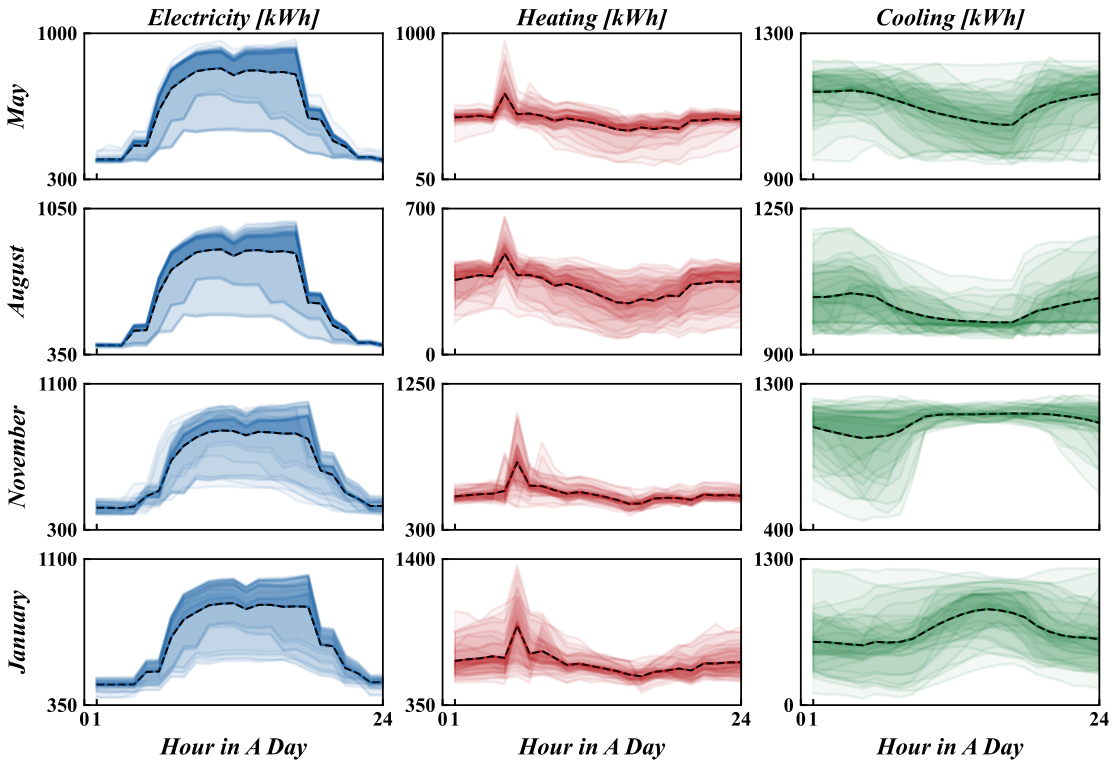

Figure 4-10. Historical energy consumption in four seasons

Then, the well-trained state space representing discrete energy consumption is visualized in the same format in Figure 4-11. Both grid plots show the hourly consumption in four typical seasons for three types of energy consumption. The black dashed line represents the average value of hourly consumption. Plots on the first, second, and third columns show the historical electricity, gas, and cooling data. Compared with historical energy consumption in Figure 4-10, the sampled energy consumption shows a similar pattern and trend for each type of energy demand, which

valid the fidelity of the sampling method. The transition matrix and the state space are used for the operation optimization of the CCHP system with uncertain demands.

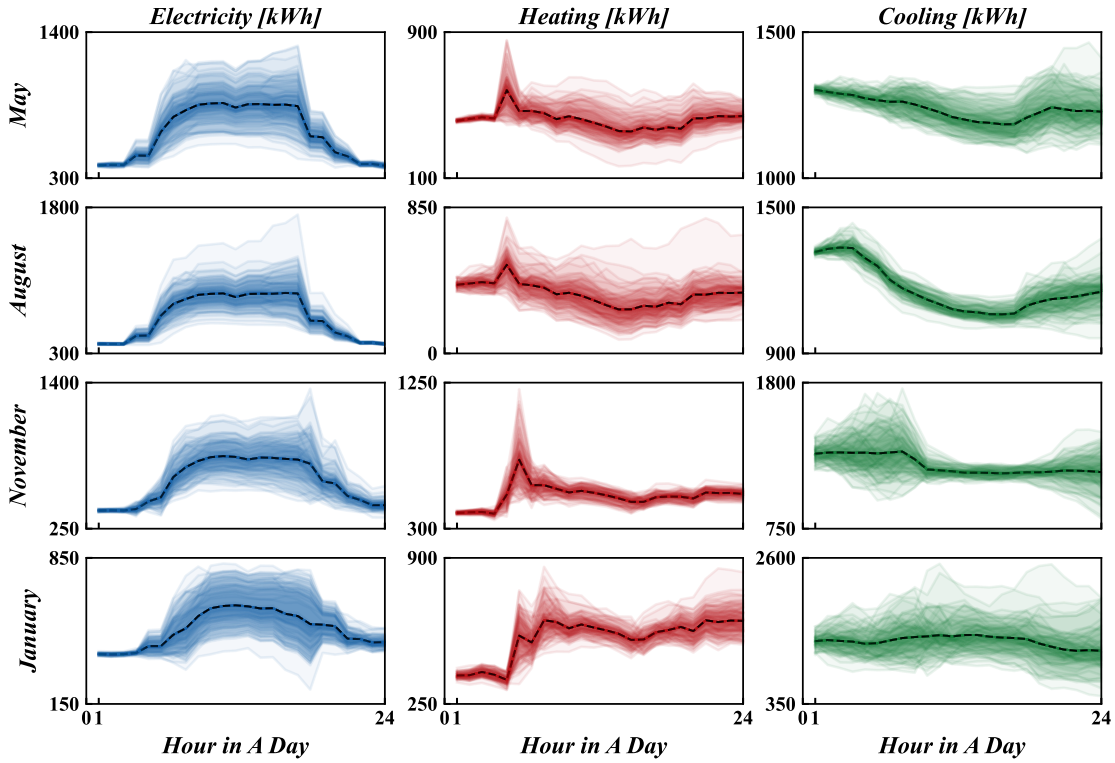

Figure 4-11. Sampled energy consumption in four seasons

4.4 Operation with uncertainty

This section illustrates the mathematical formulation of operation optimization starting for both general stochastic dynamic programming by recursive solving method and approximation approach for dynamic programming method. Then, the operation optimization based on these methodologies is then implemented and compared to the CCHP system with uncertain demands sampled from the last subsection.

4.4.1 Stochastic dynamic programming

Stochastic dynamic programming (SDP) problem forms when considering storage subsystem and uncertain energy demands. A nested format of optimization formulation can be used for concisely expressing a complete dynamic programming problem. Based on (Shapiro, Dentcheva, and Ruszczyński 2009), a standard nested formulation for a multistage stochastic linear problem can be described as Equation 4-26. It is observed that expected values are embedded in the objective function starting from stage two. In addition, the state information of the previous stage is constrained in the following stage. These two factors constitute the nested structure in the optimization problem.

$$\begin{aligned}\min_{A_1u_1+B_1u_0+C_1v_1\geq b_1} \; & p_1^\top u_1 + q_1^\top v_1 \\ & + \mathbb{E}|_{\xi_1}\left[\min_{A_2u_2+B_2u_1+C_2v_2\geq b_2} p_2^\top u_2 + q_2^\top v_2 + \cdots\right. \\ & \left. + \mathbb{E}|_{\xi_{T-1}}\left[\min_{A_Tu_T+B_Tu_{T-1}+C_Tv_T\geq b_T} p_T^\top u_T + q_T^\top v_T\right]\right]\end{aligned} \tag{4-26}$$

Here *x* and y are state variables and control variables. Specifically, these variables can be expressed as Equation 4-27 for the CCHP operation problem. ξ_t represents for the uncertainty information, all or part of parameters (*A, B, C, b, p, q*) that are evolved from the uncertainty information. Stochastic energy demands are included in ξ_t for this specific operation optimization problem. At the beginning of stage t (giving $\xi_{[t-1]}$]), the state variable u_{t-1} is regarded as information that obtains from stage *t*-1 to the current stage. In the decision process, decision variables u_t, v_t (*t* = 2,…, *T*) are considered to be functions $u_t(\xi_{[t]})$, $v_t(\xi_{[t]})$ of the stochastic process. It should be noted that the vector or matrix can be extended based on the subsystems in the CCHP system.

$$\begin{cases} u_0 = [0,0,0]^\top \\ u_t = \left[\beta_{ST_e}, \beta_{ST_h}, \beta_{ST_c}\right]^\top \\ v_t = [E_{grid}, \gamma_{MT}, \gamma_{BL}, \gamma_{ABH}, \gamma_{ORC}, \gamma_{ABC}, \gamma_{EH}, \gamma_{VCC}]^\top \end{cases} \quad \text{, for } t \in [1,T] \tag{4-27}$$

$$\begin{cases} A_t = \begin{bmatrix} Cp_{ST_e} & 0 & 0 \\ 0 & Cp_{ST_h} & 0 \\ 0 & 0 & Cp_{ST_C} \end{bmatrix} \\ B_t = \begin{bmatrix} -Cp_{ST_e} & 0 & 0 \\ 0 & -Cp_{ST_h} & 0 \\ 0 & 0 & -Cp_{ST_C} \end{bmatrix} \\ C_t = \begin{bmatrix} 1 & Cp_{MT} & 0 & 0 & Cp_{ORC} & 0 & -\frac{Cp_{EH}}{\eta_{EH}} & -\frac{Cp_{VCC}}{\eta_{VCC}} \\ 0 & Cp_{MT}\frac{\eta'_{MT}}{\eta_{MT}} & Cp_{BL} & -\frac{Cp_{ABH}}{\eta_{ABH}} & -\frac{Cp_{ORC}}{\eta_{ORC}} & -\frac{Cp_{ABC}}{\eta_{ABC}} & Cp_{EH} & 0 \\ 0 & 0 & 0 & 0 & 0 & Cp_{ABC} & 0 & Cp_{VCC} \end{bmatrix} \end{cases} \tag{4-28}$$

for $t \in [1,T]$

$$b_t = \left[\xi_t^{Electrity} \quad \xi_t^{Heating} \quad \xi_t^{Cooling}\right], \text{for } t \in [1,T] \tag{4-29}$$

$$\begin{cases} p_t = [0,0,0]^\top \\ q_t = \left[c_{elec}^t, \frac{Cp_{MT}}{\eta_{MT}} c_{fuel}^t, \frac{Cp_{BL}}{\eta_{BL}} c_{fuel}^t, 0,0,0,0,0\right]^\top \end{cases} \text{, for } t \in [1,T] \tag{4-30}$$

The nested formulation provides a mathematical description of the multistage stochastic program for the CCHP system with storage. The principle of optimality is used to solve the operation problem, in which dynamic programming techniques process a backward optimization over the time frame. The corresponding equations can be expressed in Equation 4-31.

$$Q_t(u_{t-1}, \xi_t) = \min_{A_t x_t + B_t x_{t-1} + C_t y_t \geq b_t} p_T^\top u_T + q_T^\top v_T + \mathcal{Q}_{t+1}(u_t, \xi_t) \tag{4-31}$$

Here, the expected cost-to-go (recourse) function $\mathcal{Q}_{t+1}(u_t)$ can be represented as:

$$\mathcal{Q}_{t+1}(u_t, \xi_t) := \begin{cases} \mathbb{E}\{Q_{t+1}(u_t, \xi_{t+1})\}, & \text{for } t = T-1, \dots, 1 \\ 0, & \text{for } t = T \end{cases} \tag{4-32}$$

It is noticed that to determine the optimal, any state variable completely (x_t) at every stage (t) should be informed through this backward optimization. Moreover, the stochastic process of the realization of related information should be discretized to make the problem solvable. The discretized Markov chain operation optimization of Equation 4-31 can be expressed based on the stochastic sampling method in the previous subsection. The extensive discrete formulation is expressed as Equation 4-33.

$$\begin{cases} \hat{Q}_t(u_{t-1}, s_t) = \min\limits_{A_t u_t + B_t u_{t-1} + C_t v_t \geq b_t} p_T^\top u_T + q_T^\top v_T + \hat{Q}_{t+1}(u_t, s_t) \\ \hat{Q}_1(u_0, s_1) = \min\limits_{A_1 u_1 + B_1 u_0 + C_1 v_1 \geq b_1} p_1^\top u_1 + q_1^\top v_1 + \hat{Q}_2(u_1, s_1) \end{cases} \quad 4\text{-}33$$

Where $s_t \in S_t, t = T, \dots 2$, and the expected future cost can be indicated as Equation 4-34.

$$\hat{Q}_{t+1}(u_t, \mathrm{s}_t) := \begin{cases} \sum_{s_{t+1} \in S_{t+1}} \Pr_{s_t, s_{t+1}} \hat{Q}_{t+1}(u_t, s_{t+1}), \text{for } t = T-1, \dots, 1 \\ 0, \qquad \text{for } t = T \end{cases} \quad 4\text{-}34$$

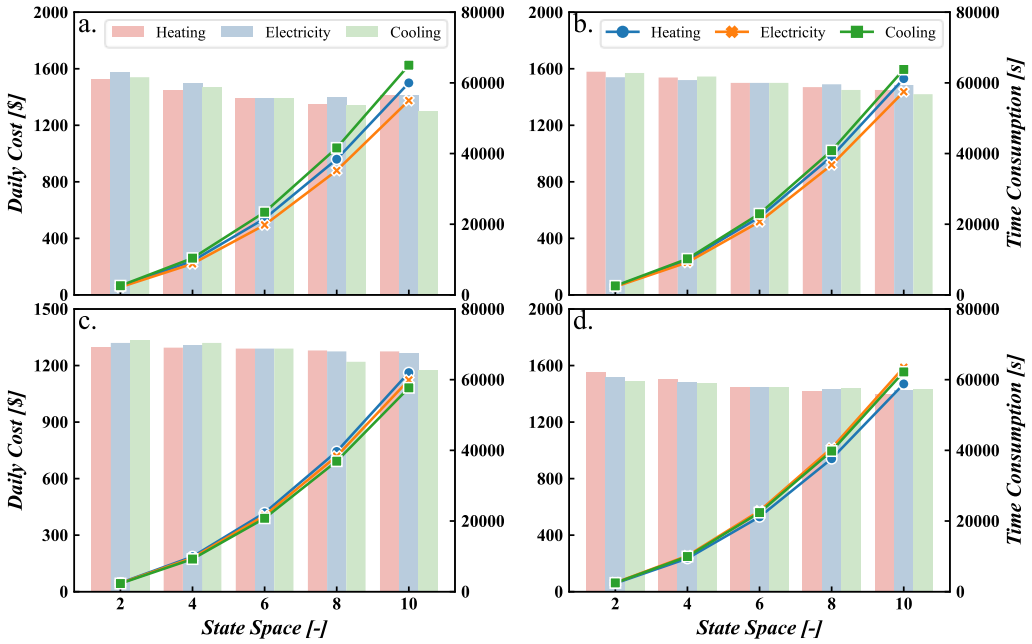

Figure 4-12. Daily operation costs by implementing the SDP method

This recursive approach is implemented for system operation with uncertainty. The result is displayed in Figure 4-12. The system operation cost shows a decreasing trend with the increase of discretization number for storage. For computational efficiency, the influence of discretization on storage is investigated independently. The discretization number for other storage subsystems is fixed to 6 when investigated one of them. It is observed that current discretization cannot fully take advantage of storage before encountering the computational efficiency problem. Further investigation for the finer discretization of storage is even more difficult for more storage and longer periods.

Table 4-4 shows a simple example of computational intractability in discretizing storage and uncertainty variables. It is noticed that the calculation time of entire state space reaches over 8 trillion for eight storage subsystems with 15 distinct state spaces under three uncertainties with a sampling size of 15 on each of them. This contributes to the curse of dimensionality in SDP. Therefore, a more efficient method should be applied to deal with this problem.

Table 4-4. Curse of dimensionality in terms of state variable

Number of state variables (N_{sv})	Number of discrete state (N_{ds})	Number of uncertain variables (N_{uv})	Number of sampling size (N_{ss})	Number of discrete problems $N_{ds}^{N_{sv}} N_{ss}^{N_{uv}}$
2	5	1	5	$(5)^2(5)^1$
	10		10	$(10)^2(10)^1$
	15		15	$(15)^2(15)^1$
4	5	2	5	$(5)^4(5)^2$
	10		10	$(10)^4(10)^2$
	15		15	$(15)^4(15)^2$
8	5	3	5	$(5)^8(5)^3$
	10		10	$(10)^8(10)^3$
	15		15	$(15)^8(15)^3 \approx 8.65 \cdot 10^{12}$

4.4.2 Approximation for stochastic dynamic programming

It is stated that every possible decision in state space at each stage should be evaluated so as to generate high dimensional calculation process for the dynamic programming process. Various techniques have been developed to approximate dynamic programming by reducing the state space discretization or approximating the future cost. Thus, it tackles this high-dimensional issue and makes it computationally tractable. Stochastic dual dynamic programming (SDDP) is one of the algorithms that has been successfully implemented for large-scale planning problems. It was first developed to solve the reservoir system's planning problem (Pereira and Pinto 1991). The approximation process utilizes the technique of Benders Decomposition to find a set of linear constraints that construct a piecewise linear approximation of future cost function. Moreover, SDDP only approximates the areas of future cost that most likely occur in the optimization problem, which further improves computational efficiency. The details of this technique, from dual dynamic programming to stochastic problems, is explained in the following sections.

4.4.2.1 Dual dynamic programming

Even though reducing state-space discretization is feasible to solve dynamic problems, it may result in less accuracy or computation inefficiency. On the other hand, interpolation of state space and analytical functions are the two streams for approximating future costs in dynamic programming. Dual dynamic programming belongs to the latter with a set of linear constraints for approximating the future cost function. It is a relatively straightforward alternative to discretize the state space if the future cost function is convex.

Considering a simple two-stage problem without uncertainty variables such as Equation 4-26, and the initial states of storage subsystems are empty ($x_0 = [0,0,0]$), then the two-stage problem can be expressed as Equation 4-35.

$$\begin{aligned} &\min p_1 u_1 + q_1 v_1 + p_2 u_2 + q_2 v_2 \\ &\text{s.t. } A_1 u_1 + C_1 v_1 \geq b_1 \\ &A_2 u_2 + B_2 u_1 + C_2 v_2 + D_1 v_1 \geq b_2 \end{aligned} \tag{4-35}$$

Where D_1 is an zero vector in this problem. The variables and parameters matrix is simplified to make this illustration process easier, as expressed in Equation 4-36.

$$\begin{aligned} &\min c_1 x_1 + c_2 x_2 \\ &\text{s.t. } F_1 x_1 \geq b_1 \\ &E_1 x_1 + F_2 x_2 \geq b_2 \end{aligned} \tag{4-36}$$

Where $x_1 = [u_1, v_1]$, $x_2 = [u_2, v_2]$, $c_1 = [p_1, q_1]^\top$, $c_2 = [p_2, q_2]^\top$, $F_1 = [A_1, C_1]$, $F_2 = [A_2, C_2]$, $E_1 = [A_2, D_1]$. The above problem is shown in Equation 4-37 to illustrate the dynamic property. It consists of a master problem representing the current cost and a subproblem indicating the future cost. The original dynamic programming evaluates the subproblem in a manner of discretizing state variable x_1 with a set of trial values $\{\hat{x}_1^i, i = 1, \dots, n\}$ and solves the subproblem for each trial value. As mentioned earlier, this dynamic program suffers computational intractability and can be resolved by future cost approximation.

$$\begin{aligned} &\text{Master problem: } \begin{cases} \min \mathrm{c}_1 x_1 + \alpha_2(x_1) \\ \text{s.t. } F_1 x_1 \geq b_1 \end{cases} \\ &\text{Subproblem: } \begin{cases} \alpha_2(x_1) = \min c_2 x_2 \\ \text{s.t. } F_2 x_2 \geq b_2 - E_1 x_1 \end{cases} \end{aligned} \tag{4-37}$$

First, the future cost $\alpha_2(x_1)$ can be characterized by the dual format as in Equation 4-38.

$$\alpha_2(x_1) = \max \pi_2(b_2 - E_1 x_1)$$
$$\text{s.t.}\ \pi_2 F_2 \leq c_2 \qquad \text{4-38}$$

Where π_1 is the dual variable corresponding to the original subproblem. It is interesting to see that the vertices corresponding to the constraints of the dual problem can be revealed before knowing the variable x_1 from the master problem. Then let $\Pi = \{\pi_2^1, \ldots, \pi_2^n\}$ denotes the set of all vertices of the constraint set. The subproblem can be solved by evaluating all the candidate vertices: $\alpha_2(x_1) = \max \pi_2^i(b_2 - E_1 x_1)$ for all *i*=1,…*n*, which can be expressed as Equation 4-39.

$$\alpha_2(x_1) = \min \alpha$$
$$\text{s.t.}\ \alpha \geq \pi_2^1(b_2 - E_1 x_1)$$
$$\ldots \qquad \text{4-39}$$
$$\alpha \geq \pi_2^n(b_2 - E_1 x_1)$$

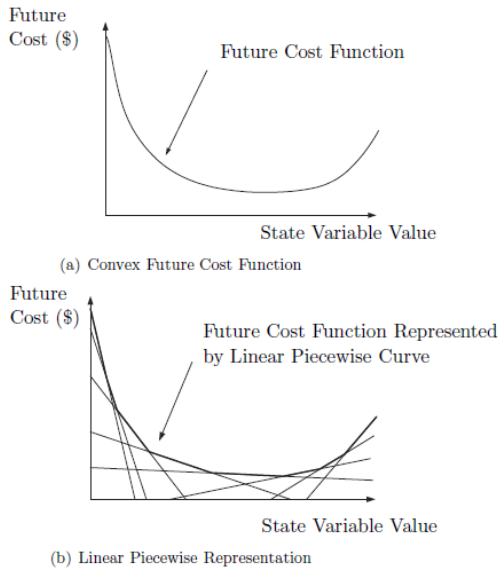

Figure 4-13. Approximation of future cost function

It is clear that the future cost function $\alpha_2(x_1)$ is a piecewise linear function in terms of the decision variable x_1. This characteristic can be interpreted as shown in Figure 4-13, which can also be called Benders cuts (Newham 2008). It should be noticed that this approximation is still valid even when the future cost is nonlinear but convex.

Whereas it is sometimes impossible to calculate all vertices in set Π. Then by calculating the dual variables $\pi^{\hat{x}_1^i}$ corresponding to the subproblem with trial value $\hat{x}_1^i$ in master problem. The supporting hyperplanes of the future cost function $\alpha_2(x_1)$ can be continuously added from these associated multipliers $\{\pi^{\hat{x}_1^i}, i = 1, \dots, r\}$, then the approximation to the future cost function can be summarized as Equation 4-40.

$$\begin{aligned} &\hat{\alpha}_2(x_1) = \min \alpha_2 \\ &\text{s.t. } \alpha_2 \geq \pi^{\hat{x}_1^i}(b_2 - E_1 x_1) \text{ for } i = 1, \dots, r \end{aligned} \tag{4-40}$$

It is understandable that $\hat{\alpha}_2$ is a lower bound to the future cost function α_2 since only a limited number of hyperplanes ($r \leq n$) have been added to the approximation process. And the master problem of the DP can be expressed as Equation 4-41.

$$\begin{aligned} &z = \min c_1 x_1 + \alpha_2 \\ &\text{s.t. } F_1 x_1 \geq b_1 \\ &\alpha_2 \geq \pi^{\hat{x}_1^i}(b_2 - E_1 x_1) \text{ for } i = 1, \dots, r \end{aligned} \tag{4-41}$$

The above Benders cut might be inefficient for calculation. Therefore, a more explicit way to add cut iteratively is by plugging in the trial value $x_1 = \hat{x}_1^i$. Solving Equation 4-38 with trial value $\hat{x}_1^i$, the approximation of the second stage at around trial value would be $\alpha_2(\hat{x}_1^i) = \pi^{\hat{x}_1^i}(b_2 - E_1 \hat{x}_1^i)$ and substitute $\pi^{\hat{x}_1^i} b_2$ into Equation 4-41. Then, the cuts in the above equation can be expressed as Equation 4-42.

$$\alpha_2 \geq \alpha_2\left(\hat{x}_1^i\right) - \pi^{\hat{x}_1^i} E_1\left(x_1 - \hat{x}_1^i\right) \text{ for } i = 1, \ldots, r \qquad \text{4-42}$$

Since $\hat{\alpha}_2$ is a lower bound to the future cost function, the solution of the above problem is a lower bound to the master problem. Moreover, the subproblem is more constrained than the original problem as the variable in the master problem is fixed at the trial value $\hat{x}_1$. Therefore, the added restriction results in an upper bound to the original optimization problem. The upper bound can be obtained by solving the subproblem for trial value $\hat{x}_1$ in Equation 4-37. Overall, the lower and upper bound solution can be expressed as Equation 4-43.

$$\begin{aligned} \underline{z} &= c_1\hat{x}_1 + \hat{\alpha}_2(x_1) \\ \overline{z} &= c_1\hat{x}_1 + \alpha_2(\hat{x}_1) \end{aligned} \qquad \text{4-43}$$

Then, the difference ϵ between predicted future cost ($\hat{\alpha}_2$) and the actual future cost ($\alpha_2(\hat{x}_1)$) of the current solution $\hat{x}_2$ can be used to evaluate whether this optimization problem has been solved. Otherwise, new trial values ($\hat{x}_1^i$) will be added to determine additional vertices ($\pi^{\hat{x}_1^i}$) and future lower this difference. An optimal solution found in the last iteration $\hat{x}_1^{k-1}$ will be used as an additional trial decision to ensure the approximation occurs around the neighbor of the optimal solution for the next iteration. The advantages are summarized as follows: no discretization is needed for the dual dynamic solving process, boundaries are provided for each iteration, and solutions from the previous iteration can be efficiently used in the next iteration. The two-stage dual problem can be extended to multiple stages, as shown in Figure 4-14.

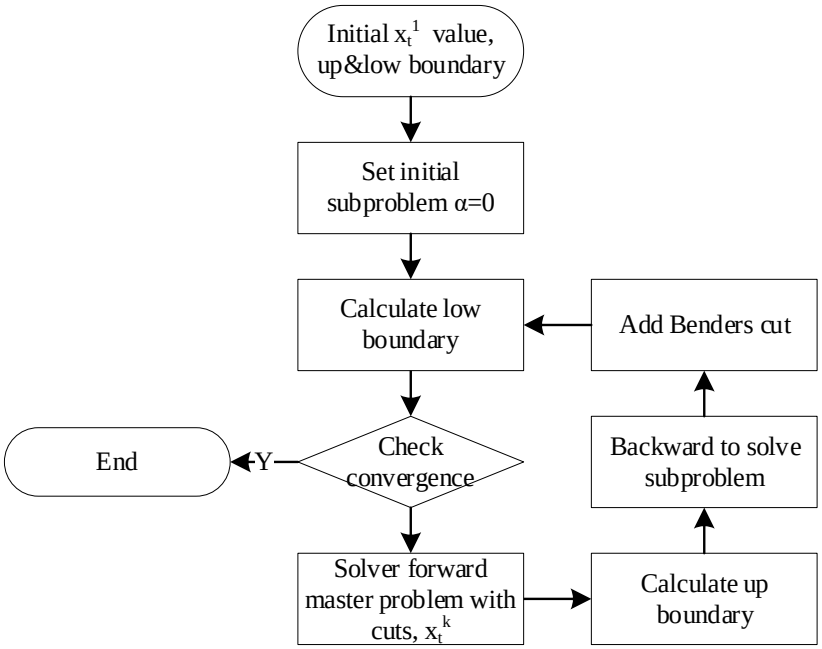

Figure 4-14. Flow chart of multi-stage dual dynamic programming

The solving process for the multi-stage dynamic problem can be explained as follows:

- Step 1: Initialize the value of approximated future cost $\hat{\alpha}_{t+1}(x_t) = 0$ for $t = 1, \dots, T-1$, the upper and lower boundary $\overline{z} = \infty$, $\underline{z} = 0$, iteration number *k*=1, and tolerance ϵ.
- Step 2: Solve the master problem based on Equation 4-37 to obtain x_1^k and calculate the lower boundary of the first stage as $\underline{z}^k$ and check for the convergence based on the lower and upper boundary.
- Step 3: If the difference in boundaries falls within tolerance, then stop. Otherwise go to step 4.
- Step 4: Forward pass, for time period *t* = 2,…T, solve the master problem with the trial decision $\hat{x}_{t-1}^k$ for stage t as shown in Equation 4-44, where $\pi^{\hat{x}_t^{k-1}}$ is the dual variable corresponding to trial decision $\hat{x}_t^{k-1}$ of previous iteration at the

current stage, and store the optimum decision variables of the current iteration as $\hat{x}_t^k$, which will become additional Benders cuts for the backward pass.

$$\begin{aligned} &\hat{\alpha}_t = \min[c_t x_t + \hat{\alpha}_{t+1}] \\ &\text{s.t. } F_t x_t \geq b_t - E_{t-1}\hat{x}_{t-1}^k \\ &\hat{\alpha}_{t+1} \geq \hat{\alpha}_{t+1}\left(\hat{x}_t^{k-1}\right) - \pi^{\hat{x}_t^{k-1}} E_t\left(x_t - \hat{x}_t^{k-1}\right) \end{aligned} \quad 4\text{-}44$$

- Step 5: New upper boundary $\overline{z}$ is calculated based on Equation 4-45.

$$\overline{z} = \sum_{t=1}^{T} c_t \hat{x}_t \quad 4\text{-}45$$

- Step 6: Proceed to backward recursion. Solving the optimization problem for stage t = T, …,2 as Equation 4-46 indicates. $\pi^{\hat{x}_t^k}$ is the dual variable found by solving the dual subproblem at time *t*+1 on this backward pass. $\hat{x}_{t-1}^k$ is the trial solution found from the forward pass master problem at time *t*-1, iteration *k*. Assume that $\pi^{\hat{x}_{t-1}^k}$ is the dual variable associated with the constraint of Equation 4-46, this dual variable will be used for calculating the Benders cut of Equation 4-47 and added to the corresponding forward pass master problem and backward pass problem.

$$\begin{aligned} &\hat{\alpha}_t = \min[c_t x_t + \hat{\alpha}_{t+1}] \\ &\text{s.t. } F_t x_t \geq b_t - E_{t-1}\hat{x}_{t-1}^k \\ &\hat{\alpha}_{t+1} \geq \hat{\alpha}_{t+1}\left(\hat{x}_t^k\right) - \pi^{\hat{x}_t^k} E_t\left(x_t - \hat{x}_t^k\right) \end{aligned} \quad 4\text{-}46$$

$$\hat{\alpha}_t \geq \hat{\alpha}_t\left(\hat{x}_{t-1}^k\right) - \pi^{\hat{x}_{t-1}^k} E_{t-1}\left(x_{t-1} - \hat{x}_{t-1}^k\right) \quad 4\text{-}47$$

- Step 7: Increase the iteration count *k* to *k*+1 and return to step 2.

4.4.2.2 Stochastic dual dynamic programming

It is inevitable to consider uncertain information during the operation of the CCHP system. Taking stochastic variables into consideration for system operation would provide insights into the effect of uncertainty on system optimum operation. The DDP has the natural advantage of extending to the stochastic scenario. SDDP has a similar solving process as DDP, where the whole optimization process is split into a series of mater problems and subproblems. Benders cuts still approximate the subproblems. Moreover, the algorithm stops when the difference between the upper and lower boundary falls into the threshold range. However, stochastic variables affect the calculation process of the subproblems, where each of them is represented by discrete potential subproblems with corresponding probability. Therefore, an expected Benders cut is calculated for every possible subproblem instead of single Benders cut during the approximation process. The upper and lower bounds become expected ones by importing the expected approximation of the original subproblem.

A two-stage SDDP problem can be expressed as Equation 4-48. It is worthwhile to mention that the two-stage Markov chain SDDP can be treated the same as the two-stage independent SDDP problem after discretization of uncertainty. Therefore, the problem is formulated in the fashion of the stage-independent case for the convenience of explanation. The multi-stage problem will be illustrated as Markov chain SDDP.

$$\begin{aligned}
&\min\left[c_1x_1 + Pr_1c_2x_{2,1} + Pr_2c_2x_{2,2} + \cdots + Pr_nc_2x_{2,m}\right] \\
&\text{s.t. } F_1x_1 \geq b_1 \\
&E_1x_1 + F_2x_{2,1} \geq b_{2,1} \\
&E_1x_1 + F_2x_{2,2} \geq b_{2,2} \\
&\ldots \\
&E_1x_1 + F_2x_{2,m} \geq b_{2,m}
\end{aligned} \qquad \text{4-48}$$

The two-stage problem is split into one master problem and *m* subproblems with a probability of Pr_m, the master problem and each subproblem are expressed as 4-49.

$$\text{Master problem: } \begin{cases} \min[c_1 x_1 + \bar{\alpha}_2(x_1)] \\ \text{s.t. } F_1 x_1 \geq b_1 \end{cases}$$

$$\text{Subproblem: } \begin{cases} \hat{\alpha}_{2,j}(x_1) = \min c_2 x_{2,j} \\ \text{s.t. } F_2 x_{2,j} \geq b_{2,j} - E_1 x_1 \end{cases} \qquad 4\text{-}49$$

$$\bar{\alpha}_2(x_1) = \sum_{j=1}^{m} Pr_j \hat{\alpha}_{2,j}(x_1)$$

Similar to DDP, dual variable $\pi_j^{\hat{x}_1^i}$ is produced when solving the subproblem indicated in Equation 4-50 by plugging in the trial decision variable $\hat{x}_1^i$ from the master problem.

$$\begin{gathered} \hat{\alpha}_{2,j}(x_1) = \min c_2 x_{2,j} \\ \text{s.t. } F_2 x_{2,j} \geq b_{2,j} - E_1 \hat{x}_1^i \text{ for } i = 1, \dots, r \end{gathered} \qquad 4\text{-}50$$

For solving all subproblems for j =1,…,m, each subproblem finds a dual variable $\pi_j^{\hat{x}_1^i}$ and optimal solution $\hat{\alpha}_{2,j}(x_1)$. The expected value of the dual variable is given by Equation 4-51.

$$\bar{\pi}^{\hat{x}_1^i} = \sum_{j=1}^{m} Pr_j \pi_j^{\hat{x}_1^i} \qquad 4\text{-}51$$

The above-expected dual variable $\bar{\pi}^{\hat{x}_1}$ is used to calculate the Benders cut shown in Equation 4-52 for the master problem.

$$\alpha_2(x_1) \geq \bar{\alpha}_2(x_1) - \bar{\pi}^{\hat{x}_1^i} E_1 \left(x_1 - \hat{x}_1^i\right) \qquad 4\text{-}52$$

The upper bound of the SDDP is developed similarly to a deterministic problem, whereas the upper bound is an expected value provided by Equation 4-53.

$$\bar{z} = \frac{1}{\mathrm{m}} \sum_{i=1}^{m} \sum_{t=1}^{T} c_t \hat{x}_t \qquad 4\text{-}53$$

Now, the lower bound of the two-stage SDDP problem can be calculated using Equation 4-54.

$$\underline{z} = c_1 \hat{x}_1 + \bar{\alpha}_2(x_1) \qquad 4\text{-}54$$

Different from the deterministic problem when the upper and lower boundary are compared directly, the upper bound is an expected value. There is uncertainty around the estimated upper bound. This uncertainty can be provided by the standard deviation of the expected upper boundary in Equation 4-55.

$$\sigma = \frac{1}{\mathrm{m}-1} \left[\sum_{i=1}^{m} (z_i - \bar{z})^2 \right]^{\frac{1}{2}} \qquad 4\text{-}55$$

Where z_i is upper bound that is calculated on *i*th forward pass. It is now capable of quantifying the upper bound with confidence intervals. For instance, there is 95% confidence that the actual upper bound will be found within the range $[\bar{z} - 1.96\sigma, \bar{z} + 1.96\sigma]$.

Similar to the DDP, a two-stage SDDP can be extended to a multi-stage problem. The solving process for multi-stage SDDP following next seven steps:

- Step 1: Given the state space of the Markov chain S_t, transition matrix $\{Pr_{s_{t-1},s_t} | s_{t-1} \in S_{t-1}, s_t \in S_t\}$ for *t*=2, …, T and initial state s_1. Initializing the value of approximated future cost $\bar{\alpha}_t^0(x_t, s_t) = 0$ for $t = 2, \ldots, T$ and any

Markovian state, set iteration number k=1. the lower boundary $\underline{z} = -\infty$, and specific tolerance ϵ.

- Step 2: Evaluate the gap between the lower boundary and a specific number of independent simulations in the following way:

Assuming there are N times of evaluations on the constructed policy for the discretized problem by employing Monte Carlo simulation. In each evaluation, there is a sample path generated independently from the discretized problem that leads to an optimal solution ($\hat{x}_{tm}$) and policy value as Equation 4-56.

$$V_m = \sum_{t=1}^{T} [c_t \hat{x}_{tm}], m = 1, \dots, M \quad \text{4-56}$$

$$\bar{V}_M = \frac{\sum_{m=1}^{M} V_m}{M}$$
$$S_M^2 = \frac{\sum_{m=1}^{M} (V_m - \bar{V}_M)^2}{M-1} \quad \text{4-57}$$

$\bar{V}_M$ and S_M^2 represent the mean value and variance, respectively. Let v^* denote the expected value of the policy, $\frac{\bar{V}_M - v^*}{S_M/\sqrt{M}}$ converges to standard normal distribution $N(0, 1)$ according to the central limit theorem. Then it can be claimed that an approximate (1-α) one-sided confidence interval for the expected value is $\bar{V}_M + z_{1-\alpha}\frac{S_M}{\sqrt{M}}$, where $\Pr(Z \geq z_{1-\alpha}) = \alpha$ with $Z \sim N(0,1)$. The upper end of the confidence interval provides an upper bound of the optimal value of the discretized problem with the confidence of approximately $1-\alpha$. Therefore, with confidence $1-\alpha$, the optimality gap for the discretized problem is smaller than Equation 4-58.

$$gap = \frac{\bar{V}_M + z_{1-\alpha}\frac{S_M}{\sqrt{M}} - \underline{z}}{\underline{z}} \quad 4\text{-}58$$

- Step 3: Check for convergence. If the gap difference is smaller than the tolerance, then stop. Otherwise, it goes to the forward master problem pass.
- Step 4: Forward pass, sample the stochastic variable s_t from the state space with a probability of Pr_{s_{t-1},s_t} for each period $t = 2,\ldots T$. Then solve the master problem with the optimum decision $\hat{x}_{t-1}$ for stage t = 1, …, T as shown in Equation 4-59 and store the optimum decision variables (x_t) of current iteration as $\hat{x}_t$, which will be used for additional Benders cuts in the backward pass. α_{t+1}^{k-1} is the approximated cost of the next stage from the last iteration.

$$\hat{x}_t = \underset{F_t x_t \geq s_t - E_{t-1}\hat{x}_{t-1}}{\operatorname{argmin}} [c_t x_t + \alpha_{t+1}^{k-1}(x_t, s_t)] \quad 4\text{-}59$$

- Step 5: Proceed to backward recursion. For stage t = T, …,2, solving the optimization problem as Equation 4-60 indicates to obtain optimal value $\mathfrak{Q}_{t,s_t}$ and dual solution $\pi^{\hat{x}_{t-1},s_t}$ corresponding to $\hat{x}_{t-1}^k$ (the trial solution found from forward pass master problem at time *t*-1, iteration *k*) for each Markov state value ($s_t \in S_t$) in stage *t*. Then, for each Markov state value ($s_{t-1} \in S_{t-1}$) in stage *t*-1, calculate the expected optimal value $\mathfrak{Q}_{t,s_{t-1}}$ and expected dual solution $\bar{\pi}_t^{s_{t-1}}$ based on Equation 4-61 and 4-62. Then add additional Benders cut $\theta(x_t, s_{t-1})$ for next time step as indicated in Equation 4-63.

$$\mathfrak{Q}_{t,s_t} = \min_{F_t x_t \geq s_t - E_{t-1}\hat{x}_{t-1}} [c_t x_t + \alpha_{t+1}^{k}(x_t, s_t)] \quad 4\text{-}60$$

$$\mathfrak{Q}_{t,s_{t-1}} = \sum_{s_t \in S_t} Pr_{s_{t-1},s_t}\mathfrak{Q}_{t,s_t} \quad 4\text{-}61$$

$$\bar{\pi}_t^{s_{t-1}} = \sum_{s_t \in S_t} Pr_{s_{t-1},s_t} \pi^{\hat{x}_{t-1},s_t} \quad 4\text{-}62$$

$$\alpha_t^k(x_t, s_{t-1}) = \{\theta \in \alpha_t^{k-1}(x_t, s_{t-1}), \theta(x_t, s_{t-1}) \geq \bar{\pi}_t^{s_{t-1}}(x - \hat{x}_{t-1}) + \Omega_{t,s_{t-1}}\} \quad 4\text{-}63$$

- Step 6: The policy value and lower boundary can be calculated based on the computed value and approximation of future cost from Equations 4-64 and 4-65, respectively.

$$\bar{z}^k \geq \sum_{t=1}^{T} \hat{x}_t^k \quad 4\text{-}64$$

$$\underline{z}^k = \min_{F_1 x_1 \geq s_1 - E_0 \hat{x}_0} [c_1 x_1 + \alpha_2^k(x_1, s_1)] \quad 4\text{-}65$$

- Step 7: Increase the iteration count *k* to *k*+1 and return to step 2.

In this thesis, integer variables are involved in the operation optimization of the CCHP system. The SDDP should be replaced by stochastic dual dynamic integer programming (SDDiP) algorithm to solve this multistage stochastic programming problem. The details of how to generalize SDDiP from SDDP can be viewed in Zou's research paper (Zou, Ahmed, and Sun 2019) and Ding's thesis (Ding 2020). Strengthened Benders cuts are used to deal with integer variables due to their superiority compared with the original Benders cut.

4.4.2.3 SDDP for CCHP system

Figure 4-15 shows the simulation results of operational costs change along with iteration. The fluctuated upper bound is smoothed by fitting the policy array through a

convex and monotone decreasing function, as shown in Equation 4-66 (Löhndorf and Shapiro 2019). y_i and $\hat{y}_i$ are smoothed and original policy value, respectively. The deterministic bound shows a stable value around \$1,330 and smoothed policy values reaches to \$1,372. The potential annual cost is about 22.33 \$/m^2 when considering the floor area. The difference between deterministic bound and smoothed policy value gradually and stably decreases to the range of tolerance the credibility of convergence. In addition, the relatively small gap between bound and policy value yields an optimal operation cost value for the CCHP system.

$$\min \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 \qquad 4\text{-}66$$

$$y_i \leq y_{i-1}$$

$$2y_{i-1} \leq y_i + y_{i-2}$$

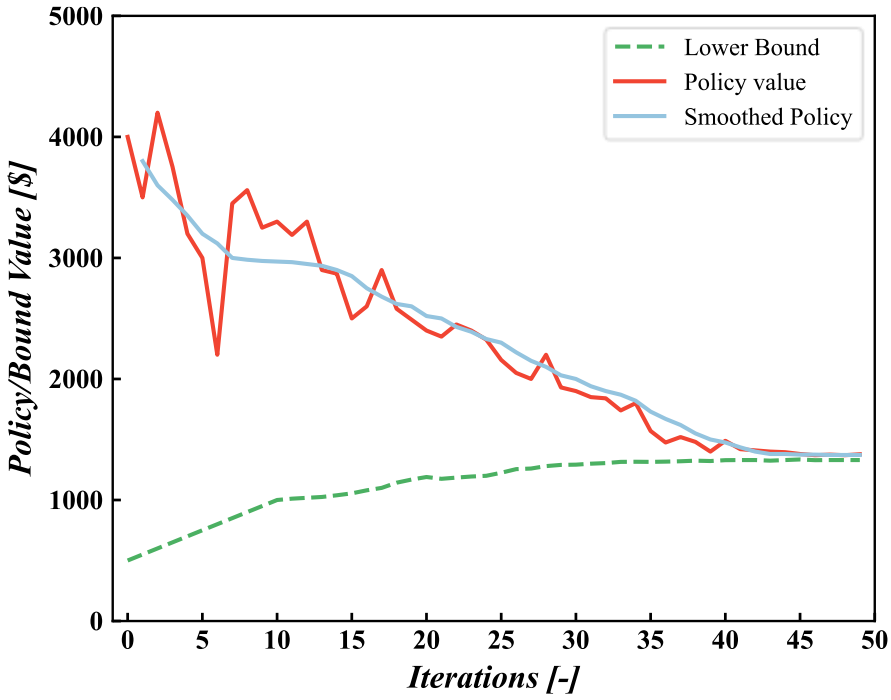

Figure 4-15. The boundary of SDDP along with iteration

Table 4-5. The simulation results by changing the uncertainty sampling

MCA #	MCA time [s]	Solution time [s]	Lower bound [$]	Policy value [$]	Gap [%]
50	275	77	1,330	1,372	3.2
75	1,347	128	1,353	1,410	4.2
100	3,792	161	1,378	1,441	4.5

The discretization number of uncertainty has been explored to investigate the influence of discretization of uncertainty on operation optimization performance. The result is shown in Table 4-5. It is observed that solving process time increases almost linearly with the discretization from an efficient computational point of view, and the time consumption falls within an acceptable range. The optimality gap grows with discretization, and the bound remains relatively stable.

4.5 Summary

The energy consumption of a hospital with a floor area of 22,422 m^2 located at College Park, Maryland, is forecasted by the ML model of Chapter 2 for analyzing the operation performance. A general graph representation is developed for the CCHP system. CCHP systems representation can be easily expanded or depressed by adding or deleting nodes and edges to the original graph representation. The mathematical expression that conveys the energy flow can then be formulated based on the listed graph representation. The CCHP system is first assumed to operate without uncertainty. Load-following strategies and optimization-based methods were implemented for the CCHP system as a preliminary view and comparison. The result shows that optimization-based operation strategies outperform prescheduled FHL for almost all the months in a year. The storage in the CCHP system can mitigate the fluctuation of subsystems and provide a smoother operation stack with the system without storage.

Moreover, the thermal storages have a higher impact on the fluctuation of these subsystems than the electric storage. With the help of storage subsystems, the capacity of the primary mover and waste heat recovery subsystem should be decreased to diminish the capital cost. The capacity of thermally driven generation subsystems should be increased to further boost the subsystems' performance.

As for system operation with uncertainty, the sampled hospital energy demands are provided to an autoregression and stochastic approximation combined model for training the Markov chain process. The combined model was validated through the original forecasted dataset. The sampling result shows its credibility for extracting uncertainty of building energy demands. The recursive approach is first investigated with the discretized Markov state and probability transaction matrix. The system operation cost shows a trend with the increase of discretization number for storage. However, the recursive way is trapped in computational inefficiency before adopting a fine and appropriate discretization of storage subsystems. Therefore, the operation process for CCHP with uncertainty cannot take full advantage of storage, such as energy cost reduction and fluctuation mitigation. Stochastic dual dynamic programming is introduced and utilized to approximate the original problem. The SDDP is implemented for operation optimization and yields an optimal daily cost of \$1,372, and a potential annual cost is 22.33 \$/m^2. The convergence and optimum gap information demonstrated the feasibility of applying an SDDP-based algorithm for CCHP optimization with uncertainty. Moreover, the result indicates computation complexity increases linearly with the discretization of uncertain information.

5 Conclusions and Future Work

5.1 Conclusions

Due to the complexity of CCHP systems and environmental uncertainty, conventional design and operation strategies that depend on expertise or experience might lose effectiveness and protract the prototyping process. This thesis utilized automation-oriented approaches, including ML and optimization, to accelerate decision-making for CCHP systems, from demand planning to system design and operation optimization.

The performances of two DL models were compared with the non-DL model and discussed in terms of absolute and relative error in building types and different tasks aspects. A principal component analysis (PCA) was combined with the output of the internal layer to illustrate the performance of the LSTM model. The hyperparameters, including layers, and input steps in the DL model, were investigated, and the influence of loss function on multiple prediction tasks has been analyzed in terms of accuracy and computational efficiency. Recommendations for the model, hyperparameters, and loss function were provided for this multi vectors prediction on multiple building types.

A graph theory knowledge-based configuration representation approach is developed with the requirement of basic thermodynamic laws. A simulated annealing-based configuration optimization framework is proposed. The framework is validated by optimizing S-CO_2 power generation system configurations with simple and complex component number limitations. The influence of initial configuration on the effectiveness and convergence of the framework is explored.

The energy consumption of a hospital with a floor area of 22,422 m^2 located at College Park, Maryland, is forecasted by the ML model of Chapter 2 for analyzing the operation performance. An autoregression and stochastic approximation combined model is implemented for training a Markov chain process. Stochastic dynamic programming and approximation approaches are applied for CCHP system operation.

5.2 Contributions

The contributions of this thesis are as follows:

1) The energy forecast task fills the gap in DL models' short-term and detailed prediction for multiple building types. Two types of DL models (CNN and LSTM) have been constructed aimed at multiple energy types prediction for multi-building types simultaneously for the first time. Overall, the performances of the LSTM model on most of the building types are better than the CNN model. The LSTM model is more suitable for this multiple energy prediction. The long short-term memory (LSTM) model forecasts 50.7% of the tasks with a coefficient of variation of root mean square error (CVRMSE) lower than 20%. Moreover, 60% of the tasks predicted by LSTM satisfy ASHRAE Guideline 14 with a CVRMSE under 30%. Other feature engineering techniques or embedding methods that can differentiate sufficient building characteristics should be compared for predicting multiple types of buildings.

2) The design optimization framework is the first to implement graphic knowledge and basic thermodynamic laws to generate new CO_2 (S-CO_2) power system configurations. It is validated by optimizing the S-CO_2 power generation system's configurations under simple and complex component number limitations. The optimized S-CO_2 power

system reaches 49.8% thermal efficiency. This efficiency is 2.3% higher than the state of the art.

3) A flexible graph-based representation is constructed for simplified CCHP systems. The optimization-based operation overperforms the best load-following strategy by 14% of the annual cost. The autoregression and stochastic approximation combined model was validated through the original forecasted dataset. The approximated dynamic programming algorithm SDDP is used to make the operation optimization process computationally tractable. The result shows that the daily operation cost is about $1,372, and a potential annual cost is 22.33 $/m^2 for the hospital at College Park, Maryland.

The thesis resulted in the following relevant publications:

1) Lei Gao, Yunho Hwang, and Tao Cao. "An overview of optimization technologies applied in combined cooling, heating and power systems." Renewable and Sustainable Energy Reviews 114 (2019): 109344.
2) Jingye Yang, Lei Gao, Zhenhong Ye, Yunho Hwang, Jiangping Chen. "Binary-objective optimization of latest low-GWP alternatives to R245fa for organic Rankine cycle application." Energy (2020): 119336.
3) Lei Gao, Tao Cao, Yunho Hwang, Reinhard Radermacher. "Graph-based configuration optimization for S-CO2 power generation systems." Energy Conversion and Management 244 (2021): 114448.

4) Lei Gao, Tianyuan Liu, Tao Cao, Yunho Hwang, Reinhard Radermacher. "Comparing deep learning models for multi energy vectors prediction on multiple types of building." Applied Energy 301 (2021): 117486.
5) Lei Gao, Tao Cao, Yunho Hwang, Reinhard Radermacher. "Robustness analysis in supercritical CO2 power generation system configuration optimization." Energy 204 (2022): 122956.
6) Lin, Lingnan, Lei Gao, Mark A Kedzierski, Yunho Hwang. "A general model for flow boiling heat transfer in microfin tubes based on a new neural network architecture." Energy and AI (2022): 100151.
7) Lei Gao, Yunho Hwang, Gyeong Sung Kim, "Thermoeconomic Analysis of a Solar MVC Desalination System." November 05, 2018. ASME International Mechanical Engineering Congress and Exposition, Pittsburgh, Pennsylvania, USA
8) Tao Cao, Lei Gao, Vikrant C Aute, Yunho Hwang. "A Data-driven Model Development for Generalized Building Energy Predictions." July 2021 International High Performance Buildings Conference, Purdue, Indiana, USA.
9) Lei Gao, Yunho Hwang, Reinhard Radermacher. Stochastic dynamic optimal operation for combined cooling, heating, and power system with uncertainty. Energy Conversion and Management. (in preparation)

5.3 Future work

Based on this thesis, the future potential investigation directions are as follows:

1) For building energy forecasts, the state-of-the-art deep learning model should be examined and compared, and the interpretability of the machine learning model in

building energy tasks should be explored. It is preferred to construct ML models that not purely data driven but also can interpret physical meaning, like heat mass transfer, thermodynamics, etc. The prediction by ML model would be more robust and reasonable when considering physical information of the buildings.

2) For the system design optimization part, all three aspects, as shown in Figure 3-5, should be investigated more: a) for the configuration optimizer: a more detailed exploration of the hyperparameters and different metaheuristic/heuristic approaches should be investigated; b) for the configuration generation, other generating techniques can be included, a data-driven way might enable intelligent algorithm (machine learning) to obtain new configuration efficiently; and c) for configuration solver, a regression machine learning model is a viable method that would accelerate whole optimization process.

3) For system operation optimization, a) the linear representation of subsystems needs to be upgraded to a nonlinear case to approximate more realistic subsystems' performance; b) Multiple optimization metrics should be considered objectives, such as emission and primary energy consumption; c) the risk aversion model should also be proposed for operation under uncertainty. Constructing a modeling platform that can efficiently simulate the whole life span of a CCHP system under different risk preferences.

Appendix

Equations A1 to A5 mathematically describe the T-graph of a recompression S-CO2 power generation system. Specifically, in Eq. A1, Ω describes a set of available component types. V represents vertices in the T-graph. Eq. A2 describes the mapping process from vertex (V) in the T-graph to the set of available components (Ω). Eq. A3 describes the sets of the inlet and outlet connectors for every vertex. The first element of every inlet-connector and the second element of every outlet-connector in the set represent the vertex's inlet and outlet edge number, respectively. Eq. A4 illustrates the paired connector at the ends of one edge. The vertex set (V), connector set (C^{in} and C^{out}) and edge set (E) will be reindexed and updated when inserting or deleting a component in the configuration. Eq. A5 describes the mapping process from the connector (C^{in} and C^{out}) to the thermodynamic properties (mass, pressure, temperature/enthalpy).

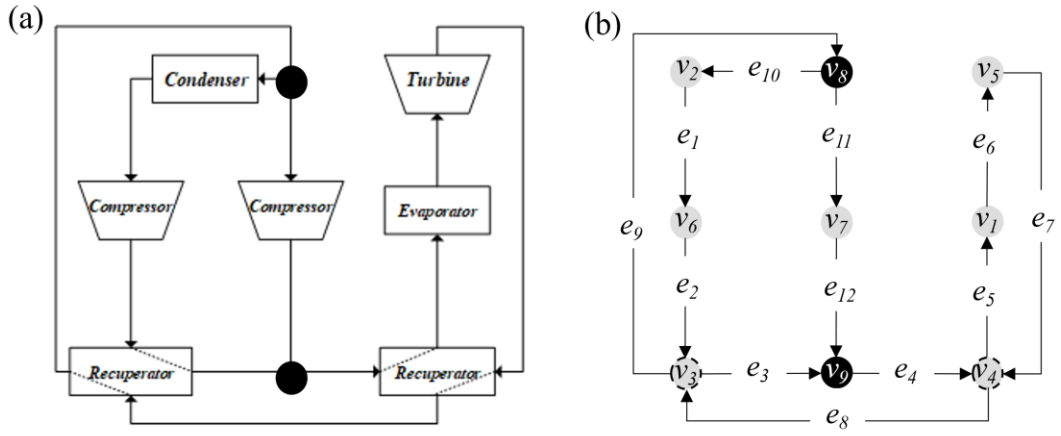

Figure A1. Recompression cycle representation: (a) original cycle; (b) T-graph representation

Features of object-oriented programming language are utilized to construct classes of vertex and configuration. This construction helps the programming process because: once a class is established, same properties (like pressure ratio of vertex or component of configuration) and method (find path, find component, evaluate efficiency, etc.) can be used repeatedly for same type of class by initiating an instance. Any systems can be represented by vertices and edges as expressed in thermodynamic graph. These vertices and edges should contain all the required information and methods that used to evaluate the thermodynamic system performance. Once a vertex instance is initiated from vertex class, it contains all required information includes vertices type shown in Eq. A2, the inlet and outlet state shown in Eq. A3. And the vertex also includes the method that mapping outlet conditions from the inlet conditions as shown in Eq. A5. Similarly, a configuration instance is initiated from the configuration class and contains all vertex instances in a configuration. This configuration instance also contains the connection of different vertices shown in Eq. A4. It has methods to find path of any two vertices, the component types of a path, and evaluate the system performance based on the connection. It also contains the methods that are used to implement the mutation of optimization process, such as insertion and deletion. That is to say, the connection expressed in Eq. A4 can be manipulated by these methods. It is convenient to evaluate and iterate a system in the optimization framework with these properties and methods.

$$\begin{cases} \Omega \coloneqq \{\text{Turbine}, \text{Compressor}, \text{HX}_{\text{Source}}, \text{HX}_{\text{Sink}}, \text{HX}_{\text{Recu}}, \text{Split}, \text{Merge}\} \\ V \coloneqq \{v_1, v_2, \dots, v_9\}, V_f \coloneqq \emptyset \end{cases} \tag{A1}$$

$$\begin{cases} \omega^v(v_1) \coloneqq \text{HX}_{\text{Source}} \\ \omega^v(v_2) \coloneqq \text{HX}_{\text{Sink}} \\ \omega^v(v_3) \coloneqq \text{HX}_{\text{Re}} \\ \omega^v(v_4) \coloneqq \text{HX}_{\text{Re}} \\ \omega^v(v_5) \coloneqq \text{Turbine} \\ \omega^v(v_6) \coloneqq \text{Compressor} \\ \omega^v(v_7) \coloneqq \text{Compressor} \\ \omega^v(v_8) \coloneqq \text{Split} \\ \omega^v(v_9) \coloneqq \text{Merge} \end{cases} \tag{A2}$$

$$\begin{cases} C^{in} \coloneqq \{(1, v_i) \ \ for\ i \in \{1, \dots, 9\} \cup (2, v_i) \ \text{ for } i \in \{3,4,9\}\} \\ C^{out} \coloneqq \{(v_i, 1) \ \ for\ i \in \{1, \dots, 9\} \cup (v_i, 2) \ \text{ for } i \in \{3,4,8\}\} \end{cases} \tag{A3}$$

$$\begin{cases} E \coloneqq \{e_1, \dots, e_{12}\} \\ e_1 \coloneqq (c_2^{out}, c_6^{in}) \\ e_2 \coloneqq (c_6^{out}, c_3^{in}) \\ e_3 \coloneqq (c_3^{out}, c_9^{in}) \\ e_4 \coloneqq (c_9^{out}, c_4^{in}) \\ e_5 \coloneqq (c_4^{out}, c_1^{in}) \\ e_6 \coloneqq (c_1^{out}, c_5^{in}) \\ e_7 \coloneqq (c_5^{out}, c_{11}^{in}) \\ e_8 \coloneqq \left(c_{11}^{out}, c_{10}^{in}\right) \\ e_9 \coloneqq \left(c_{10}^{out}, c_8^{in}\right) \\ e_{10} \coloneqq \left(c_8^{out}, c_2^{in}\right) \\ e_{11} \coloneqq \left(c_{12}^{out}, c_7^{in}\right) \\ e_{12} \coloneqq (c_7^{out}, c_{12}^{in}) \end{cases} \tag{A4}$$

$$\begin{cases} m^{in}, p^{in}, T^{in} \coloneqq \pi^c\left(c^{in}\right) \text{ for } c^{in} \in C^{in} \\ m^{out}, p^{out}, T^{out} \coloneqq \pi^c(c^{out}) \ \text{ for } c^{out} \in C^{out} \\ f^{\pi}\left(\pi^c\left(c_i^{in}\right)\right) = \pi^c(c_i^{out}) \ \text{ for } i \ \{1,2,5,6,7\} \\ f^{\pi}\left(\pi^c\left(c_3^{in}\right), \pi^c\left(c_{10}^{in}\right)\right) = \pi^c(c_3^{out}), \pi^c(c_{10}^{out}) \\ f^{\pi}\left(\pi^c\left(c_4^{in}\right), \pi^c\left(c_{11}^{in}\right)\right) = \pi^c(c_4^{out}), \pi^c(c_{11}^{out}) \\ f^{\pi}\left(\pi^c\left(c_9^{in}\right), \pi^c\left(c_{12}^{in}\right)\right) = \pi^c(c_9^{out}) \\ f^{\pi}\left(\pi^c\left(c_8^{in}\right)\right) = \pi^c(c_8^{out}), \pi^c(c_{12}^{out}) \end{cases} \tag{A5}$$

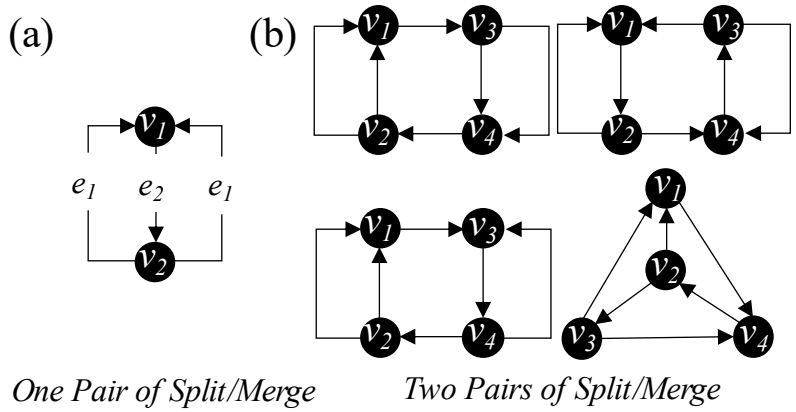

Figure A2. Permutation of possible configuration

Two steps can manually achieve any SC graphs. First, make a connection within only splits/merges (three-edge vertices) of a system without considering other components (two-edge vertices). Second, insert other components (two-edge vertices). The final decoupled configuration with only splits/merges (three-edge vertices) can be viewed in Figure A2 when the maximum number of splits is limited to two. It should be noticed that the isomorphism graph is not considered.

Table A1. Permutation of one split/merge and two splits/merges

Component #	Component on three edges (Figure A2 a)	Component on six different edges	Component on six same edges
3	6	224	12
4	9	504	20
5	12	1008	28
6	16	1848	44
7	20	3168	56
8	25	5148	80

The configuration number of a system without split/merge is simple. The connection number is to count the type of N different components form a circle, which is A_{N-1}^{N-1} compared with A_N^N if not strongly connected is required. For one split/merge (Figure a) or two splits/merges (Figure b), we first treat every component the same and place N

of them on edges and then do full permutation (A_N^N) for all the N components (Identical objects into identical bins). For single split/merge and low component number volume (<10), we can manually find the type of place the same component on three edges with two of them the same, which is listed in the second column in Table A1. It is not straightforward for more split/merge to do the manual calculation. Therefore, we try to find the upper and lower limits based on whether these edges are the same. The total type number, when considering these edges are different, is C_{N+M-1}^M, and when the edge is the same, we consider the partitions of an integer problem ("Partition (Number Theory)" 2020; "Identical Objects into Identical Bins | Brilliant Math & Science Wiki" 2020). Then, we can calculate the total type of placing the component on the same edge.

Bibliography

Afzali, Sayyed Faridoddin, and Vladimir Mahalec. 2018. “Novel Performance Curves to Determine Optimal Operation of CCHP Systems.” *Applied Energy* 226 (September): 1009–36. https://doi.org/10.1016/j.apenergy.2018.06.024.

Ahn, Yoonhan, Seong Jun Bae, Minseok Kim, Seong Kuk Cho, Seungjoon Baik, Jeong Ik Lee, and Jae Eun Cha. 2015. “Review of Supercritical CO2 Power Cycle Technology and Current Status of Research and Development.” *Nuclear Engineering and Technology* 47 (6): 647–61. https://doi.org/10.1016/j.net.2015.06.009.

Amasyali, Kadir, and Nora M. El-Gohary. 2018. “A Review of Data-Driven Building Energy Consumption Prediction Studies.” *Renewable and Sustainable Energy Reviews* 81 (January): 1192–1205. https://doi.org/10.1016/j.rser.2017.04.095.

Artzner, Philippe, Freddy Delbaen, Jean-Marc Eber, and David Heath. 1999. “Coherent Measures of Risk.” *Mathematical Finance* 9 (3): 203–28. https://doi.org/10.1111/1467-9965.00068.

ASHRAE. 2020. “ASHRAE Guideline 14-2014 - Measurement of Energy, Demand, and Water Savings.” June 20, 2020. https://webstore.ansi.org/standards/ashrae/ashraeguideline142014.

Bae, Seong Jun, Yoonhan Ahn, Jekyoung Lee, and Jeong Ik Lee. 2014. “Various Supercritical Carbon Dioxide Cycle Layouts Study for Molten Carbonate Fuel Cell Application.” *Journal of Power Sources* 270 (December): 608–18. https://doi.org/10.1016/j.jpowsour.2014.07.121.

Bao, Junjiang, Ruixiang Zhang, Yan Lin, Ning Zhang, Xiaopeng Zhang, and Gaohong He. 2018. “Simultaneous Optimization of System Structure and Working Fluid for the Three-Stage Condensation Rankine Cycle Utilizing LNG Cold Energy.” *Applied Thermal Engineering* 140 (July): 120–30. https://doi.org/10.1016/j.applthermaleng.2018.05.049.

Bao, Junjiang, Ruixiang Zhang, Tong Yuan, Xiaopeng Zhang, Ning Zhang, and Gaohong He. 2018. “A Simultaneous Approach to Optimize the Component and Composition of Zeotropic Mixture for Power Generation Systems.” *Energy Conversion and Management* 165 (June): 354–62. https://doi.org/10.1016/j.enconman.2018.03.072.

Bazaraa, Mokhtar S., John J. Jarvis, and Hanif D. Sherali. 2009. *Linear Programming and Network Flows: Bazaraa/Linear*. Hoboken, NJ, USA: John Wiley & Sons, Inc. https://doi.org/10.1002/9780471703778.

Bell, Ian H., Jorrit Wronski, Sylvain Quoilin, and Vincent Lemort. 2014. “Pure and Pseudo-Pure Fluid Thermophysical Property Evaluation and the Open-Source Thermophysical Property Library CoolProp.” *Industrial & Engineering Chemistry Research* 53 (6): 2498–2508. https://doi.org/10.1021/ie4033999.

Birge, John R., and François Louveaux. 2011. *Introduction to Stochastic Programming*. Springer Series in Operations Research and Financial Engineering. New York, NY: Springer New York. https://doi.org/10.1007/978-1-4614-0237-4.

Bourdeau, Mathieu, Xiao qiang Zhai, Elyes Nefzaoui, Xiaofeng Guo, and Patrice Chatellier. 2019. “Modeling and Forecasting Building Energy Consumption: A

Review of Data-Driven Techniques." *Sustainable Cities and Society* 48 (July): 101533. https://doi.org/10.1016/j.scs.2019.101533.

Brun, Klaus, Peter Friedman, and Richard Dennis. 2017. *Fundamentals and Applications of Supercritical Carbon Dioxide (SCO2) Based Power Cycles*. Woodhead Publishing.

Cao, Tao, Yunho Hwang, and Reinhard Radermacher. 2017. "Development of an Optimization Based Design Framework for Microgrid Energy Systems." *Energy* 140 (December): 340–51. https://doi.org/10.1016/j.energy.2017.08.120.

Cedillos Alvarado, Dagoberto, Salvador Acha, Nilay Shah, and Christos N. Markides. 2016. "A Technology Selection and Operation (TSO) Optimisation Model for Distributed Energy Systems: Mathematical Formulation and Case Study." *Applied Energy* 180 (October): 491–503. https://doi.org/10.1016/j.apenergy.2016.08.013.

Chae, Young Tae, Raya Horesh, Youngdeok Hwang, and Young M. Lee. 2016. "Artificial Neural Network Model for Forecasting Sub-Hourly Electricity Usage in Commercial Buildings." *Energy and Buildings* 111 (January): 184–94. https://doi.org/10.1016/j.enbuild.2015.11.045.

Chen, Tianqi, and Carlos Guestrin. 2016. "XGBoost: A Scalable Tree Boosting System." In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 785–94. San Francisco California USA: ACM. https://doi.org/10.1145/2939672.2939785.

Chou, Jui-Sheng, and Dac-Khuong Bui. 2014. "Modeling Heating and Cooling Loads by Artificial Intelligence for Energy-Efficient Building Design." *Energy and Buildings* 82 (October): 437–46. https://doi.org/10.1016/j.enbuild.2014.07.036.

Conejo, Antonio J., Miguel Carrión, and Juan M. Morales. 2010. *Decision Making Under Uncertainty in Electricity Markets*. Vol. 153. International Series in Operations Research & Management Science. Boston, MA: Springer US. https://doi.org/10.1007/978-1-4419-7421-1.

Crawley, Drury B., Linda K. Lawrie, Frederick C. Winkelmann, W. F. Buhl, Y. Joe Huang, Curtis O. Pedersen, Richard K. Strand, et al. 2001. "EnergyPlus: Creating a New-Generation Building Energy Simulation Program." *Energy and Buildings*, Special Issue: BUILDING SIMULATION'99, 33 (4): 319–31. https://doi.org/10.1016/S0378-7788(00)00114-6.

Crespi, Francesco, Giacomo Gavagnin, David Sánchez, and Gonzalo S. Martínez. 2017. "Supercritical Carbon Dioxide Cycles for Power Generation: A Review." *Applied Energy* 195 (June): 152–83. https://doi.org/10.1016/j.apenergy.2017.02.048.

Culaba, Alvin B., Aaron Jules R. Del Rosario, Aristotle T. Ubando, and Jo-Shu Chang. 2020. "Machine Learning-based Energy Consumption Clustering and Forecasting for Mixed-use Buildings." *International Journal of Energy Research* 44 (12): 9659–73. https://doi.org/10.1002/er.5523.

Ding, Lingquan. 2020. "Multistage Stochastic Programming," May. https://smartech.gatech.edu/handle/1853/63591.

Domingos, Pedro. 2015. *The Master Algorithm: How the Quest for the Ultimate Learning Machine Will Remake Our World*. New York: Basic Books, a member of the Perseus Books Group.

Domínguez, Ruth, Sebastiano Vitali, Miguel Carrión, and Vittorio Moriggia. 2021. "Analysing Decarbonizing Strategies in the European Power System Applying Stochastic Dominance Constraints." *Energy Economics* 101 (September): 105438. https://doi.org/10.1016/j.eneco.2021.105438.

Dong, Bing, Cheng Cao, and Siew Eang Lee. 2005. "Applying Support Vector Machines to Predict Building Energy Consumption in Tropical Region." *Energy and Buildings* 37 (5): 545–53. https://doi.org/10.1016/j.enbuild.2004.09.009.

Dostal, Vaclav, Pavel Hejzlar, and Michael J. Driscoll. 2006. "High-Performance Supercritical Carbon Dioxide Cycle for Next-Generation Nuclear Reactors." *Nuclear Technology* 154 (3): 265–82. https://doi.org/10.13182/NT154-265.

Elsido, Cristina, Alberto Mian, and Emanuele Martelli. 2017. "A Systematic Methodology for the Techno-Economic Optimization of Organic Rankine Cycles." *Energy Procedia* 129 (September): 26–33. https://doi.org/10.1016/j.egypro.2017.09.171.

Emmerich, Michael, Monika Grötzner, and Martin Schütz. 2001. "Design of Graph-Based Evolutionary Algorithms: A Case Study for Chemical Process Networks." *Evolutionary Computation* 9 (3): 329–54. https://doi.org/10.1162/106365601750406028.

Fan, Cheng, Yongjun Sun, Fu Xiao, Jie Ma, Dasheng Lee, Jiayuan Wang, and Yen Chieh Tseng. 2020. "Statistical Investigations of Transfer Learning-Based Methodology for Short-Term Building Energy Predictions." *Applied Energy* 262 (March): 114499. https://doi.org/10.1016/j.apenergy.2020.114499.

Fan, Cheng, Yongjun Sun, Yang Zhao, Mengjie Song, and Jiayuan Wang. 2019. "Deep Learning-Based Feature Engineering Methods for Improved Building Energy Prediction." *Applied Energy* 240 (April): 35–45. https://doi.org/10.1016/j.apenergy.2019.02.052.

Fan, Cheng, Fu Xiao, Chengchu Yan, Chengliang Liu, Zhengdao Li, and Jiayuan Wang. 2019. "A Novel Methodology to Explain and Evaluate Data-Driven Building Energy Performance Models Based on Interpretable Machine Learning." *Applied Energy* 235 (February): 1551–60. https://doi.org/10.1016/j.apenergy.2018.11.081.

Fang, F., Q. H. Wang, and Y. Shi. 2012. "A Novel Optimal Operational Strategy for the CCHP System Based on Two Operating Modes." *IEEE Transactions on Power Systems* 27 (2): 1032–41. https://doi.org/10.1109/TPWRS.2011.2175490.

Farmani, Farid, Mehdi Parvizimosaed, Hassan Monsef, and Ashkan Rahimi-Kian. 2018. "A Conceptual Model of a Smart Energy Management System for a Residential Building Equipped with CCHP System." *International Journal of Electrical Power & Energy Systems* 95 (February): 523–36. https://doi.org/10.1016/j.ijepes.2017.09.016.

Fekri, Mohammad Navid, Ananda Mohon Ghosh, and Katarina Grolinger. 2020. "Generating Energy Data for Machine Learning with Recurrent Generative Adversarial Networks." *Energies* 13 (1): 130. https://doi.org/10.3390/en13010130.

Gao, Lei. 2020. “Evolution of S-CO2 Recompression Cycle - Lei Gao / Ph.D. Student.” December 27, 2020. https://leigao-ceee.github.io/portfolio/evolution_SCO2/.

Gao, Lei, Yunho Hwang, and Tao Cao. 2019. “An Overview of Optimization Technologies Applied in Combined Cooling, Heating and Power Systems.” *Renewable and Sustainable Energy Reviews* 114 (October): 109344. https://doi.org/10.1016/j.rser.2019.109344.

Gao, Lei, Tianyuan Liu, Tao Cao, Yunho Hwang, and Reinhard Radermacher. 2021. “Comparing Deep Learning Models for Multi Energy Vectors Prediction on Multiple Types of Building.” *Applied Energy* 301 (November): 117486. https://doi.org/10.1016/j.apenergy.2021.117486.

Garcia, Ramon Ferreiro, Jose Carbia Carril, Javier Romero Gomez, and Manuel Romero Gomez. 2016. “Energy and Entropy Analysis of Closed Adiabatic Expansion Based Trilateral Cycles.” *Energy Conversion and Management* 119 (July): 49–59. https://doi.org/10.1016/j.enconman.2016.04.031.

Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. 2016. *Deep Learning*. MIT Press.

Grekas, Denis N., and Christos A. Frangopoulos. 2007. “Automatic Synthesis of Mathematical Models Using Graph Theory for Optimisation of Thermal Energy Systems.” *Energy Conversion and Management*, 19th International Conference on Efficiency, Cost, Optimization, Simulation and Environmental Impact of Energy Systems, 48 (11): 2818–26. https://doi.org/10.1016/j.enconman.2007.06.044.

Gu, Wei, Yiyuan Tang, Shuyong Peng, Delin Wang, Wanxing Sheng, and Keyan Liu. 2015. “Optimal Configuration and Analysis of Combined Cooling, Heating, and Power Microgrid with Thermal Storage Tank under Uncertainty.” *Journal of Renewable and Sustainable Energy* 7 (1): 013104. https://doi.org/10.1063/1.4904434.

Gu, Wei, Zhi Wu, Rui Bo, Wei Liu, Gan Zhou, Wu Chen, and Zaijun Wu. 2014. “Modeling, Planning and Optimal Energy Management of Combined Cooling, Heating and Power Microgrid: A Review.” *International Journal of Electrical Power & Energy Systems* 54 (January): 26–37. https://doi.org/10.1016/j.ijepes.2013.06.028.

Gunn, Steve R. 1998. “Support Vector Machines for Classification and Regression,” 66.

Guo, Tinghao, Daniel Herber, and James T. Allison. 2019. “Circuit Synthesis Using Generative Adversarial Networks (GANs).” In *AIAA Scitech 2019 Forum*. San Diego, California: American Institute of Aeronautics and Astronautics. https://doi.org/10.2514/6.2019-2350.

Hajabdollahi, Hassan. 2015. “Evaluation of Cooling and Thermal Energy Storage Tanks in Optimization of Multi-Generation System.” *Journal of Energy Storage* 4 (December): 1–13. https://doi.org/10.1016/j.est.2015.08.004.

Harris, Charles R., K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, et al. 2020. “Array Programming with NumPy.” *Nature* 585 (7825): 357–62. https://doi.org/10.1038/s41586-020-2649-2.

Hastie, Trevor, Robert Tibshirani, and Jerome Friedman. 2009. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition*. Springer Science & Business Media.

Hawkins, D., S. M. Hong, R. Raslan, D. Mumovic, and S. Hanna. 2012. "Determinants of Energy Use in UK Higher Education Buildings Using Statistical and Artificial Neural Network Methods." *International Journal of Sustainable Built Environment* 1 (1): 50–63. https://doi.org/10.1016/j.ijsbe.2012.05.002.

Hemmati, Reza, Hedayat Saboori, and Saeid Saboori. 2016. "Stochastic Risk-Averse Coordinated Scheduling of Grid Integrated Energy Storage Units in Transmission Constrained Wind-Thermal Systems within a Conditional Value-at-Risk Framework." *Energy* 113 (October): 762–75. https://doi.org/10.1016/j.energy.2016.07.089.

Hendron, R, and C Engebrecht. 2010. "Building America House Simulation Protocols (Revised)." NREL/TP-550-49246, DOE/GO-102010-3141, 989422. https://doi.org/10.2172/989422.

Hensman, James, Alex Matthews, and Zoubin Ghahramani. 2014. "Scalable Variational Gaussian Process Classification." arXiv. https://doi.org/10.48550/arXiv.1411.2005.

Heo, Yeonsook, and Victor M. Zavala. 2012. "Gaussian Process Modeling for Measurement and Verification of Building Energy Savings." *Energy and Buildings* 53 (October): 7–18. https://doi.org/10.1016/j.enbuild.2012.06.024.

"Identical Objects into Identical Bins | Brilliant Math & Science Wiki." 2020. December 17, 2020. https://brilliant.org/wiki/identical-objects-into-identical-bins/.

Iglesias Garcia, Steven, Ramon Ferreiro Garcia, Jose Carbia Carril, and Denis Iglesias Garcia. 2017. "Critical Review of the First-Law Efficiency in Different Power Combined Cycle Architectures." *Energy Conversion and Management* 148 (September): 844–59. https://doi.org/10.1016/j.enconman.2017.06.037.

Jabari, Farkhondeh, Sayyad Nojavan, and Behnam Mohammadi Ivatloo. 2016. "Designing and Optimizing a Novel Advanced Adiabatic Compressed Air Energy Storage and Air Source Heat Pump Based μ-Combined Cooling, Heating and Power System." *Energy* 116 (December): 64–77. https://doi.org/10.1016/j.energy.2016.09.106.

James, Gareth, Daniela Witten, Trevor Hastie, and Robert Tibshirani. 2013. *An Introduction to Statistical Learning: With Applications in R*. Springer Science & Business Media.

Javadi, Mohammad Sadegh, Ali Esmaeel Nezhad, Matthew Gough, Mohamed Lotfi, Amjad Anvari-Moghaddam, Pedro H. J. Nardelli, Subham Sahoo, and João P. S. Catalão. 2021. "Conditional Value-at-Risk Model for Smart Home Energy Management Systems." *E-Prime - Advances in Electrical Engineering, Electronics and Energy* 1 (January): 100006. https://doi.org/10.1016/j.prime.2021.100006.

Ji, Ling, Bei-Bei Zhang, Guo-He Huang, Yu-Lei Xie, and Dong-Xiao Niu. 2018. "Explicit Cost-Risk Tradeoff for Optimal Energy Management in CCHP Microgrid System under Fuzzy-Risk Preferences." *Energy Economics* 70: 525–35. https://doi.org/10.1016/j.eneco.2018.01.017.

Jian, Peiru, Tiantong Guo, Dan Wang, Esmaeil Valipour, and Sayyad Nojavan. 2022. "Risk-Based Energy Management of Industrial Buildings in Smart Cities and Peer-to-Peer Electricity Trading Using Second-Order Stochastic Dominance Procedure." *Sustainable Cities and Society* 77 (February): 103550. https://doi.org/10.1016/j.scs.2021.103550.

Jiang, Ning, Wenqiao Han, Fengyuan Guo, Hangsheng Yu, Yingjie Xu, and Ning Mao. 2018. "A Novel Heat Exchanger Network Retrofit Approach Based on Performance Reassessment." *Energy Conversion and Management* 177 (December): 477–92. https://doi.org/10.1016/j.enconman.2018.10.001.

Kalogirou, Soteris A., and Milorad Bojic. 2000. "Artificial Neural Networks for the Prediction of the Energy Consumption of a Passive Solar Building." *Energy* 25 (5): 479–91. https://doi.org/10.1016/S0360-5442(99)00086-9.

Kim, Tae-Young, and Sung-Bae Cho. 2019. "Predicting Residential Energy Consumption Using CNN-LSTM Neural Networks." *Energy* 182 (September): 72–81. https://doi.org/10.1016/j.energy.2019.05.230.

Kingma, Diederik P., and Jimmy Ba. 2017. "Adam: A Method for Stochastic Optimization." *ArXiv:1412.6980 [Cs]*, January. http://arxiv.org/abs/1412.6980.

Kirkpatrick, S., C. D. Gelatt, and M. P. Vecchi. 1983. "Optimization by Simulated Annealing." *Science* 220 (4598): 671–80. https://doi.org/10.1126/science.220.4598.671.

Klemeš, Jiří Jaromír, Qiu-Wang Wang, Petar Sabev Varbanov, Min Zeng, Hon Huin Chin, Nathan Sanjay Lal, Nian-Qi Li, Bohong Wang, Xue-Chao Wang, and Timothy Gordon Walmsley. 2020. "Heat Transfer Enhancement, Intensification and Optimisation in Heat Exchanger Network Retrofit and Operation." *Renewable and Sustainable Energy Reviews* 120 (March): 109644. https://doi.org/10.1016/j.rser.2019.109644.

Koschwitz, D., J. Frisch, and C. van Treeck. 2018. "Data-Driven Heating and Cooling Load Predictions for Non-Residential Buildings Based on Support Vector Machine Regression and NARX Recurrent Neural Network: A Comparative Study on District Scale." *Energy* 165 (December): 134–42. https://doi.org/10.1016/j.energy.2018.09.068.

Kuang, Jiyuan, Chenghui Zhang, and Bo Sun. 2019. "Stochastic Dynamic Solution for Off-Design Operation Optimization of Combined Cooling, Heating, and Power Systems with Energy Storage." *Applied Thermal Engineering* 163 (December): 114356. https://doi.org/10.1016/j.applthermaleng.2019.114356.

Lazzaretto, A., and F. Segato. 2001. "Thermodynamic Optimization of the HAT Cycle Plant Structure—Part I: Optimization of the 'Basic Plant Configuration.'" *Journal of Engineering for Gas Turbines and Power* 123 (1): 1–7. https://doi.org/10.1115/1.1338999.

Lee, Ung, Jeongwoo Jeon, Chonghun Han, and Youngsub Lim. 2017. "Superstructure Based Techno-Economic Optimization of the Organic Rankine Cycle Using LNG Cryogenic Energy." *Energy* 137 (October): 83–94. https://doi.org/10.1016/j.energy.2017.07.019.

Li, Ming-Jia, Han-Hui Zhu, Jia-Qi Guo, Kun Wang, and Wen-Quan Tao. 2017. "The Development Technology and Applications of Supercritical CO2 Power Cycle in Nuclear Energy, Solar Energy and Other Energy Industries." *Applied*

Thermal Engineering 126 (November): 255–75. https://doi.org/10.1016/j.applthermaleng.2017.07.173.

Li, Xiwang, and Jin Wen. 2014. "Review of Building Energy Modeling for Control and Operation." *Renewable and Sustainable Energy Reviews* 37 (September): 517–37. https://doi.org/10.1016/j.rser.2014.05.056.

Li, Zhenning, Vikrant Aute, and Jiazhen Ling. 2019. "Tube-Fin Heat Exchanger Circuitry Optimization Using Integer Permutation Based Genetic Algorithm." *International Journal of Refrigeration* 103 (July): 135–44. https://doi.org/10.1016/j.ijrefrig.2019.04.006.

Liu, Wei, Guanyi Chen, Beibei Yan, Zhihua Zhou, Haowei Du, and Jian Zuo. 2015. "Hourly Operation Strategy of a CCHP System with GSHP and Thermal Energy Storage (TES) under Variable Loads: A Case Study." *Energy and Buildings* 93 (April): 143–53. https://doi.org/10.1016/j.enbuild.2015.02.030.

LLNL. 2020. "LLNL Flow Charts." Energy Flow Charts: Charting the Complex Relationships among Energy, Water, and Carbon. June 30, 2020. https://flowcharts.llnl.gov/.

Löhndorf, Nils, and Alexander Shapiro. 2019. "Modeling Time-Dependent Randomness in Stochastic Dual Dynamic Programming." *European Journal of Operational Research* 273 (2): 650–61. https://doi.org/10.1016/j.ejor.2018.08.001.

Lu, Yongxi, Abhishek Kumar, Shuangfei Zhai, Yu Cheng, Tara Javidi, and Rogerio Feris. 2016. "Fully-Adaptive Feature Sharing in Multi-Task Networks with Applications in Person Attribute Classification." *ArXiv:1611.05377 [Cs]*, November. http://arxiv.org/abs/1611.05377.

Mago, Pedro J., and Anna K. Hueffed. 2010. "Evaluation of a Turbine Driven CCHP System for Large Office Buildings under Different Operating Strategies." *Energy and Buildings* 42 (10): 1628–36. https://doi.org/10.1016/j.enbuild.2010.04.005.

Mago, P.J., and L.M. Chamra. 2009. "Analysis and Optimization of CCHP Systems Based on Energy, Economical, and Environmental Considerations." *Energy and Buildings* 41 (10): 1099–1106. https://doi.org/10.1016/j.enbuild.2009.05.014.

Mago, P.J., L.M. Chamra, and J. Ramsay. 2010. "Micro-Combined Cooling, Heating and Power Systems Hybrid Electric-Thermal Load Following Operation." *Applied Thermal Engineering* 30 (8–9): 800–806. https://doi.org/10.1016/j.applthermaleng.2009.12.008.

Marino, Carlos, Mohammad Marufuzzaman, Mengqi Hu, and M. D. Sarder. 2018. "Developing a CCHP-Microgrid Operation Decision Model under Uncertainty." *Computers & Industrial Engineering* 115: 354–67. https://doi.org/10.1016/j.cie.2017.11.021.

Marino, Daniel L., Kasun Amarasinghe, and Milos Manic. 2016. "Building Energy Load Forecasting Using Deep Neural Networks." In *IECON 2016 - 42nd Annual Conference of the IEEE Industrial Electronics Society*, 7046–51. https://doi.org/10.1109/IECON.2016.7793413.

Mencarelli, Luca, Qi Chen, Alexandre Pagot, and Ignacio E. Grossmann. 2020. "A Review on Superstructure Optimization Approaches in Process System

Engineering." *Computers & Chemical Engineering* 136 (May): 106808. https://doi.org/10.1016/j.compchemeng.2020.106808.

Montoya-Bueno, Sergio, Jos Ignacio Muoz, and Javier Contreras. 2015. "A Stochastic Investment Model for Renewable Generation in Distribution Systems." *IEEE Transactions on Sustainable Energy* 6 (4): 1466–74. https://doi.org/10.1109/TSTE.2015.2444438.

Mu, Yunfei, Congshan Wang, Yan Cao, Hongjie Jia, Qingzhu Zhang, and Xiaodan Yu. 2022. "A CVaR-Based Risk Assessment Method for Park-Level Integrated Energy System Considering the Uncertainties and Correlation of Energy Prices." *Energy* 247 (May): 123549. https://doi.org/10.1016/j.energy.2022.123549.

Neises, T., and C. Turchi. 2014. "A Comparison of Supercritical Carbon Dioxide Power Cycle Configurations with an Emphasis on CSP Applications." *Energy Procedia*, Proceedings of the SolarPACES 2013 International Conference, 49 (January): 1187–96. https://doi.org/10.1016/j.egypro.2014.03.128.

Newham, Nikki. 2008. "Power System Investment Planning Using Stochastic Dual Dynamic Programming." University of Canterbury. Electrical and Computer Engineering. https://ir.canterbury.ac.nz/handle/10092/1975.

Nikolaev, Alexander G., and Sheldon H. Jacobson. 2010. "Simulated Annealing." In *Handbook of Metaheuristics*, edited by Michel Gendreau and Jean-Yves Potvin, 1–39. International Series in Operations Research & Management Science. Boston, MA: Springer US. https://doi.org/10.1007/978-1-4419-1665-5_1.

OpenEI. 2020. "Energy Information and Data | OpenEI.Org." June 20, 2020. https://data.openei.org/submissions/153.

Park, Hyo Seon, Minhyun Lee, Hyuna Kang, Taehoon Hong, and Jaewook Jeong. 2016. "Development of a New Energy Benchmark for Improving the Operational Rating System of Office Buildings Using Various Data-Mining Techniques." *Applied Energy* 173 (July): 225–37. https://doi.org/10.1016/j.apenergy.2016.04.035.

"Partition (Number Theory)." 2020. Wikipedia. December 27, 2020. https://en.wikipedia.org/w/index.php?title=Partition_(number_theory)&oldid=996545662.

Paszke, Adam, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. "Automatic Differentiation in PyTorch," October. https://openreview.net/forum?id=BJJsrmfCZ.

Pereira, M. V. F., and L. M. V. G. Pinto. 1991. "Multi-Stage Stochastic Optimization Applied to Energy Planning." *Mathematical Programming* 52 (1–3): 359–75. https://doi.org/10.1007/BF01582895.

Price, Alkes L, Nick J Patterson, Robert M Plenge, Michael E Weinblatt, Nancy A Shadick, and David Reich. 2006. "Principal Components Analysis Corrects for Stratification in Genome-Wide Association Studies." *Nature Genetics* 38 (8): 904–9. https://doi.org/10.1038/ng1847.

Rahman, Aowabin, Vivek Srikumar, and Amanda D. Smith. 2018. "Predicting Electricity Consumption for Commercial and Residential Buildings Using Deep Recurrent Neural Networks." *Applied Energy* 212 (February): 372–85. https://doi.org/10.1016/j.apenergy.2017.12.051.

Rathor, Sumit K., and D. Saxena. 2020. “Energy Management System for Smart Grid: An Overview and Key Issues.” *International Journal of Energy Research* 44 (6): 4067–4109. https://doi.org/10.1002/er.4883.

Regenwetter, Lyle, Amin Heyrani Nobari, and Faez Ahmed. 2022. “Deep Generative Models in Engineering Design: A Review.” *ArXiv:2110.10863 [Cs, Stat]*, March. http://arxiv.org/abs/2110.10863.

Robinson, Caleb, Bistra Dilkina, Jeffrey Hubbs, Wenwen Zhang, Subhrajit Guhathakurta, Marilyn A. Brown, and Ram M. Pendyala. 2017. “Machine Learning Approaches for Estimating Commercial Building Energy Consumption.” *Applied Energy* 208 (December): 889–904. https://doi.org/10.1016/j.apenergy.2017.09.060.

Ruan, Yingjun, Qingrong Liu, Zhengwei Li, and Jiazheng Wu. 2016. “Optimization and Analysis of Building Combined Cooling, Heating and Power (BCHP) Plants with Chilled Ice Thermal Storage System.” *Applied Energy* 179 (October): 738–54. https://doi.org/10.1016/j.apenergy.2016.07.009.

Ruder, Sebastian. 2017. “An Overview of Multi-Task Learning in Deep Neural Networks.” *ArXiv:1706.05098 [Cs, Stat]*, June. http://arxiv.org/abs/1706.05098.

Sanaye, Sepehr, and Behzad Niroomand. 2007. “Simulation of Heat Exchanger Network (HEN) and Planning the Optimum Cleaning Schedule.” *Energy Conversion and Management* 48 (5): 1450–61. https://doi.org/10.1016/j.enconman.2006.12.006.

Seyedzadeh, Saleh, Farzad Pour Rahimian, Stephen Oliver, Sergio Rodriguez, and Ivan Glesk. 2020. “Machine Learning Modelling for Predicting Non-Domestic Buildings Energy Performance: A Model to Support Deep Energy Retrofit Decision-Making.” *Applied Energy* 279 (December): 115908. https://doi.org/10.1016/j.apenergy.2020.115908.

Seyedzadeh, Saleh, Farzad Pour Rahimian, Parag Rastogi, and Ivan Glesk. 2019. “Tuning Machine Learning Models for Prediction of Building Energy Loads.” *Sustainable Cities and Society* 47 (May): 101484. https://doi.org/10.1016/j.scs.2019.101484.

Shapiro, Alexander, Darinka Dentcheva, and Andrzej Ruszczyński. 2009. *Lectures on Stochastic Programming: Modeling and Theory*. Society for Industrial and Applied Mathematics. https://doi.org/10.1137/1.9780898718751.

Shapiro, Alexander, and Andy Philpott. n.d. “A Tutorial on Stochastic Programming,” 35.

Somu, Nivethitha, Gauthama Raman M r, and Krithi Ramamritham. 2020. “A Hybrid Model for Building Energy Consumption Forecasting Using Long Short Term Memory Networks.” *Applied Energy* 261 (March): 114131. https://doi.org/10.1016/j.apenergy.2019.114131.

Sun, Ying, Fariborz Haghighat, and Benjamin C. M. Fung. 2020. “A Review of The-State-of-the-Art in Data-Driven Approaches for Building Energy Prediction.” *Energy and Buildings* 221 (August): 110022. https://doi.org/10.1016/j.enbuild.2020.110022.

Toffolo, A., S. Rech, and A. Lazzaretto. 2018. “Combination of Elementary Processes to Form a General Energy System Configuration.” In . American Society of

Mechanical Engineers Digital Collection. https://doi.org/10.1115/IMECE2017-71653.

Toffolo, Andrea. 2014. “A Synthesis/Design Optimization Algorithm for Rankine Cycle Based Energy Systems.” *Energy* 66 (March): 115–27. https://doi.org/10.1016/j.energy.2014.01.070.

Toimil, Daniel, and Alberto Gómez. 2017. “Review of Metaheuristics Applied to Heat Exchanger Network Design.” *International Transactions in Operational Research* 24 (1–2): 7–26. https://doi.org/10.1111/itor.12296.

Turchi, Craig S., Zhiwen Ma, Ty Neises, and Michael Wagner. 2012. “Thermodynamic Study of Advanced Supercritical Carbon Dioxide Power Cycles for High Performance Concentrating Solar Power Systems.” In , 375. ASME. https://doi.org/10.1115/ES2012-91179.

Virtanen, Pauli, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, et al. 2020. “SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python.” *Nature Methods* 17 (3): 261–72. https://doi.org/10.1038/s41592-019-0686-2.

Voll, Philip. 2014. *Automated optimization-based synthesis of distributed energy supply systems*. 1. Aufl. Aachener Beiträge zur Technischen Thermodynamik 1. Aachen: Wissenschaftsverl. Mainz.

Walker, Shalika, Waqas Khan, Katarina Katic, Wim Maassen, and Wim Zeiler. 2020. “Accuracy of Different Machine Learning Algorithms and Added-Value of Predicting Aggregated-Level Energy Performance of Commercial Buildings.” *Energy and Buildings* 209 (February): 109705. https://doi.org/10.1016/j.enbuild.2019.109705.

Wang, Jiang-Jiang, You-Yin Jing, Chun-Fa Zhang, Guo-Hua Shi, and Xu-Tao Zhang. 2008. “A Fuzzy Multi-Criteria Decision-Making Model for Trigeneration System.” *Energy Policy* 36 (10): 3823–32. https://doi.org/10.1016/j.enpol.2008.07.002.

Wang, Ligang, Yongping Yang, Changqing Dong, Tatiana Morosuk, and George Tsatsaronis. 2014. “Parametric Optimization of Supercritical Coal-Fired Power Plants by MINLP and Differential Evolution.” *Energy Conversion and Management* 85 (September): 828–38. https://doi.org/10.1016/j.enconman.2014.01.006.

Wang, Ligang, Zhiping Yang, Shivom Sharma, Alberto Mian, Tzu-En Lin, George Tsatsaronis, François Maréchal, and Yongping Yang. 2019. “A Review of Evaluation, Optimization and Synthesis of Energy Systems: Methodology and Application to Thermal Power Plants.” *Energies* 12 (1): 73. https://doi.org/10.3390/en12010073.

Wang, Shengwei, and Zhenjun Ma. 2008. “Supervisory and Optimal Control of Building HVAC Systems: A Review.” *HVAC&R Research* 14 (1): 3–32. https://doi.org/10.1080/10789669.2008.10390991.

Wang, Zeyu, and Ravi S. Srinivasan. 2017. “A Review of Artificial Intelligence Based Building Energy Use Prediction: Contrasting the Capabilities of Single and Ensemble Prediction Models.” *Renewable and Sustainable Energy Reviews* 75 (August): 796–808. https://doi.org/10.1016/j.rser.2016.10.079.

Wang, Zhe, Tianzhen Hong, and Mary Ann Piette. 2020. “Building Thermal Load Prediction through Shallow Machine Learning and Deep Learning.” *Applied Energy* 263 (April): 114683. https://doi.org/10.1016/j.apenergy.2020.114683.

Westermann, Paul, Chirag Deb, Arno Schlueter, and Ralph Evins. 2020. “Unsupervised Learning of Energy Signatures to Identify the Heating System and Building Type Using Smart Meter Data.” *Applied Energy* 264 (April): 114715. https://doi.org/10.1016/j.apenergy.2020.114715.

Wilcox, S, and W Marion. 2008. “Users Manual for TMY3 Data Sets.” *Technical Report*, 58.

Winkler, Jonathan, Vikrant Aute, and Reinhard Radermacher. 2008. “*Comprehensive Investigation of Numerical Methods in Simulating a Steady-State Vapor Compression System.” *International Journal of Refrigeration* 31 (5): 930–42. https://doi.org/10.1016/j.ijrefrig.2007.08.008.

Wright, Jonathan, and Yi Zhang. 2008. “Evolutionary Synthesis of HVAC System Configurations: Experimental Results.” *HVAC&R Research* 14 (1): 57–72. https://doi.org/10.1080/10789669.2008.10390993.

Wright, Jonathan, Yi Zhang, Plamen Angelov, Victor Hanby, and Richard Buswell. 2008. “Evolutionary Synthesis of HVAC System Configurations: Algorithm Development (RP-1049).” *HVAC&R Research* 14 (1): 33–55. https://doi.org/10.1080/10789669.2008.10390992.

Wright, Steven, M. Thomas, and Rochau Gary. 2011. “Supercritical CO2 Power Cycle Development Summary at Sandia National Laboratories” No. SAND2011-6640C: 29.

Wu, D.W., and R.Z. Wang. 2006. “Combined Cooling, Heating and Power: A Review.” *Progress in Energy and Combustion Science* 32 (5–6): 459–95. https://doi.org/10.1016/j.pecs.2006.02.001.

Xu, Lei, Shengwei Wang, and Rui Tang. 2019. “Probabilistic Load Forecasting for Buildings Considering Weather Forecasting Uncertainty and Uncertain Peak Load.” *Applied Energy* 237 (March): 180–95. https://doi.org/10.1016/j.apenergy.2019.01.022.

Xuan, Ang, Xinwei Shen, Qinglai Guo, and Hongbin Sun. 2021. “A Conditional Value-at-Risk Based Planning Model for Integrated Energy System with Energy Storage and Renewables.” *Applied Energy* 294 (July): 116971. https://doi.org/10.1016/j.apenergy.2021.116971.

Yalcintas, Melek, and U. Aytun Ozturk. 2007. “An Energy Benchmarking Model Based on Artificial Neural Network Method Utilizing US Commercial Buildings Energy Consumption Survey (CBECS) Database.” *International Journal of Energy Research* 31 (4): 412–21. https://doi.org/10.1002/er.1232.

Yang, Jingye, Lei Gao, Zhenhong Ye, Yunho Hwang, and Jiangping Chen. 2021. “Binary-Objective Optimization of Latest Low-GWP Alternatives to R245fa for Organic Rankine Cycle Application.” *Energy* 217 (February): 119336. https://doi.org/10.1016/j.energy.2020.119336.

Yee, T. F., and I. E. Grossmann. 1990. “Simultaneous Optimization Models for Heat Integration—II. Heat Exchanger Network Synthesis.” *Computers & Chemical Engineering* 14 (10): 1165–84. https://doi.org/10.1016/0098-1354(90)85010-8.

Yeomans, Hector, and Ignacio E. Grossmann. 1999. “A Systematic Modeling Framework of Superstructure Optimization in Process Synthesis.” *Computers & Chemical Engineering* 23 (6): 709–31. https://doi.org/10.1016/S0098-1354(99)00003-4.

Yu, Haoshui, John Eason, Lorenz T. Biegler, and Xiao Feng. 2017. “Process Integration and Superstructure Optimization of Organic Rankine Cycles (ORCs) with Heat Exchanger Network Synthesis.” *Computers & Chemical Engineering*, In honor of Professor Rafiqul Gani, 107 (December): 257–70. https://doi.org/10.1016/j.compchemeng.2017.05.013.

Yu, Zhun, Fariborz Haghighat, Benjamin C. M. Fung, and Hiroshi Yoshino. 2010. “A Decision Tree Method for Building Energy Demand Modeling.” *Energy and Buildings* 42 (10): 1637–46. https://doi.org/10.1016/j.enbuild.2010.04.006.

Yun, Geun Young, Hyo Joo Kong, Hyoin Kim, and Jeong Tai Kim. 2012. “A Field Survey of Visual Comfort and Lighting Energy Consumption in Open Plan Offices.” *Energy and Buildings*, Sustainable and healthy buildings, 46 (March): 146–51. https://doi.org/10.1016/j.enbuild.2011.10.035.

Zhang, Xin, Kwong Fai Fong, and Shiu Yin Yuen. 2013. “A Novel Artificial Bee Colony Algorithm for HVAC Optimization Problems.” *HVAC&R Research* 19 (6): 715–31. https://doi.org/10.1080/10789669.2013.803915.

Zhang, Xu, Hao Bai, Xiancong Zhao, Ali Diabat, Jian Zhang, Huanmei Yuan, and Zefei Zhang. 2018. “Multi-Objective Optimisation and Fast Decision-Making Method for Working Fluid Selection in Organic Rankine Cycle with Low-Temperature Waste Heat Source in Industry.” *Energy Conversion and Management* 172 (September): 200–211. https://doi.org/10.1016/j.enconman.2018.07.021.

Zhang, Yu, and Qiang Yang. 2018. “A Survey on Multi-Task Learning.” *ArXiv:1707.08114 [Cs]*, July. http://arxiv.org/abs/1707.08114.

Zhao, Hai-xiang, and Frédéric Magoulès. 2012. “A Review on the Prediction of Building Energy Consumption.” *Renewable and Sustainable Energy Reviews* 16 (6): 3586–92. https://doi.org/10.1016/j.rser.2012.02.049.

Zheng, C.Y., J.Y. Wu, and X.Q. Zhai. 2014. “A Novel Operation Strategy for CCHP Systems Based on Minimum Distance.” *Applied Energy* 128 (September): 325–35. https://doi.org/10.1016/j.apenergy.2014.04.084.

Zheng, C.Y., J.Y. Wu, X.Q. Zhai, and R.Z. Wang. 2017. “A Novel Thermal Storage Strategy for CCHP System Based on Energy Demands and State of Storage Tank.” *International Journal of Electrical Power & Energy Systems* 85 (February): 117–29. https://doi.org/10.1016/j.ijepes.2016.08.008.

Zheng, Xuyue, Guoce Wu, Yuwei Qiu, Xiangyan Zhan, Nilay Shah, Ning Li, and Yingru Zhao. 2018. “A MINLP Multi-Objective Optimization Model for Operational Planning of a Case Study CCHP System in Urban China.” *Applied Energy* 210 (January): 1126–40. https://doi.org/10.1016/j.apenergy.2017.06.038.

Zhong, Hai, Jiajun Wang, Hongjie Jia, Yunfei Mu, and Shilei Lv. 2019. “Vector Field-Based Support Vector Regression for Building Energy Consumption Prediction.” *Applied Energy* 242 (May): 403–14. https://doi.org/10.1016/j.apenergy.2019.03.078.

Zhou, Yizhou, Zhinong Wei, Guoqiang Sun, Kwok W. Cheung, Haixiang Zang, and Sheng Chen. 2018. “A Robust Optimization Approach for Integrated Community Energy System in Energy and Ancillary Service Markets.” *Energy* 148: 1–15. https://doi.org/10.1016/j.energy.2018.01.078.

Zou, Jikai, Shabbir Ahmed, and Xu Andy Sun. 2019. “Stochastic Dual Dynamic Integer Programming.” *Mathematical Programming* 175 (1): 461–502. https://doi.org/10.1007/s10107-018-1249-5.