Multiscale modeling in mathematical programming: Application of Clustering

by

Falah Alhameli

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Chemical Engineering

Waterloo, Ontario, Canada, 2017

Examining Committee Membership

The following served on the Examining Committee for this thesis. The decision of the Examining Committee is by majority vote.

External Examiner	Prof. Hossam Gaber Professor, Energy Systems and Nuclear Science University of Ontario, Institute of Technology
Supervisor	Prof. Ali Elkamel Professor, Chemical Engineering
Internal Members	Prof. Peter Douglas Professor, Chemical Engineering
	Prof. Yuri A.W. Shardt Professor, Chemical Engineering
Internal-external Member	Prof. Ramadan El-Shatshat Professor, Electrical & Computer Engineering

Author's Declaration

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Integration across decision levels of a supply chain is a key point in improving returns on investment. For example, planning and scheduling are usually carried out separately although they are interdependent of each other. Integration of planning and scheduling results in better coordination between decision levels and a reduction in operating costs. Integration of different time scales leads to large scale problems which are usually computationally intractable. Different approaches have been proposed to tackle the problem in terms of modeling and solution methods. However, most of them are problem specific or applicable only to short time horizons. Clustering has the potential to handle such a problem by grouping similar input parameters (like demand or price) together. This will considerably shrink the model size and make it more computationally tractable while at the same time not compromising solution accuracy.

Therefore, the aim of this thesis is to develop a new class of clustering algorithms that are based on mathematical programming techniques in order to support the integration of planning applications of different time scales (strategic, tactical, and operational) in process systems engineering. The clustering algorithms were formulated using integer programming with IAE (integral absolute error) as a similarity measure. The initial formulation was a Mixed Integer Nonlinear Program (MINLP) and then reduced to a Mixed Integer Linear Program (MILP) using exact linearization techniques. The model resulted in two different clustering algorithms: normal and sequence clustering. Two case studies were presented to assess outputs and computational performance of the algorithms. Electricity demand and solar radiation data were clustered in these case studies. Both clustering algorithms captured the trend in the data. However, the computational burden of the model was prohibitive to tackle large planning horizons.

In order to deal with computational complexity, a heuristic algorithm was developed utilizing an iterative scheme. The heuristic was first applied to clustering the electricity demand in the original cases studied for validation purposes. The quality of the solutions from the heuristic algorithm were checked against the MILP optimal solutions and it was found that the heuristic algorithm is able to provide good quality solutions and even succeeded in finding the optimal solution for simulation runs carried out. The heuristic algorithm was applied to clustering the electricity demand for a whole year with a small computational effort and providing clusters with high intra-cluster similarity and low inter-cluster similarity.

In order to illustrate the use of the clustering procedure in solving large scale planning model, the clustered electricity demand was used as input to a Unit Commitment (UC) model with the objective to

evaluate the solution quality when clustered demand is applied. The UC problem is a classical problem in electrical power production where the production of a set of electrical generators is coordinated in order to meet the energy demand at minimum cost or maximize revenues from energy production. The results showed a great advantage in term of solution time for the clustering technique compared to the regular solution when no clustering of demand was applied. Moreover, the error of objective function was within 0.5 % of the non-clustered demand for all cases. In addition, a sensitivity analysis study suggested that high quality solutions could still be achieved with smaller number of clusters.

The clustering algorithm was extended in order to take into account multiple attributes at the same time such as clustering simultaneously demand for electricity and heat. In this respect, the objective function had different scales due to the different units of measurements of the attributes, and the problem was dealt with as a multi-objective optimization problem. The weighting method was chosen as the optimization approach and to be able to appropriately scale the different attributes. The clustering algorithm was successfully applied to simultaneously cluster hourly electricity and heat demands for the whole year. The Pareto front was captured for all runs with the weight factor combinations considered in this study. The results show that a better objective function is achieved when the number of clusters increases for both normal and sequence clustering. Normal clustering and as expected leads to a better objective function, error average and standard deviation than sequential clustering due to the additional restrictions of sequencing requirements imposed on the model. Clusters that take into account the time of occurrence of events and abide to certain minimum sequencing restrictions are also needed in planning operations in order to minimize the number of set-ups and inconvenience to operators. The statistical analysis of the heat demand was challenging as suggested by the results, due to the huge fluctuation in the heat demand. Moreover, calculations of relative error were problematic for the demand that was close to zero. The results indicated that in the case when operations are flexible or in the case of just classifying demand patterns, normal clustering should be used since it has a major advantage in terms of solution quality over sequence clustering. For the case of simultaneously clustering heat and electricity, it was required to employ many clusters of electricity that sometimes overlap with each other. These clusters could not be merged since they correspond to different days and the clusters of heat demand for these days are different. Nevertheless, the proposed algorithm was able to obtain groups that simultaneously cluster the two attributes and hence can provide computational advantages when solving integrated planning models that deal with more than one demand attribute.

The clustered electricity and heat demands were used as inputs to an energy hub model, with the objective of evaluating the solution quality when multiple clustered demand attributes are applied to planning models. The average error of objective function was -1.7 % for normal clustering while for sequence clustering it was -4.2 %. Increasing the number of clusters was found to enhance the solution quality for both normal and sequence clustering. For this particular example, varying the weight factors did not have a drastic effect on the values of the objective function. This is due mainly to a symmetry or inverse similarity in the heat and electricity demands.

Acknowledgements

All the praises are due to Allah, the Most Beneficent and the Most Merciful for blessing me with the strength and patience to seek knowledge and accomplish this work.

I would like to express my sincere gratitude to my advisor Prof. Ali Elkamel for the continuous support of my PhD study and related research, for his patience, motivation, and immense knowledge. His guidance helped me in all the time of research and writing of this thesis. I could not have imagined having a better advisor and mentor for my PhD study.

Besides my advisor, I would like to thank the rest of my thesis committee: Prof. Peter Douglas, Prof. Hossam Gaber, Prof. Ramandan El Shatshat, and Prof. Yuri A. W. Shardt, for their insightful comments and encouragement, but also for the hard questions which incented me to widen my research from various perspectives.

I'm profoundly indebted to my wife for her unwavering support and continuous patience during my PhD study. I would also like to extend my gratitude to my parents, parents-in-law, and siblings for all the emotional support they provided me throughout writing this thesis and my life in general.

Last but not the least, I would like to acknowledge and extend my heartfelt gratitude to my country, United Arab Emirates, and my academic sponsor, ADNOC (Abu Dhabi National Oil Company), for the financial support.

Dedication

To my beloved Wife, Parents, Parents-in-law, Family, and Country

Table of Contents

List of Figures

List of Tables

Nomenclature

Sets

a	Attributes (in case of one attribute clustering, the set of a is dropped)
c	Clusters
d	Days
h	Hours
i	Unit
t	Time period

Parameters

a_i, b_i	Coefficients of the fuel cost function of unit *i*
b	Unit conversion for natural gas
Csc_i	Cold start costs of unit *i*
D_t	Power load demand for time period *t*
Dc_i	Shut-down cost of unit *i*
DR_i	Ramp-down limit of unit *i*
η_{CHP}^{elec}	Electrical efficiency for CHP
η_{CHP}^{heat}	Thermal efficiency for CHP
η_{boiler}^{heat}	Thermal efficiency for boiler
Hsc_i	Hot start cost of unit *i*
$L_{a,d,h}$	Demand load of attribute *a* for *h* hour in *d* day
$L_{a,h}^{L}$	Lower bound of attribute *a* load for hour *h*
$L_{a,h}^{U}$	Upper bound of attribute *a* load for hour *h*
$L_{d,h}^{elec}$	Hourly electricity demand
$L_{d,h}^{heat}$	Hourly heat demand

Max_{boiler}	Max capacity of boiler
Max_{CHP}	Max capacity of CHP
N_t	Number of repetitions (frequency) for corresponding *t* time
OM_{boiler}	Operation and maintenance cost for boiler
OM_{CHP}	Operation and maintenance cost for CHP
P_i^L	Minimum power generation of unit *i*
P_i^U	Maximum power generation of unit *i*
$Price_h^{Grid}$	Hourly electricity price from grid
$Price_{NG}$	Natural gas price
R_t	Spinning reserve required at time period *t*
SD_i	Maximum shutdown rate of unit *i*
SU_i	Maximum start-up rate of unit *i*
T_i^{ini}	Initial status of unit *i*
T_i^{cold}	Cold start hours of unit *i*
TU_i	Minimum uptime of unit *i*
TD_i	Minimum downtime of unit *i*
UR_i	Ramp-up limit of unit *i*

Positive Variables

$CD_{i,t}$	Shut-down cost of unit *i* in period *t*
$Cl_{a,c,h}$	Demand for hour *h* in cluster c and attribute *a*
$CU_{i,t}$	Start-up cost of unit *i* in period *t*
$ELEC_{d,h}^{Grid}$	Electricity from grid in *d* day and *h* hour
$ELEC_{d,h}^{CHP}$	Electricity from CHP in *d* day and *h* hour
$HEAT_{d,h}^{Boiler}$	Heat from boiler in *d* day and *h* hour
$HEAT_{d,h}^{CHP}$	Heat from CHP in *d* day and *h* hour

$I_{a,d,h}$	Absolute difference between *L* and *C* for hour *h* in *d* day of attribute *a*
$NG_{d,h}^{CHP}$	Natural gas for CHP in *d* day and h hour
$NG_{d,h}^{Boiler}$	Natural gas for boiler in *d* day and *h* hour
$P_{a,h,d,c}$	Relaxation variable
$P_{i,t}$	Power output of unit *i* in period *t*

Binary Variables

$U_{i,t}$	Binary variable representing the on/off status of unit *i* at period *t*
$x_{d,c}$	Assignment of load for day *d* joining cluster *c*, 0 otherwise

Constants

Δ	Grid spacing for trapezoidal rule (for hourly discretized curve, $\Delta = 1$)
I	Total number of thermal generating units
T	Length of the planning time horizon

Chapter 1
Introduction

Supply chain management and optimization have proven their superior performance in driving profits while maintaining customer satisfaction[1]. The goal of supply chain management is to more efficiently coordinate the components of supply chain, such as production and transportation, as the complexity of modern facilities reaches a new level when corporations go beyond borders. There are three decision levels in supply chain management; strategic, tactical and operational. The strategic decision level deals with the structure of the supply chain, like location and technology selection. The tactical decisions are concerned with setting production targets and ensuring transportation between the facilities. The operation level is involved in task assignments and sequence on a daily or weekly basis.

Researchers and practitioners used to carry out these decisions on a sequential basis[2], with strategic decisions carried out first, then tactical decisions, and so on. All these decision levels depend on each other and have different time scales. Therefore, integration between these decision levels is a key point to reach improved efficiency and profit margins[3]. The planning decisions at the strategic (long-term), tactical (medum-term), and operational (short-term) should be made simultaneously in order to achieve a global optimum. Strategic decisions are concerned with determining the structure of the supply chain, tactical planning focuses on decisions that pertain for example to assignment of production targets to facilities and the transportation and distribution problem, while operational planning or scheduling deals with day to day or weekly operations. Because of the difference in time scales of the three components of the supply chain management problem, the integration usually leads to a multiscale model which is computationally intractable. Converting the different time scales in this multiscale model to the shortest planning period (i.e. detailed scheduling over a long horizon) leads to very large and intractable problems. Relaxing constraints or employing surrogate models to create support vectors (i.e. via off-line simulation or historical data) might lead to infeasible operations (i.e. detailed schedules cannot be obtained to meet the planned production targets).

In this thesis we aim at tackling the integrated supply chain model covering strategic, tactical, and operational decisions through the use of a clustering approach. Since utilizing a shorter time period (e.g. hour) to obtain optimal decisions leads to large intractable models, we aim at reducing model size through representing the yearly days by "typical" days that are representative of the year of operation. Clustering in this context focuses on classifying periods of operation in different groups with similar demand patterns and characteristics. Although clustering has been used extensively in many applications, clustering of demand patterns has been little studied and is more complex because it is a multi-dimensional problem that also deals with shape (i.e. trajectory) of the hourly demand curves and also often times has different attributes (e.g. simultaneous demand for electricity and heat in an energy hub). Furthermore, we take here a mathematical programming based approach and formulate the clustering problem first as a Mixed Integer Nonlinear Program (MINLP) and then use appropriate linearization schemes to obtain an exact linearization of the model and render it to a Mixed Integer Linear Program (MILP).

Power planning and management has been known to be a difficult problem to tackle computationally, especially with the increased use of intermittent renewable energy generation which adds more variability to the input parameters of the planning model. Running such large models under different scenarios in order to deal with uncertainty or performing sensitivity analysis studies magnifies the intractability of the planning model. Therefore, the proposed clustering model in this thesis will be illustrated on a model from the power sector to illustrate the advantages in the reduction of computational burden and at the same time maintaining accuracy of the obtained solutions. An energy hub model that deals with two different attributes (heat and electricity) will also be used for illustration purposes but mainly to elucidate that clustering does not compromise the accuracy of the solutions.

The following are the main contributions of this research:

- Develop a mathematical programming based models for clustering shape-based time-series data (e.g. demand or trajectory data) to be used in reducing the computational complexity of integrated multiscale planning models.
- Study the effect of cluster size and planning model horizon on cluster accuracy using different similarity measures.
- Compare the widely used L_1 (Least Absolute value) and L_2 (Euclidian) similarity measures and discuss the advantages of the L_1 measure for certain planning applications and in maintaining computational tractability.
- Illustrate the reduction in computational complexity on an example from power planning.

- Evaluate the solution accuracy when clustering is applied to multiscale planning models from power industry and energy hubs.
- Develop mathematical programming models for clustering problems with multiple attributes (e.g. simultaneous demand for energy and electricity).
- Illustrate the multiple attribute clustering model on a case study based on an energy hub.

The remainder of this thesis is organized as follows:

Chapter 2: This chapter gives a review of the literature in the area of production planning, integration of scheduling and planning, and clustering methods. The chapter starts by introducing multiscale modeling in general and discusses the multi-scale features including spatial and temporal scales. The chapter then goes to the integration problem of models of different scales to study the intricate details of systems and system components. The supply chain problem is then introduced and its three components: long term (strategic) medium term (tactical) and short term operational), are discussed. A brief survey of deterministic and stochastic planning models is also given. The aim is to illustrate the complexity of such problems when solved even individually. The integration problem that is aimed at integrating the three levels of the supply chain management problem is discussed next. In particular, the chapter focuses on integrating planning and scheduling since few recent articles appeared in the literature. The solution techniques are mentioned and the chapter mentions that almost no solution technique considered a clustering approach to solve the general integration problem. Clustering was used recently to classify demand patterns using heuristics and the scope was limited. The chapter ends with a survey of clustering methods in general and focuses on the heuristic nature of such methods which leads to sub-optimal solutions.

Chapter 3: This chapter starts by defining the clustering problem in general that is aimed at obtaining groups or clusters with high intra-cluster and low inter-cluster similarity. Different similarity measures are then discussed and their advantages and disadvantages are listed. The chapter focuses on the two mainly used measures: (1) L_2 norm (Euclidian, Least square method) and L_1 norm (Lease absolute value method), discusses their use, and explains why L_1 norm can be beneficial to use under certain applications and mainly in order to maintain linearity of the model. The chapter then introduces a Mixed Integer Nonlinear Programming (MINLP) model that is able to provide clusters under the desired similarity measure. The MINLP model is then linearized using appropriate techniques to get an exact linearization model in terms of an MILP. The model is flexible in using any desired similarity measure and simulation runs in the chapter elucidate that the computational effort associated with using the L_1 measure is much less than that of the L_2 measure while at the same time maintaining cluster quality.

Chapter 3 treats also the problem of number of clusters and sequence clustering. The latter has been almost ignored in the open literature but can arise in many practical applications mainly in order to reduce the number of change-overs and set-ups. The presented model is able to come up with both normal clusters and clusters that abide to certain sequencing and take into account the time occurrences of demand patterns. Several simulation runs are presented in the chapter to study the effect of the number of clusters, number of days (or planning horizon), and also the similarity measure used.

Chapter 4: In this chapter a heuristic algorithm is proposed in order to mitigate the computational burden associated with the full MILP model of Chapter 3. The cases studied in Chapter 3 showed some issues with computational effort when dealing with large planning horizons. Chapter 4 presents an iterative scheme that is aible to provide suboptimal solutions to the clustering problem. The heuristic is based on the mathematical programming model of Chapter 3 and starts by an initialization step or generation of initial clusters through a randomization process. The scenarios are generated in Excel by randomizing between maximum and minimum of each hour for the entire demand. The procedure then considers each scenario, fix the clusters and solve the resulting day assignment to get an upper bound on the solution. Next, the heuristic fixes the day assignment and solves for clustered demand in order to get a lower bound solution. Convergence is achieved when the lower and upper bounds are within a certain tolerance. In the case studies considered a tolerance of 10^{-4} was used. The heuristic is based on the mathematical programming approach of Chapter 3 since the upper and lower bounds or day assignment and clustered demands are obtained by solving reduced versions of the MILP original model. The chapter illustrates the proposed heuristic on two case studies. In the first case study, the objective was mainly to study the accuracy of the obtained models and therefore a small horizon case was considered that consisted of twenty days only. The second case study considered a full year of operation. Both case studies illustrated that the proposed heuristic is able to significantly reduce the computational effort in obtaining the clusters and at the same time maintaining solution quality.

Chapter 5: This chapter presents an application of the clustering algorithm developed in Chapter 4 to the unit commitment model and investigates the impact on solution quality. The chapter presents an assessment of the application of clustering and compares the outputs of normal and sequence clustering against the full planning model that does not employ clustering. The Unit Commitment (UC) model was chosen for this study since it is a well-known model that amends itself nicely to the application of clustering of demand curves. The clustered electricity demand from the previous chapter was used as inputs to the UC model. The results show a great computational advantage for the clustered cases compared to the original full scale case and at the same time solution accuracy is maintained. Furthermore, different case studies showed that increasing model size has a minor impact on the solution error. The simulation results suggest also that high solution quality can be achieved with a smaller number of clusters.

Chapter 6: This chapter extends the clustering model of Chapter 3 to take into account multiple attributes. The weighting method is used as a multi-objective optimization approach. Since in general the attributes have different scales or units, which renders the problem a multi-objective optimization problem. Since the resulting model is computationally intensive, the heuristic algorithm developed in Chapter 4 was extended to consider all weight factors' combinations. The algorithm was used on a case study that involves hourly electricity and heat demands for one year. The results show that a better objective function is achieved when the number of clusters increases for both normal and sequence clustering. The normal clustering results are found to be better than the sequence clustering in terms of objective function, error average and standard deviation. It was observed that normal clustering has a major advantage over sequence clustering since many clusters of electricity demand, especially sequence clusters, overlap with each other and cannot be merged. Therefore, for applications that do not require sequencing it is advantageous to use normal clustering to be able to deal with large scale models.

Chapter 7: This chapter presents an application of the clustering model of Chapter 6 to energy hubs. The latter involves multiple attributes that include simultaneously heat and electricity demand. The outputs of normal and sequence clustering are obtained and compared. The chapter also investigates the impact on solution accuracy when multiple clustered demands are applied to the energy hub planning model. It was found that the clustered solutions underestimate the objective function value. Normal clustering is closer to the optimal case than sequence clustering. As the number of clusters is increased the solution quality is enhanced for both normal and sequence clustering. Varying the weight factors does not have a drastic effect on the values of the objective function. However, normal clustering gave better solution qualities than sequence clustering because of the flexibility in terms of sequence restrictions.

Chapter 8: This chapter provides the final conclusions of the thesis that the proposed clustering algorithms possess unique features to yield both normal and sequence clusters using the same construction. The algorithms can be applied to any two dimensional data like the examples used in the case studies (solar radiation, heat and electricity demand). The chapter discusses also future work and extensions to the thesis.

Chapter 2
Literature review

This section reviews the literature regarding multiscale modeling in mathematical programming. First, we consider the work that deals with the integrated scheduling and planning problem since it has been dominating the literature in the last decade as an example of multiscale modeling. After that, we present an overview of clustering algorithms and their applications in multiscale modeling.

2.1 Multiscale Modelling

Multiscale phenomena are part of our daily life. Modern humanity organizes time in terms of days, months, and years. This, as a consequence of multiscale dynamics of the solar system [4]. Multiscale systems featuring a wide range of spatial-temporal scales occur in many scientific domains. Conventional approaches to modelling focus on one scale. If the main focus of interest is a system's macroscale behavior, the microscale is modelled using constitutive relations. Conversely, if the focus is the microscale, it is considered nothing compelling happens at a macroscale, and the process is assumed homogenous at larger scales. For instance, engineers do not need to understand the interactions at atomic level in the materials to build structures. The constitutive relations, which play a key role in modeling, are frequently captured empirically using simple approaches such as linearization, Taylor expansion and symmetry. It is exceptional the success attain by such simple approaches in applied sciences and engineering. For example, the Navier-Stokes equations can accurately describe virtually all the phenomena of simple fluids, which are typically nonlinear. Nonetheless, the extension of such simple empirical approaches to more complex systems is very challenging. Globally, empirical approaches have had limited accomplishments for representing complex or small scale systems where discrete or finite size effects are meaningful. Accordingly, multiscale modeling arises from the need to overcome the limitations of both aforementioned approaches (macro and microscale). This is by simultaneously taking into account models at different scales, and aiming an approach that features the macroscale's models efficiency and microscale's models accuracy. Evaluating a problem simultaneously from different scales and levels is a more comprehensive modeling approach. This represents a basic change in modeling[4].

Although multiscale modeling considers diverse areas and terminology, there exist common challenges to be addressed. Such challenges include validation and design tools for programming and executing simulations. Also, it is worth noticing that multiscale problems do not frequently present a closed solution; except for some ideally assumed cases. Moreover, multiscale systems are characterized by a form of approximation, corresponding to an error below some threshold scale of interest. The distinct terminology used for scale bridging in multiscale systems varies among areas. For instance, terms such as projection, upscaling, model reduction, and physical analogy could be employed to illustrate the practice of reducing the multiscale problem complexity to an insightful, but tractable representation. Accordingly, approximation is implemented to replicate noteworthy quantities at greater length and time scales. This approach widens the scale range at affordable computational time. On the other hand, it is not feasible to approximate everything since information is loss at each step[5]. Multiscale methodologies can be classified into three different categories: descriptive, correlative, and analytical. Each one of these categories are described in detailed next.

2.1.1 Descriptive methodology

The present methodology, as its name reflects, only describes the appearance of numerous structures on diverse scales without analyzing their interrelationships and the mechanisms behind their formation. The majority of the existing literature on multiscale systems are related to this method. This method is employed for stationary as well as dynamic systems. Nonetheless, only dynamic systems experiencing slow changes (e.g., human body, plants) can be described using this method[6]. For example, finite element methods can be applied to model macroscopic processes and obtain the continuum properties on the macroscale (i.e., time and length scales smaller than 1 h and 0.1 m, respectively). Moreover, material parameters must be analyzed based on dominant physical mechanisms and separately modeled. This implies separating the problem into different scales according to their corresponding mechanism. Thus, information from higher scales cannot be influenced by those from lower scales[7].

2.1.2 Correlative methodology

The descriptive methodology's weaknesses encourage the application of alternative methods such as the correlative. The latter includes four different strategies: bottom-up, top-down, con-current, and middle-out. The bottom-up strategy for model construction states that complex systems can be understood at the higher scale by studying the lower scale mechanisms. Many examples are available, Darcy's law on the macroscale could be formulated from Navier-Stokes equations on the mesoscale; which at the same time could be obtain from Boltzmann equation on the microscale. It is valid to suppose that not well-understood mechanisms on lower scales can lead to magnify deviations on higher scales. But, if the lower scale relationships do not significantly influence the higher scales behavior; lower scales deviations can be neglected[6]. This is the case of the gas-solid two-phase flow behavior; where the molecular arrange of the solid is unknown. Nevertheless, presently tools to efficiently construct bottom-up models are scarce. On the other hand, the top-down strategy is based on the principle that large scale models are built and refined by successively adding lower scale ones up to meeting detail and accuracy targets. This method is typically employed in plants' designs where firstly the plant is modeled at large scale; successively the design is changed and refined by modeling each equipment involved on the lower scales. Also, laboratory scales results can be used to improve the modeling approach. Nevertheless, the main limitation of the top-down strategy is the underlying difficulty to keep track of the model's evolution over the lifetime of the process[8]. The remaining two strategies (i.e., con-current and middle-out) application is limited compared with the previously discussed ones. Nonetheless, the four strategies are limited in nature by the fact that they all focus on one-way coupling and the interactions between neighboring scales; whereas two-way coupling is only achievable when incorporating the analytical methodology; which takes into account the implicit interdependence between scales.

2.1.3 Analytical methodology

In multiscale structures, the number of parameters involved is typically higher than that of available equations; which turns out into multiple available solutions for the equation system. In such type of scenario, stability must be accounted typically in the form of an extremum of certain function of the parameters. The latter is what characterizes the analytical method[9]. From a computational perspective, an analogous approach called variational method has been proposed[10]. The method uses variational criterions obtained from studying the dominant mechanisms of the systems, and focus on the overall system's behavior. Furthermore, some authors have proposed that besides structure, stability criterion, and dominant mechanisms; the compromise between sub-mechanisms should be study[11]. This because resolution and compromise are basis for the application of the analytical method. Moreover, there is a lack of study regarding the analytical method. This mainly due to the fact that formulating the stability condition is troublesome. Nevertheless, this method main advantage is the integration of holism and reductionism; which is very promising for predicting criterion.

2.2 Production planning

Planning is the preparation of resources for the future in order to maximize a company's profit. Accordingly, planning involves not only the production processes within the company, but also the exogenous demand supply chains. Effective capacity expansion is key for a company to remain successful throughout time. This because capacity planning involves extensive capital investment. Moreover, capacity expansion planning considers a strategic planning of timing, locations, and sizes of future capacity expansions. These decisions are typically taken based on forecasts of the demands, prices, raw material availabilities, and final products obsolescence. It is important to notice that the success of the strategic decisions rely on the forecasts accuracy and planning techniques effectiveness facilitating the decision making process[12]. There are three main strategies used to plan capacity: lead, average, and lagging. Lead capacity involves expanding the capacity preparing for demand growth. This approach might be the result of a marketing plan to gain new customers by entering a market offering exclusive pricing or products volume. Also, this strategy can assist companies to anticipate rise in demand or high volumes during specific timeframes. On the other hand, average capacity considers expanding capacity to meet the average expected demand; therefore, there will be times the supply will fall short and other when it will exceed the demand. Following this approach, it is expected that there is a 50% chance capacity will surpass the demand and also lag behind. Moreover, the lagging capacity strategy concerns expanding capacity only after a demand surge is recorded. Even though customer service is initially hurt, this approach considers customers will return once capacity is expanded.

Given the time-dependent nature of planning problems a lot of attention has been directed to multiperiod planning in the literature. Planning problems are typically subject to different constraints such as: production capacities including or not expansion, demands satisfaction, availability of resources, inventory requirements, and material balance. The production arrangement may involve the selection of specific units or process types according to changing exogenous conditions commonly unknown[13]. As a result, most planning problems are formulated as mixed integer programming models. This type of formulation enables the selection of discrete variables such as unit selection and capacity expansion size per time period. Furthermore, capacity can be expanded by either installing additional units or increasing the units/process efficiency through innovation. In either case capacity expansion takes much longer than demand to change given the lead time requirements for new units. Inventory is a resource that can act as a buffer between the supply-demand balances; nonetheless, its effect is very limited towards this purpose. Accordingly, capacity expansion is the only feasible resource that can be used to address the supply-demand balance issues.

2.2.1 Deterministic and stochastic approach

The literature shows that most capacity expansion planning problems are addressed mainly from two perspectives: deterministic and stochastic. The deterministic problems considered that the values of certain parameters are known a priori over a given planning horizon; while stochastic ones allow for uncertainty in certain parameters. This type of work started to be addressed since the 1950s; which is the reason many researchers have addressed this topic. The evolution of both formulation approaches from a modeling perspective is presented next following a chronological order.

2.2.1.1 Deterministic approach

One of the first works on deterministic capacity expansion was presented by Wagner and Whitin[14], they developed a forward algorithm to solve the dynamic version of the economic lot size model which allowed for the possibility of demands for a single item, inventory holding charges, and cost variations over different time periods, aiming to minimize inventory costs while meeting known demand in every period. Later, Veinott and Wagner[15] analyzed the capacity scheduling problem for contracting warehouse capacity over *n* time periods at minimum cost. The programming model was also applied to a special case of the transshipment network model solved using linear programming techniques either as an ordinary or a reduced problem. For the reduced problem a rule is specified for the efficient computation of the costs coefficients. This problem was solved using dynamic programming repercussion to fully exploit the model structure. In a continuation of their previous work, Veinott and Wagner[16] propose specific mathematical functions to represent the model's cost coefficients and simplify the solution algorithm; especially for large class problems. The algorithmic simplification demonstrated to be significantly more efficient than the conventional transshipment calculations. Also, the algorithmic simplifications are studied in detail.

Barchi et al.[17] formulated a linear integer programming model for the combined production-inventory and capacity expansion problem. The formulation considers a single product deterministic demand over *n* periods without backordering. A linear transformation is applied to decompose the model into two subproblems (fixed and variable costs) and attain a global optimum. Himmelblau and Bickel[18] developed a procedure to determine the optimal expansion schedule for an existing chemical plant for a set of *n* possible expansions. The aim is to maximize the plant's present worth of net profit subject to meeting an overtime increasing set of demand. A branch and bound algorithm combined with nonlinear programming was also developed and applied. Grossmann and Santibanez[19] illustrate that from mathematical point of view synthesis problems lead to mixed integer nonlinear programmes because discrete and continuous decisions are involved in the design. Accordingly, the authors presented a procedure to effectively tackled process synthesis problems using mixed integer linear programming techniques. The authors concluded that linearity assumptions can be considered reasonable in some cases; otherwise nonlinearities can be handled using discrete variables. Shimizu and Takamatsu[20] present a procedure to expand plant capacities over time using a multiobjective approach. The problem is formulated as a multiobjective mixed integer linear programming model using a stepwise approach to provide the optimal compromise solution for decision making. The approach is based on flexible constraints relaxation and linear programming sensitivity analysis.

Hiller and Shapiro[21] proposed a mixed integer programming approach for optimizing coordinated production and capacity expansion decisions in a company when taking into account production and market learning effects. The model is driven by data since it explicitly includes resource constraints, and can capture discontinuities and economies of scale. The latter two factors are closely related to a company's capital expansion plans. Jimenez and Rudd[22] present a recursive mixed integer programming model for industrial applications. The approach provides the short steps sequence required to achieve a particular long-range development plan. Sahinidis et al.[23] presents a MILP model for the optimal selection and expansion of processes under varying demands and chemical prices forecasts over long term periods. In order to improve the computational time different strategies are investigated including branch and bound, integer cuts, strong cutting planes, benders decomposition and heuristics. In a follow-up work, Sahinidis and Grossmann[24] proposed two reformulations for the same MILP problem that enables much quicker solutions than the original formulation. The reformulations are based on a variable disaggregation technique that exploits the lot sizing substructures. The first reformulation consists of a NLP model that quickly yields good suboptimal solutions; while the second is a MILP formulation for exact solutions leading to up to one order of magnitude quicker computational results in large problems due to tighter linear programming relaxation.

Li and Tirupati[25] present a multiproduct dynamic investment model for making technology choices and expansion decisions over a fixed planning horizon. The aim is to determine the optimal mix of specialized and flexible capacity technologies to take advantage of products swing demands while minimizing total investment cost. A two phased solving approach based on subproblems sequence from the planning horizon is presented as well as some heuristics for scheduling the capacity expansion. Lee et al.[26] proposed an integrated mixed-integer nonlinear programming model including the production and distribution systems. This approach allows savings by trade-offs between the costs of the whole network instead of minimizing each cost separately. The formulation was transformed into convex by replacing variables composing bilinear terms with the exponential transformation; whereas the outer approximation algorithm was used to solve the problem. Hugo and Pistikopoulos[27] proposed a mathematical model that explicitly considers lifecycle assessment criteria into the supply chain networks design and planning.

Accordingly, the resulting formulation is a multiobjective optimization model with quantitative decision-support tools for environmentally conscious strategic investment planning. Bashiri et al.[28] presented a mathematical model considering a multiple-echelon, multiple-commodity production-distribution networks. The model includes various time resolutions depending on the decision type. For instance, high resolution for tactical and low resolution for strategic decisions. Also, the network expansion is planned based on cumulative net incomes; instead of a fixed fund amount or maximum number of facilities. Most recently, Mariel and Minner[29] proposed a mixed integer program for representing multistage processes including multiple products from different sites. In order to capture global market forces the model considers duties and drawbacks from imported materials used for export manufacturing; which allows reducing the network expenditures.

2.2.1.2 Stochastic approach

Regarding stochastic capacity expansion planning problems, one of the first to address such type of problems was Manne[30] who implemented the use of probabilities instead of assuming a fixed growth rate for products demand when planning the excess capacity of a new plant. The random-walk pattern is applied to determine demand. The model considers as key features the economies of scale for the plant construction, and backlogs accumulation penalties. Giglio[31] considered the situation where the product demand and plant's life are both stochastic. Thus, capacity expansion volume and timing must be carefully taken into account in the analysis. The proposed approach can handle stationary and nonstationary demand functions. Also, certain approximations are introduced to enable the use of modified deterministic models to solve stochastic problems. Hazell[32] developed a linear alternative to the quadratic modeling approach that allows maintaining most of the desired features associated with quadratic models. Leondes and Nandi[33] propose an algorithm for solving capacity expansion problems in systems under uncertain demand. The problem was formulated as a two-stage programming model where the independent variable is considered to be a first-stage demand. Chao[34] analyzes the peak load pricing problem simultaneously considering uncertainty in the demand and capacity. Uncertainty in the demand was considered in a more general form compared with previous studies. Likewise, the installed capacity was accounted in the form of random availability; which is a major source of uncertainty adding to power supply shortages.

Sherali et al.[35] considers a capacity expansion problem for the electricity sector taking into account the marginal cost pricing under discretized stochastic demand forecast. The problem was formulated as a two-stage linear programming model with recourse. The aim is to determine a marginal cost pricing strategy for allocating capital costs for optimal capacity planning. Modiano[36] derived certain demand functions for primary resources under economic uncertainty which are data driven. The main decisions concern to whether expand capacity and/or deploy new technologies. Two alternative action models are evaluated: here-and-now and wait-and-see. Dantzig[37] addresses different methods for solving multi-staged systems such as Bender's Decomposition, high speed sampling or Monte Carlo simulation combined with parallel processors can be used to effectively solve planning problems under an uncertainty environment. For instance, parallel processors can help to address the fundamental problems of planning, scheduling, design, and comprehensive systems control. Eppen at al.[38] proposed a multiproduct, multiplant, multiperiod model to address the capacity planning problem under risk. The model considers the inherent trade-offs between risk and profit when addressing capital investment decisions by incorporating elements of scenario planning. Paraskevopoulos et al.[39] incorporate the sensitivity approach to a previously developed nonlinear deterministic model for robust capacity expansion planning. The aim is to evade the complications of nonlinear stochastic models. The robust control diminishes the objective function sensitiveness to departures from zero error compared with the deterministic risky case. The study addresses the effect of demand uncertainty on excess capacity. The main findings are that demand uncertainty and intertemporal rationality potentially result into periods with large excess capacity.

Liu and Sahinidis[40] stated that uncertainty on prices and demands does not significantly affect the quality of deterministic MILP solutions in planning problems as long as plans can be adjusted in terms of production rate and sales/purchases quantity with no economic penalty. Later, Liu and Sahinidis[41] explored the economic implications of uncertainty over plants' operations. In order to address this issue, they developed a two-stage stochastic model for multiprocess planning considering uncertainty. Although the model is analogous to previously develop deterministic models, the proposed model size is restrictive for its wide application. However, the authors developed a Benders-based decomposition algorithm that employs Monte Carlo sampling to determine the expected value of the objective function. This approach significantly helps to reduce the computational time involved in solving large problems since it does not involve solving a large number of optimization subproblems. Maravelias and Grossman[42] proposed a multiperiod large scale MILP model that determines from a pool of options those products that must be tested , the corresponding schedules, network design decisions, and production profiles for the existing and new items. The model takes into account diverse trade-offs with the aim to maximize the expected net present value of multiple projects. Given the size of the formulated problem a Lagrangean decomposition heuristic is proposed to solve the problem more efficiently.

MirHassani et al.[43] studies two complementary modeling approaches and solution methods to address the intractability associated to conventional supply chain planning problems under uncertainty. The first method considers wait-and-see models; while the second proposes a two-stage integer stochastic modeling approach for the formulation and subsequent solution to the problem. Results from the first method are used to solve the latter approach. Chen and Lee[44] proposed a MINLP model for a multi-echelon supply chain network. The formulation incorporates two types of uncertainties: market demands and product prices. They employ a scenario-based analysis to model the market demand while fuzzy variables are used to denote product prices. The fuzzy variables are used to denote the sellers' and buyers' incompatible preference over product prices. Santoso et al.[45] introduce a stochastic mathematical model and corresponding solution algorithm to address large scale supply chain design problems. The solution approach includes the sample average approximation method combined with an accelerated Benders decomposition algorithm to obtain high quality solutions at relatively short computing times. Bidhandi and Yusuff [46] propose a modified solution method for solving supply chain design problems under an uncertain environment. The improved solution method integrates sample average approximation with accelerated Bender's decomposition to improve the solution computational time. Additionally, the surrogate constraints method is used to accelerate the decomposition algorithm. Baghalian at al.[47] developed a stochastic multiproduct supply chain network that considers both demand and supply side uncertainties at the same time. A path-based formulation is employed to represent the supply-side uncertainties such as possible disruptions in manufacturers, distribution centers and connecting routes. The model also takes into account reliability cut-set and robust optimization concepts. In order to solve the model to global optimality, the piecewise linearization method was applied to the formulation. Hatefi and Jolai[48] propose a robust and reliable mathematical model for the design of an integrated forward-reverse logistics supply chain. The model takes into account parameter uncertainties and facility disruptions at the same time. Additionally, augmented *p*-robust constraints are applied to control the network's reliability for the disruption scenarios. Chen at al.[49] developed a behavioral supply chain model using cellular automata, which includes strategic interactions between neighboring facilities and its corresponding effects on the whole supply chain, to analyze the effectiveness of typical recovery strategies employed to treat unpredictable disasters disrupting supply chains.

The previously described works show how deterministic and stochastic planning problems have evolved over the last 60 years, from small scale problems with simple solution algorithms to very complex large scale mathematical models requiring the combination of special software and solution techniques to be properly solved at affordable computational times.

2.3 Integrated scheduling and planning

A supply chain is an integrated business system where various business entities work together in an effort to: acquire raw materials, convert raw materials into final products, and deliver the products to retailers or directly to customers. The planning and scheduling of activities in the supply chain are very complex and may take place inside the enterprise or across the whole supply chain. The objective is to obtain products with high quality at lower costs, lower inventory levels, and high performance levels[50]. For instance, it has been found that inventory levels can be significantly reduced leading to significant savings without upsetting customer's satisfaction levels. This by raising the efficiency levels across the entire supply chain through proper coordination of material, information, and financial flows[51]. Accordingly, the planning problem cover both a wide range of activities as well as time scales.

Long term or strategic planning is used to determine the supply chain structure such as facilities capacities and locations; while medium term or tactical planning involves decisions like assignment of facilities' production targets as well as transportation between the supply chain business entities. On the other hand, short term planning is done on a daily or weekly basis in order to perform the assignment and sequence of tasks in the different units. Short term planning at the production level is known as scheduling. However, given the relationships between the different supply chain levels, there are diverse trade-offs between decisions taking place at each level. If one aims to achieve a solution with global optimum, the interdependence between the different levels should be considered, and the planning decisions made simultaneously. Accordingly, the planning problem must be integrated in nature[3].

The planning and scheduling problem regularly arises in the manufacturing and supply chain context where tasks must be assigned to facilities and scheduled on each facility. However, each task is subject to deadlines, while precedence and other side constraints might be required. Even though these type of problems are common, it has been proven that they are difficult to solve. Manufacturing systems include diverse issues that are intrinsically related to each other. This is the reason why they must be approached altogether and simultaneously solved. Traditional engineering design methods commonly known as serial engineering are applied in a sequential manner. For example, process units are chosen mainly based on unit operating costs. Although this approach may look adequate at first based on economics and may translate into simpler scheduling routines, it could originate overload troubles and create bottlenecks. Accordingly, this traditional approach can hamper the manufacturing units' capabilities[52]. In practice, scheduling and planning problems hardly include a unique consideration[53]. Conversely, these problems include multiple objectives requiring to be addressed simultaneously. Thus, the proper optimization approach must involve determining the process plan and schedule at the same time. Integrated scheduling and planning problems have received a lot of attention in process systems engineering and operation research. These types of problems can be categorized in terms of the model formulation and corresponding solution methods [3].

2.3.1 Model formulation

All models in the literature for integrated scheduling and planning problems fall into one of four categories; detailed scheduling models, relaxed and aggregated scheduling formulations, off-line surrogate models and hybrid modeling for rolling horizon approaches. A brief description of each one is presented below.

There are two approaches used to tackle integrated scheduling and planning problems with detailed scheduling: first, to include a scheduling sub-model into the planning model linked via constraints (as in Figure 2-1); second, to consider the scheduling model for the entire horizon. In this way, the planning decision variables are replaced by those of scheduling. Though both formulations ensure optimality, they usually lead to large scale models which are computationally expensive even with current computing resources.

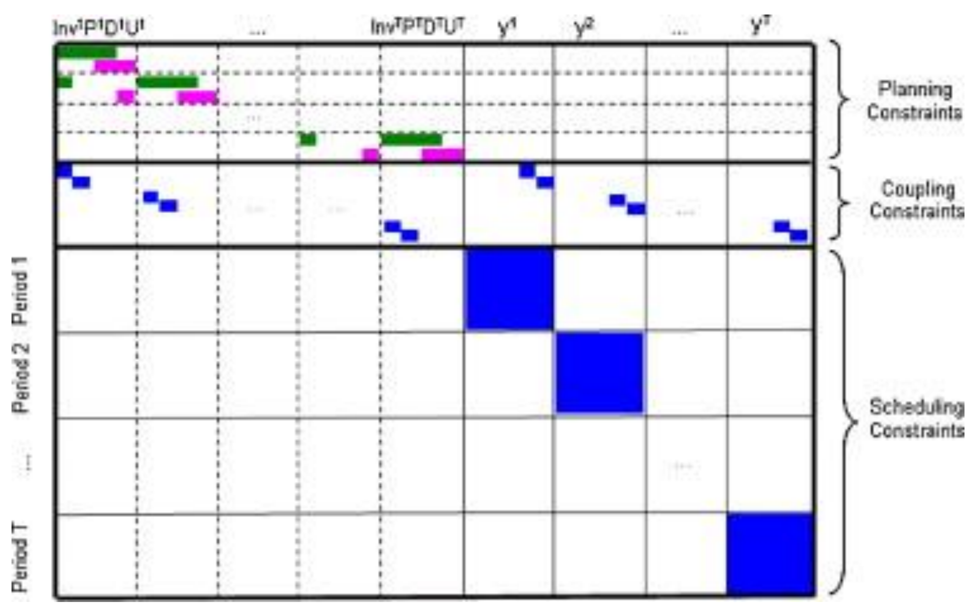

Figure 2-1: Structure of integrated scheduling and planning[54]

Relaxed and aggregated scheduling formulations offer a better alternative for dealing with computational complexity. As the term suggests, the solution optimality will be compromised by the degree of aggregation. Of the several ways to relax and aggregate scheduling models, one is to keep job assignment constraints and variables and to remove those of job sequence.

Rolling horizon approaches have been utilized initially for medium term scheduling[55]. The time frame is divided into serval subintervals. At every iteration, the early interval will be represented in detail while the rest will be aggregated. After that, the early interval is fixed, and the next interval will be in detail until all intervals are considered (see Figure 2-2). This approach is based on the fact that aggregate solution in the later period has little effect on the solution quality as a whole. It was also adopted to address integrated scheduling and planning[56,57]. However, the solution depends on the aggregate representation.

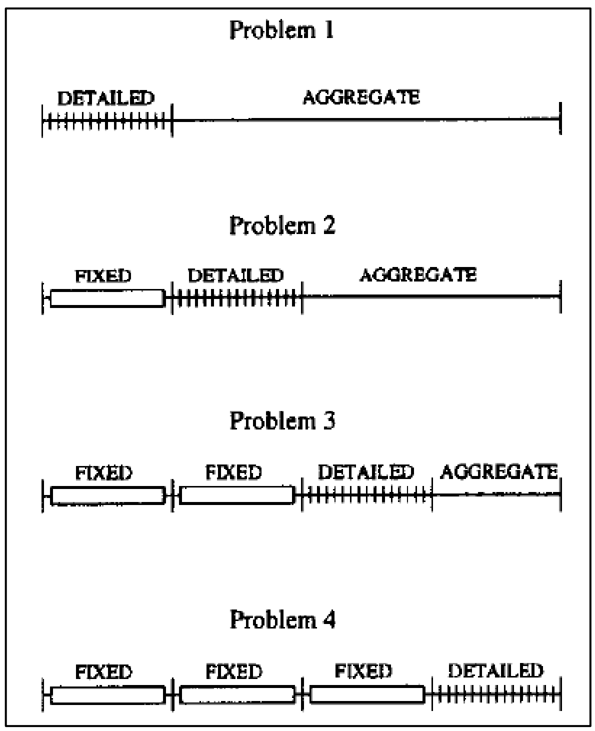

Figure 2-2: Forward rolling horizon algorithm [55]

Another way to address integrated scheduling and planning is to use off-line surrogate models, whereby calculations are carried out to simulate the scheduling. These calculations will provide an accurate description of resource constraints and production costs which will be used to generate constraints defining the feasible region for the problem [58]. The advantage of this technique is that the calculations need to be conducted only once. However, the constraint generation is a resource intensive process which might lead to a large size model.

2.3.2 Solution methods

Similar to model formulation, solution methods have been developed to address integrated scheduling and planning problems. Figure 2-3 shows the three classical methods: hierarchical, iterative, and full space. The problem is decomposed into a master problem and subproblem. The master problem deals with high decision levels like production targets while the subproblem aims to find a complete schedule. In the hierarchical approach, one-way information flows from the master problem to the subproblem. If there is a closed loop between them, it will be iterative. The full space incorporates detailed scheduling in the formulation and does not require for information exchange. All the three approaches are discussed below.

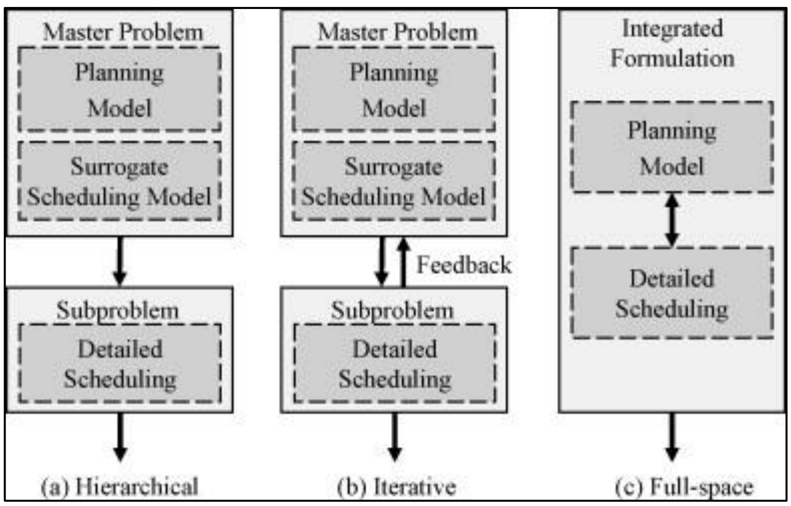

Figure 2-3: Solution methods for integrated scheduling and planning[3]

In the hierarchical approach, the master problem is solved first and then the production targets are fed to the detailed scheduling model to obtain the optimal schedule. If infeasibility is encountered in the subproblem, heuristics will be used to find a solution in the neighborhood of production targets found earlier. The most common application of the hierarchical approach is to couple it with the rolling horizon approach[56].

Most of the time infeasible schedules are obtained when aggregate representation of the resources is used. One way to take advantage of this fact is by generating cuts (i.e. valid inequalities) to avoid such a solution in the master problem[59–61]. This is the most common way to use the iterative approach. However, most of these cuts are problem specific.

The full space approach results most of the time in large scale models. In term of implementation, the obvious way is to use standard mathematical programming[62,63]. Another way is to use meta-heuristics such as simulated annealing or genetic algorithms. The most common approach, the decomposition technique, requires a careful study of model structure. Bender's decomposition is preferred when complicating variables are encountered whereas the Lagrangian relaxation/decomposition is the choice for complicating constraints[2].

All aforementioned model formulations and solution methods are modeled as MILP (mixed integer linear programming). Most of them are either problem specific or computationally intractable when a longer horizon is considered [3]. Integrated scheduling and planning has been modeled as hybrid CP/MILP[64–66]. CP (Constraint programming) has superior performance when it comes to scheduling whereas MILP is the preferred choice for planning. Combining CP and MILP requires special solution techniques but it is accessible using ILOG. The results show the hybrid CP/MILP outperforms MILP for integrated scheduling and planning problem. However, hybrid CP/MILP still faces the same computational complexity when a longer horizon is considered.

2.4 Clustering and its applications

Clustering aims to determine structure in unlabeled data sets by means of organizing the data into homogenous groups where the within-group-object similarity is minimized while the between-group-object dissimilarity is maximized. This method is required in cases when unlabeled data are available and need to be organized regardless the data type: binary, categorical, numerical, interval, ordinal, relational, textual, spatial, temporal, spatial-temporal, image, multimedia, or mixtures of the previous data types[67]. Cluster analysis groups objects (observations, events, etc.) using information found in the data defining the objects or their relationships. The aim is that objects in a given group must be similar or related to one another and dissimilar or unrelated to objects in other groups. The higher the similarity or homogeneity within a given group and the higher the dissimilarity between groups, the more effective the clustering. The objects (measurements, events) are typically denoted as points or vectors in a multidimensional space; where each dimension denotes an especial attribute (variable, measurement) describing the object. To simplify things, it is commonly regarded that all attributes feature values. As a result, a set composed of objects is denoted at least conceptually as an m by n matrix; which corresponds to a matrix composed of m rows (one for each object) and n columns (one for each attribute). The data might be transformed before used since diverse attributes can be measured on different scales. It is a common practice to standardize the data in a way that all attributes are expressed in the same scale. When the values range significantly diverge from one attribute to another, the diverging attributes scales can dominate the clustering analysis results.

Although the concept of clustering is yet not well defined, most cluster analysis are carried out to obtain a classification of the data into non-overlapping groups. Nonetheless, fuzzy clustering is a particular case which allows an object to partially be part of several groups[68]. Cluster analysis can be a very helpful tool in many areas to either directly or assist finding objects classifications. However, this tool is commonly only part of the solution to larger problems involving more steps and techniques. The great majority of clustering analyses have been carried out on static data; which comprises data with unchanged or negligible changing values. Clustering has been used to find genes and proteins with similar functionality, and identify geographical regions prone to earthquake occurrences. On the other hand, this method can also be used as a helpful starting point for other applications such as: data understanding or finding the closest neighboring points. Accordingly, whether directly used for applications or as a supporting tool, clustering has been traditionally used in a wide variety of fields like phycology, biology, statistics, pattern recognition, information retrieval, machine learning, and data mining.

Clustering has been extensively studied across different disciplines for more than 50 years [69]. All clustering algorithms can be categorized into two groups: hierarchical and partitional. In hierarchical clustering, the data subject to classification are not partitioned into a specific number of clusters or classes at a single step. Conversely, the classification is the product of a series of partitions that could be originated from a single cluster enclosing all objects to n clusters each enclosing a single object. Hierarchical clustering may be subdivided into two methods: agglomerative and divisive (top-down). In the agglomerative method, each data point starts in its own cluster and then merges with the most similar cluster recursively until the termination criterion is met. This is probably the most widely employed hierarchical method. This method produces a succession of partitions of the data: the first involves n single member clusters while the last involves a single group enclosing all n subjects. Agglomerative procedures typically fuse subjects or groups of subjects that are the closest or most analogous at each stage. The dissimilarities amongst procedures emerge due to the diverse manners in which the distance or similarity between a subject and a group enclosing several subjects, or between two groups of subjects[70].

On the other hand, the divisive method starts with all data in one cluster and then divides it into smaller clusters. It operates in the opposite direction to agglomerative methods by starting with an extensive cluster and sequentially dividing clusters. Divisive procedures can be computationally demanding if all

$2^{k-1} - 1$ probable divisions into two subclusters of a cluster of *k* objects are accounted at each stage. Nonetheless, there are relatively simple and computationally efficient methods commonly known as monothetic (single variable used to divide a given stage) for data involving *p* binary variables. The latter typically divides clusters depending on the presence or absence of each of the *p* variables. Accordingly, at each stage clusters including subjects with specific attributes are either all present or absent. The data requires to be in the form of a binary matrix. Although divisive methods are less common than agglomerative ones, the latter have the advantage that the majority of users are interested in the main structure in their data; which is given from the outset of a divisive method[71]. The data assignment is irrevocable for both methods, meaning that once the data point is assigned to join a new cluster, it cannot go back to the former cluster.

On the other hand, partitioning clustering algorithms start by dividing the data into *k* partitions, so the number of clusters must be defined *a priori*, which is not required for hierarchical clustering. After this, the partitioning clustering will iteratively allocate the data points to the clusters based on a similarity measure until a termination criterion is met. Partitioning clustering is widely preferred in pattern recognition and other disciplines especially *k*-means clustering[69]. However, these algorithms, like *k*-means, are very sensitive to the initialization step and can lead only to local minima. Therefore, they are usually carried out with different initial steps, with the best run corresponding to the lowest error. Despite such a procedure, there is no guarantee of reaching global optimality.

Mathematical programming plays a role in developing a clustering algorithm. Rao[72] presented two integer programming formulations with different distance functions. The first formulation, whose objective is to minimize the sums of squares within groups, can lead to a linear integer model under certain conditions. The other formulation's objective is to minimize the maximum distance within group but this results in a non-linear integer model. Similarly, Kusiak[73] illustrated 5 different integer programming formulations with heuristics to address clustering problem. Koontz et al.[74] developed a clustering algorithm using branch and bound. In addition, Sağlam et al.[75] formulated a MILP model for customer segmentation of digital platform company. They also developed an improved algorithm to overcome computational complexity without compromising optimality.

Clustering has also been applied in power generation applications. Balachandra and Chandru[76] grouped electricity demand for an entire year into 9 clusters in sequence order using discriminant analysis. The clusters were then used as an input for a resource constraint linear programming model for electricity system based on supply demand matching[77]. Similarly, Fazlollahi et al.[78] developed a model to cluster electricity demand using *k*-means. They extended the model to include other attributes like heat demand, electricity price and solar radiation. The clusters were used as input for operation optimization of fixed configuration energy system to supply the demand of an urban district. The solution was compared to a reference but the study didn't mention solution quality of the reference nor how it was solved.

The different clustering methods mentioned above have been used to reduce model size for tackling integrated scheduling and planning problems. This strategy is promising based on the concept of cyclic scheduling. Used extensively in process systems engineering[79–82], cyclic scheduling requires certain demands to be processed over certain time periods repeatedly within the time horizon. Similarly, clustered demands will be processed repeatedly in the same manner. Even though the solution of cyclic scheduling is not optimal in principle, it is admissible since it avoids the expense of computational complexity.

As it is apparent from the above discussion, different researchers have used different approaches in term of clustering. The resultant clusters of discriminant analysis were in sequence order while those of k-means were not[76 78]. Therefore, there is no indication of which algorithm will result in better clusters. Moreover, we don't know if the sequence will have an influence on solution quality. Thus, it is one of the objectives in this thesis to investigate these matters and to provide a detailed approach to the appropriate use of the different types of clustering for multiscale mathematical programming models.

2.4.1 Weighting variables

Weighting means to assign lesser or greater importance to a variable over other variables in determining the proximity between two objects. The weights selected for the variables show the importance the researcher assigns to each one for the classification task. The assignment could come from the researcher judgment or some specific feature from the data matrix. For instance, in cases when the researcher assigns the weights the methods proposed by Sokal and Rohlf[83] and Gordon[84] showcase good examples of indirect weighting assignments. Accordingly, the authors observed differences among the chosen objects. These differences were later modeled employing the corresponding variables and weights denoting their relative importance. Thus, weights best fitting the perceived differences are accordingly chosen. Moreover, a typical way of determining the weights from a data matrix (X) consists of defining the weights w_k of the k^{th} variable to be inversely proportional to its variability measure. The former weighting approach means that the relevance of a given variable is reduced as its variability increment. However, Fleiss and Zubin[85] stated that one of the weaknesses of this approach is that it might mitigate dissimilarities between groups on the most suitable discriminator variables. A variable variability includes variation within and between groups within the set of objects.

Diverse variability measures can be employed to define weights. For instance, for continuous variables the most common weighting approach is either its standard deviation reciprocal or its range reciprocal. If the groups are known a priori employing the within group standard deviation to denote weights would greatly help to overcome the discriminator variables problems[86]. Also, Mahalanobi's[87] generalized distance could be employed to define the distance between pair of objects using the pooled within group covariance matrix. An extension of the previous work was done by Art et al.[88], who employed an iterative algorithm to determine observation pairs prone to be within the same cluster while using them to estimate a within cluster covariance matrix. On the other hand, De Soete[89] developed an alternative criterion for determining a variable importance from the data set by finding one weight per variable that yield weighted Euclidean distances which minimizes a criterion for departure from ultrametricity. The ultrametric property is associated to several clustering methods features, especially the ability to represent the hierarch by a dendrogram.

Another method for constructing weights from the data matrix is variable selection. In this method an empirical selection procedure can be used to identify a subset of the initial variables for the cluster analysis. The method consists of assigning the value of one to selected variables while omitted variables are assigned the value zero[90,91]. The method takes place in an iterative way to identify variables that when contributing to a cluster algorithm lead to internally cohesive and externally isolated clusters. Conversely, when clustered singly generates a solution in reasonable agreement with other variables subsets. Moreover, it is important to point out that the selection of variables by itself for the cluster analysis represents a form of weighting given that the omitted variables are in practiced assigned the value of zero. Likewise, the typical standardization of variables to unit variance is also a particular case of variables weighting. In general, weights based on subjective judgments reflect an actual data classification; which at some degree is the opposite of what cluster analysis aims for: find previously unnoticed groups. Also, an optimal criterion for determining importance weights empirically has not been found yet. This since the clustering performance of distance measures based on weights seems to be a function of the cluster structure. Nevertheless, weights obtained from measuring non-importance by estimated within group variability seem to feature the greatest potential for recovering groups in successive cluster analysis[86].

2.5 Conclusion

As discussed in the previous sections, researchers have been working extensively in the last decade to tackle the problem of multiscale modeling in mathematical programming, such as integrated scheduling and planning. Although there has been a huge advancement in this field, the computational burden still exists when a longer horizon is considered. The application of clustering methodologies to this problem show a lot of promise and ample opportunities since it reduces the model size considerably. Unfortunately, only little work has been done on trajectory clustering of time series data in more than one dimension and in particular no solid modeling framework has been undertaken. Furthermore, the consideration of time of events and maintaining the sequence of the demand curves has mostly been ignored. Therefore, the objective of this thesis is to propose mathematical programming based algorithms to the trajectory clustering problem, investigate the solution accuracy when clustering is applied and establish a procedure for others to follow.

In particular, since most clustering algorithms that can be found in the literature are heuristic based, each one will result in different clusters since they are tailored to suit certain objectives[69]. As a result, it has been a practice for researchers to develop their own clustering algorithm. Our target is to develop comprehensive mathematical programming models for clustering since such an approach has been shown to deliver superior performance and can be easily customized without any major change[75]. Once the clustering model is formulated, it can be easily modified to have a sequence order feature like that developed by Balachandra and Chandru[76] and Marton et al.[92] to examine the impact of sequence in applications. Eventually, this will lead to two clustering models. Furthermore, since the proposed models will be mathematical programming based they can be easily incorporated within multi-scale planning models and this will ease the integration of the clustering and planning steps.

The models will be illustrated on a well-known scheduling model that has strong credibility in the literature. The unit commitment model is the best candidate since it has been rigorously developed and has real case applications. Unit commitment is a scheduling model for power generators[93]. The objective is to optimize generating resources to supply system load while satisfying prevailing constraints, such as minimum on/off time, ramping up/down, minimum/maximum generating capacity, and fuel and emission limit. Moreover, parameter data, like electricity demand, is easily accessible nowadays. In addition, the impact of cluster sequence order will be more evident in the unit commitment model. The unit commitment model will be solved for both clustered and regular demand of the same period. This will help in determining the accuracy when clustered demand is applied. Moreover, this will illustrate if the sequence order has an effect on the solution.

The clustering mathematical programming model is also extended to include multiple attributes, like simultaneously clustering electricity and heat demand data. This will lead to a clustering model of the multi-objective type. There are several ways to address these kind of optimization problems. The weighting method and ε-constraints method will be employed in this research.

Finally, the multi-attribute clustering model will be illustrated on an energy hub model that will be solved for both regular and clustered demand in the same period to be able to compare the computational enhancement and specially assess the accuracy of employing the clustering model.

Chapter 3
Shape-based Time Series Mathematical Programming Clustering Model

3.1 Introduction

The objective of this chapter is to develop a clustering algorithm based on a mathematical programming formulation and which can be imbedded in multiscale mathematical programming models. The clustering algorithm will be formulated as an integer programming model. Integer programming is a very flexible construction when it comes to modifications of objective function or selection process compared to the traditional heuristic approaches.

The clustering algorithm that will be introduced here belongs to the general class of time-series data clustering which as indicated in **Chapter 2** has received renewed interest recently because of the many potential applications and diverse problem domains due mainly to the unprecedented availability of large amounts of data (big data). For example in the case of smart grid applications, data is characterized by its large volume over extended time intervals and is known to be difficult to mine especially due to the additional dimensionality of shape characteristics of the demand curves. It is the aim of this chapter, therefore, to develop an algorithm that can cluster or segment such demand data taking into account not only the value of the demand itself but also temporal patterns. Furthermore, not only similarity in shape will be considered but the time at which the trajectories were created will also be regarded as important. In this way, the clustered time series data can serve in reducing the computational complexity of multi-scale models. An additional utility can be in learning the behavior of different customers or consumers be it individual consumers, individual units, or factories as a whole.

Clustering of time-series data and in particular shape or trajectory clustering has the goal of producing classes of trajectories with high intra-cluster similarity and low inter-cluster similarity. This means that the aim is to produce classes of curves or groups that are dissimilar or have a low similarity (i.e. distance) whereas the entities in each group have enough similarity to each other. Therefore for trajectory clustering similarity in both shape and time must be considered. This is an additional dimension in clustering that this chapter aims to focus on.

It is clear from the above discussion that the concept of similarity is central to any clustering algorithm. The similarity distance measures that are widely used by the majority of time series clustering methods are based on two methods[94–98] : L_2-norm (also known as least square method, LSM) and L_1-norm (known as the least absolute value method, LAVM). The least square procedure or L_2-based method is generally used in estimating regression coefficients. It is analytical tractability is well known in regression applications and the parameter estimates obtained by this methods are known to be best linear unbiased estimates (BLUE). For linear regression applications, the solution obtained is also unique. The method is however based on the assumption that the disturbance terms follow a normal distribution. There are many cases in practice where these terms do not follow a normal distribution and many demand curves are among these cases. The presence of large disturbances (or outliers) cause the assumption of normal error distribution to be violated. The methods that have been used to remove outliers have been known to produce different results either by removing a consistent measurement (sinking effect) or by failing to remove an actual outlier (hiding effect)[94].

The L_1 or LAVM criterion has been known to be less sensitive to the presence of outliers and has been known to be a robust alternative to the L_2 criteria for problems exhibiting large disturbances or incomplete data[95]. The LAVM criterion has also being known to have an important advantage that the parameter estimation problem can be formulated as a linear optimization problem. The solution for such problems is not always unique. In the case of a unique solution the unknown cluster is obtained around the estimated median as compared to the central value for the L_2 criterion. Other facts about the L_1 criterion as mentioned in the literatures are[99]: (1) For high-kurtosis disturbances, the L_1 criterion gives rise to significantly smaller standard errors than the L_2 criterion, (2) The estimators obtained from the L_1 criterion were almost normally distributed in the presence of high-kurtosis disturbances, (3) The residual errors are approximately normally distributed, (4) L_1 estimates can be obtained through linear programming, and (5) The L_1 estimates are preferred to the L_2 estimates for cases when the median is superior to the mean as an estimator of location (e.g. Cauchy, Laplace, and Logistic distributions) in the sense of having smaller asymptotic confidence ellipsoids for the regression coefficients for a given sample size.

In the developments that will follow in this chapter even though we opt to employ the L_1 criterion as a measure of performance, some L_2 results are also included for comparison purposes and also to illustrate that the modeling framework and algorithmic approach developed is general in the sense that different metrics can be easily employed. As mentioned in **Chapter 2**, generally speaking and apart from the similarity measure employed, clustering algorithms can be divided into two main groups[100–102]: partitional and hierarchical. The former aims at grouping similar objects in the same cluster and uses an iterative scheme to refine the grouping in order to minimize an objective error function. The *k*-means algorithm is the most popular algorithm in this group. It employs the L_2 criterion and starts initially with k random centers, then assigns elements to the most similar cluster, recalculates again the centers through averaging of all assigned elements for each cluster, and the procedure is repeated until the L_2 error function is minimized. The k-means algorithm for time-series data is known to have a complexity of the order $O(k.N.r.D)$ where k is the number of desired clusters, N is the numbers of objects or elements to be clustered, r is the required number of iterations until convergence, and D is the dimension of the time series data (e.g. 2 for profiles). The algorithm requires a good initial set of clusters and in general converges to a local minimum.

Hierarchical algorithms on the other hand find nested clusters in either an agglomerated or divisive manner. In the agglomerated mode, each element is assigned to its own cluster and the most similar clusters are merged successively to form a hierarchy of clusters. The divisive mode is the opposite of the agglomerative mode in the sense that the whole set of elements is assigned to one cluster only initially, and is then recursively divided into smaller clusters (see for example the heuristic of Marton et al.[92])

In the present chapter we use a different approach that is based on mathematical programing formulation. The advantage of such approach is that it can be used conveniently to cluster time-series data in higher dimensions and can find geometric shape clusters employing the techniques of mathematical programming. We opt to employ the L_1 criterion as a measure of similarity for the many reasons discussed above and mainly to maintain linearity of the model. Nevertheless the results of the L_2 criterion are also included for comparison purposes.

3.2 Model formulation

Consider a set of load curves for D days and H hours to be grouped in C clusters. The goal is to assign days to clusters with minimum dissimilarity between them. Equations 4-1 to 4-3 represent a mathematical model for clustering to minimize the integral absolute error (IAE) or L_1 norm (Least Absolute Value)

$$\min z = \frac{\Delta}{2} * \sum_{d=1}^{D} \sum_{h=1}^{H-1} I_{d,h} + I_{d,h+1} \qquad \textbf{(3-1)}$$

$$\text{s.t.} \quad \sum_{c=1}^{C} x_{d,c} = 1 \qquad \forall \ d \qquad \textbf{(3-2)}$$

$$I_{d,h} \geq \left| L_{d,h} - Cl_{c,h} \right| * x_{d,c} \qquad \forall \ h, d, c \qquad \textbf{(3-3)}$$

where: $I_{d,h}$ is the absolute difference between the actual demand curve $L_{d,h}$ and the clustered representative curve $Cl_{c,h}$ for hour h in day d, $x_{d,c}$ is a binary variable that indicates the assignment of load for day *d* joining cluster *c*, and it is equal to one if such assignment takes place and equal to 0 otherwise.

Eq. 3-1 is a numerical evaluation of the norm L_1 using the trapezoidal rule for *IAE* between loads *L* and cluster curves *C*. Other quadrature integration schemes could also be employed by accounting for the number of segments in an appropriate manner. For instance, for an odd number of segments, one can apply the trapezoidal rule for the first segment and Simpson's 1/3 rule for the remaining segments. However, we chose to employ the trapezoidal rule for illustration purposes and for the sake of simplicity. The intervals considered are small enough (in hours) compared to the overall horizon (year) and the approximation of *IAE* using the trapezoidal rule will be adequate. Eq.3-2 is a day assignment constraint that requires that each day of the year is assigned to a cluster *C* of curves. Eq.3-3 evaluates the absolute difference between the load and cluster curves to be used in the performance criterion, Eq. 3-1.

The model is a mathematical representation of clustering trajectories of time series data and is aimed to achieve clusters through the minimization of the L_1 norm. Utilizing the L_2 norm is also straightforward and requires the use of Euclidean distance in Eq. 3-1. The L_1 norm has been extensively used as a performance criterion in process control applications[103]. Figure 3-1 shows the graphical representation of *IAE* for two curves *L* and *C*. Eq.3-4 presents the mathematical equation for *IAE*[103].

$$IAE = \int_a^b |L(t) - C(t)| dt \qquad \textbf{(3-4)}$$

In addition, it is possible to cluster demand in sequence by adding the following set of constraints that uses the concept of string property[104]. Sequence clustering can be of importance in order to maintain flexibility of operations. In many instances, continuous similar operations are desired in order to minimize the inconvenience and cost of change-overs and set-ups. In order to incorporate the time dimension into the clusters and require sequencing to be conformed, the following sets of constraints are added:

$$x_{d+1,1} \leq x_{d,1} \qquad \forall\, d < D \qquad \textbf{(3-5)}$$

$$x_{d+1,c} \leq x_{d,c} + x_{d,c-1} \qquad \forall\, d < D, c > 1 \qquad \textbf{(3-6)}$$

$$x_{D,c} \leq x_{D-1,c} + x_{D-1,c-1} \qquad \forall\, c > 1 \qquad \textbf{(3-7)}$$

Eq.3-5 to 3-7 take care of sequence for first, intermediate and last clusters respectively. The following constraint (Eq. 3-8) is equivalent to Eq.3-5 to 3-7 assuming that terms that do not exist are dropped out. Several algebraic modeling languages such as GAMS have this feature built in.

$$x_{d+1,c} \leq x_{d,c} + x_{d,c-1} \qquad \forall\, d, c \qquad \textbf{(3-8)}$$

The formulated model can be used for normal clustering (Eq. 3-1 to 3-3) or sequence clustering (Eq. 3-1 to 3-3 and Eq.3-8). This formulation provides a unique platform to provide normal and sequence clustering since it utilizes the same algorithmic structure. However, both formulations are MINLP because of the usage of the absolute value and multiplication between $Cl_{c,h}$ and $x_{d,c}$ in Eq.3-3. The following section will illustrate the model linearization.

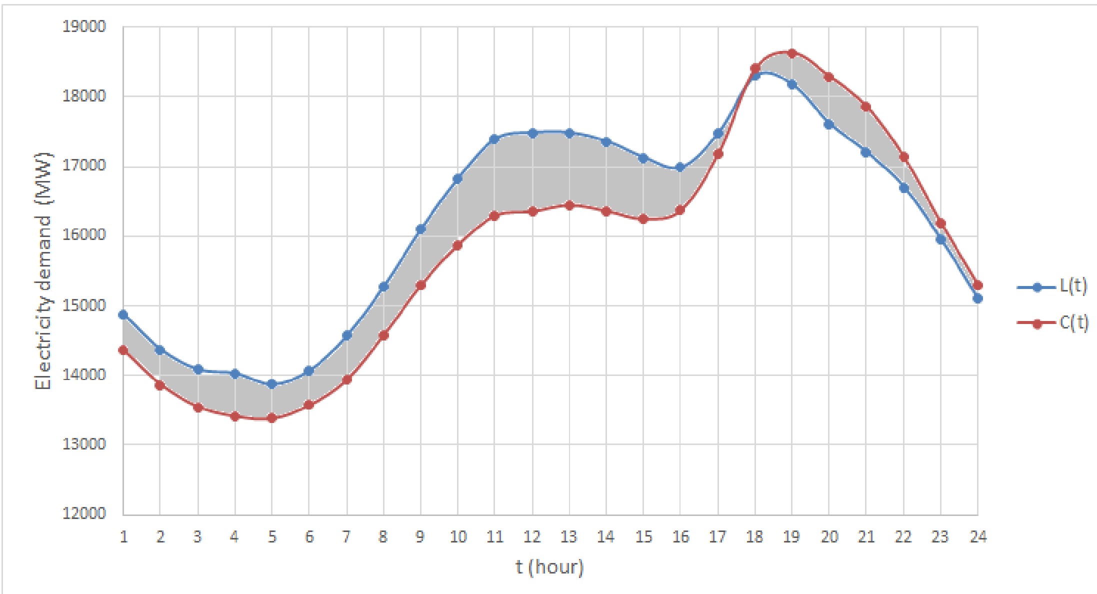

Figure 3-1: Graphical representation of IAE (the grey area) for two curves L(t) and C(t)

3.3 Model linearization

The absolute function in Eq.3-3 can be linearized using the following equations [105].

$$I_{d,h} \geq L_{d,h} * x_{d,c} - Cl_{c,h} * x_{d,c} \qquad \forall\, h, d, c \qquad \textbf{(3-9)}$$

$$I_{d,h} \geq Cl_{c,h} * x_{d,c} - L_{d,h} * x_{d,c} \qquad \forall\, h, d, c \qquad \textbf{(3-10)}$$

When the load curve is selected ($x_{d,c} = 1$), one of the constraints will take on a negative value while the other will be positive. Therefore, the constraint that has a negative right hand side will be redundant to the other, and $I_{d,h}$ will be equal to the positive difference.

Although this scheme eliminates the absolute value in the model, the multiplication between $Cl_{c,h}$ and $x_{d,c}$ still exists in Eq.9-10, which makes the model bilinear. This can be linearized further by introducing a new continuous variable $P_{h,d,c} = Cl_{c,h} * x_{d,c}$ and applying the following constraints[106,107]:

$$I_{d,h} \geq L_{d,h} * x_{d,c} - P_{h,d,c} \qquad \forall\, h, d, c \qquad \textbf{(3-11)}$$

$$I_{d,h} \geq P_{h,d,c} - L_{d,h} * x_{d,c} \qquad \forall\, h, d, c \qquad \textbf{(3-12)}$$

$$Cl_{c,h} - L_h^U * (1 - x_{d,c}) \leq P_{h,d,c} \qquad \forall\, h, d, c \qquad \textbf{(3-13)}$$

$$L_h^L * x_{d,c} \leq P_{h,d,c} \qquad \forall\, h, d, c \qquad \textbf{(3-14)}$$

$$Cl_{c,h} - L_h^L * (1 - x_{d,c}) \geq P_{h,d,c} \qquad \forall\, h, d, c \qquad \textbf{(3-15)}$$

$$L_h^U * x_{d,c} \geq P_{h,d,c} \qquad \forall\, h, d, c \qquad \textbf{(3-16)}$$

Applying the aforementioned linearization schemes renders the model to be an MILP and therefore more tractable. To conclude the modeling section, we indicate that the model for normal clustering consists of Eq.3-1 to 3-2 and Eq.3-11 to 3-16 while for sequence clustering, the model consists of Eq.3-1 to 3-2, 3-8 and Eq.3-11 to 3-16.

3.4 Case studies

The aim of the following case studies is to assess through simulation the computational performance and outputs of the clustering model derived in the previous section. Case Study 1 will use electricity demand while Case Study 2 will use solar radiation to illustrate the wide applicability of this clustering model. In Case Study 1, we also present results based on the L_2 criterion and compare them to the results of the L_1 performance measure. Although the discussions presented in section 4.1 shed light on the usages of the two performance criterion, such comparison is included here to illustrate that the developed model can also be run with the L_2 performance measure, albeit with an increase in computational complexity and little gain in accuracy of the obtained clusters. The following sub-sections will present the case studies along with results and discussions.

3.4.1 Case Study 1

The objective here is to examine the effect of increasing the number of days and clusters. First, the number of clusters is fixed and the number of days is increased and the results will be analyzed. Similarly, the number of days will be fixed in another set of simulations and the number of clusters is. Electricity demand for the province of Ontario was used to carry out this case study [108]. The demand of first 30 days of 2014 is considered (**Appendix A**, Table A-1). The following summarizes the simulation runs

- Day effect
    - 20 days and 3 clusters (20-3)
    - 25 days and 3 clusters (25-3)

  - 30 days and 3 clusters (30-3)
- Cluster effect
  - 20 days and 3 clusters (20-3)
  - 20 days and 4 clusters (20-4)
  - 20 days and 5 clusters (20-5)

The total number of runs is 10 (5 for each normal and sequence clustering). More details about the solutions obtained and the actual assignment of days to clusters is given in **Appendix A** (Tables A-2 to A-11). Summaries of these solutions along with the solution times will be detailed later.

In order to gain insight into the results, the following relative error function is used as a validation measure between the cluster and curve loads which correspond to that cluster.

$$error_{c,d,h}(\%) = \frac{Cl_{c,h} - L_{d,h}}{L_{d,h}} * 100\% \qquad \textbf{(3-17)}$$

Essentially, the above metric is the L_1 criterion scaled by the load curve to facilitate comparisons in the case when demand curves differ much in magnitude. In this way, the above error measure can be effectively used to evaluate performance because it is independent of system capacity and also independent of the unit of measurement. Indeed, this error measure is the mostly widely adopted error measure in utility forecasting literature even though high values of the error term can be anomalies instead of simply incorrect predictions[109]. Because of this last point, the error standard deviation of curves in the same cluster is also used to check that curves within the same clusters have high similarity while curves in different clusters have low similarity. This, in addition to the graphical/visual comparisons as will be made clear later.

GAMS/CPLEX [110] was used to conduct the runs on Inter(R) Xeon(R) 2.4 GHz (2 processors), 16 GB RAM workstation. We used tuned solver parameters with a tuning tool to accelerate the computational performance (only for normal clustering model). In addition, we used parallel mode computing to take advantage of the workstation's computational capability. This is because the default setting was taking a prohibitive long time to solve the model compared to the tuned parameters' instances. For example, it took 75 CPU minutes for default setting to solve the model for run 20-3 whereas it took only 123 CPU seconds using the tuned parameters. Here are the description of the tuned parameters used [111]:

- CUTPASS = -1; No limit for the number of passes that will be performed when generating cutting planes on a mixed integer model.

- HEURFREQ = -1; Do not use the node heuristic.
- PROBE = -1; Do not use probing.
- VARSEL = 4; Branch based on pseudo reduced costs.

Table 3-1 and Table 3-2 show the model statistics, optimal objective function value, solution time, error average and standard deviation for normal and sequence clustering of all runs (20-3 appeared twice in the table to make it easy to compare). The first three columns are for day effect and the rest are for cluster effect.

Model statistics for normal and sequence clustering are almost the same except for extra sets of constraints for sequence clustering. However, the solution time for normal clustering is far longer than for sequence clustering. The extra sets of constraints for sequence clustering reduce the size of the feasible region, results in a shorter solution time.

As can be noticed, increasing the model size by increasing either the number of days or clusters has a negative impact on solution time. In fact, the solution time tends to increase exponentially when model size increases for normal clustering. The increase of solution time in sequence clustering is not as severe as in normal clustering. This demonstrates that the model is hard to solve even with a small number of binary variables. The optimal objective function value increases with an increase in the number of days and drops with an increase in the number of clusters.

An ideal cluster would be compact and isolated [69]. In other words, an ideal cluster would have an error average close to zero with minimum standard deviation. The error average in normal clustering appears to reach stability with day increase while it drops with cluster increase. This indicates that more days added might fall into already established clusters rather than form new clusters. This also reveals that there is some sort of optimal day-cluster ratio. The error average in sequence clustering fluctuates with an increase of day or cluster. The error standard deviation in normal clustering decreases with both day and cluster increase. This holds true all the time for the cluster increase. The error average for both normal and sequence clustering are comparable in magnitude while the error standard deviation for sequence clustering is almost twice that for normal clustering.

Table 3-1: Summary of model and computational statistics for normal clustering (electricity demand)

	20-3	25-3	30-3	20-3	20-4	20-5
Constraints	8,660	10,825	12,990	8,660	11,540	14,420
Continuous variables	1,992	2,472	2,952	1,992	2,496	3,000
Binary variables	60	75	90	60	80	100
Optimal objective function value (MWh)	209,317.5	255,528.0	311,788.0	209,317.5	177,328.0	155,193.5
CPU time (s)	123	591	10,615	123	1,205	5,709
Error average (%)	0.49	-0.19	0.01	0.49	0.14	-0.17
Error Std (%)	3.73	3.58	3.51	3.73	3.12	2.79

Table 3-2: Summary of model and computational statistics for sequence clustering (electricity demand)

	20-3	25-3	30-3	20-3	20-4	20-5
Constraints	8,720	10,900	13,080	8,720	11,620	14,520
Continuous variables	1,992	2,472	2,952	1,992	2,496	3,000
Binary variables	60	75	90	60	80	100
Optimal objective function value (MWh)	353,157.0	453,443.0	522,376.5	353,157.0	312,150.0	271,054.0
CPU time (s)	3	4	5	3	5	22
Error average (%)	0.36	1.88	1.39	0.36	0.60	0.01
Error Std (%)	5.86	6.04	5.69	5.86	5.33	4.85

Figure 3-2 and Figure 3-3 show cluster 1 and its corresponding daily loads for run 30-3 of normal and sequence clustering respectively. As can be seen, normal cluster 1 captures the behavior and trend of all load curves. It is hard sometimes to spot the cluster since it overlaps with other load curves. Even in the case of an outlier, the cluster remains with the majority. Researchers when using k-means algorithm for clustering time-series data have used special algorithms to treat outliers but with no apparent success as discussed in section 4.1 this. However, this model showed no problem to include outliers in the cluster with no effect[92]. Sequence cluster 1 does the same except that it does not have the flexibility of normal clustering. As a result, there is an obvious gap between the loads even though the cluster remains with the majority.

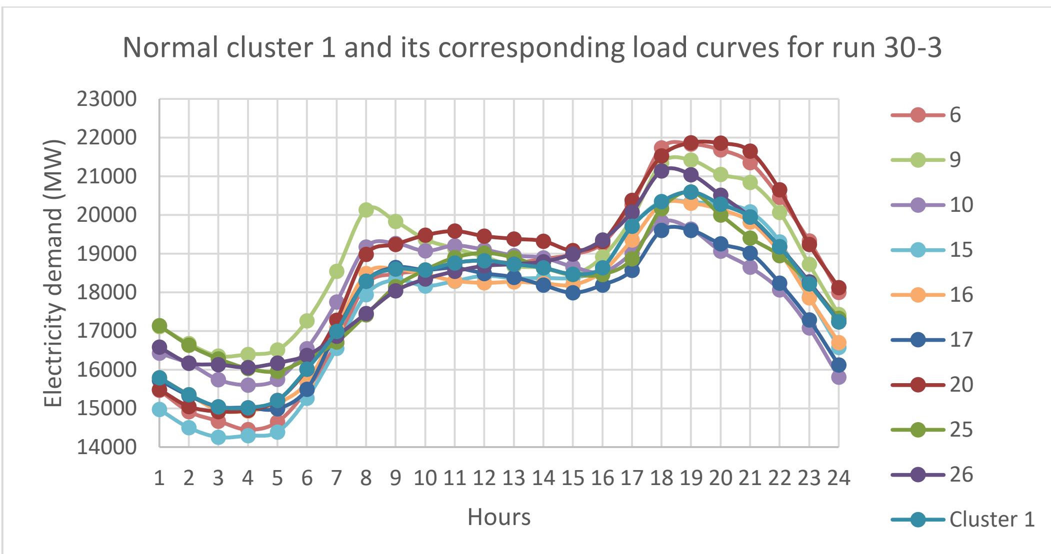

Figure 3-2: Normal cluster 1 and its corresponding load curves for run 30-3

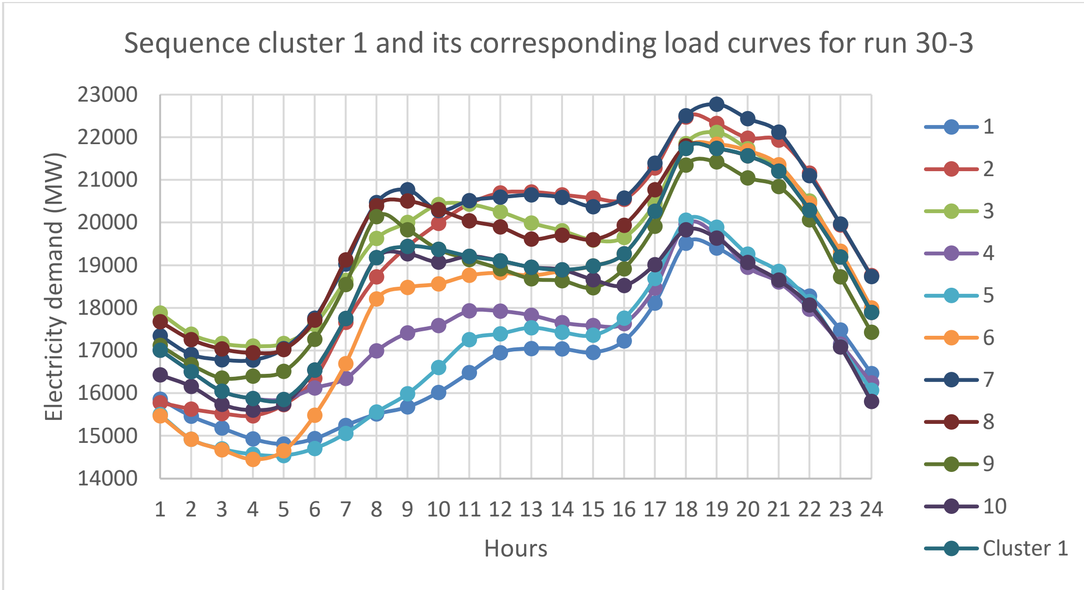

Figure 3-3: Sequence cluster 1 and its corresponding load curves for run 30-3

Figure 3-4 and Figure 3-5 present normal and sequence clusters for run 30-3 respectively. It seems that normal clusters 1, 2 and 3 capture the load for average, low and high demand respectively. One interesting finding of the model output is that the cluster always takes a value of curve loads rather than going between the curves. This was noticed while performing an error analysis. There were zeros in the error matrix because the cluster chose to take the value of that curve load. This is a well-known result due to the use of the L_1 criterion which gives an estimate of the median instead of the mean. The sequence clusters 1 and 3 seem to be almost the same. Sequence clustering might be required in several applications as discussed previously in order to have a schedule or plan with minimum set-up and transition times.

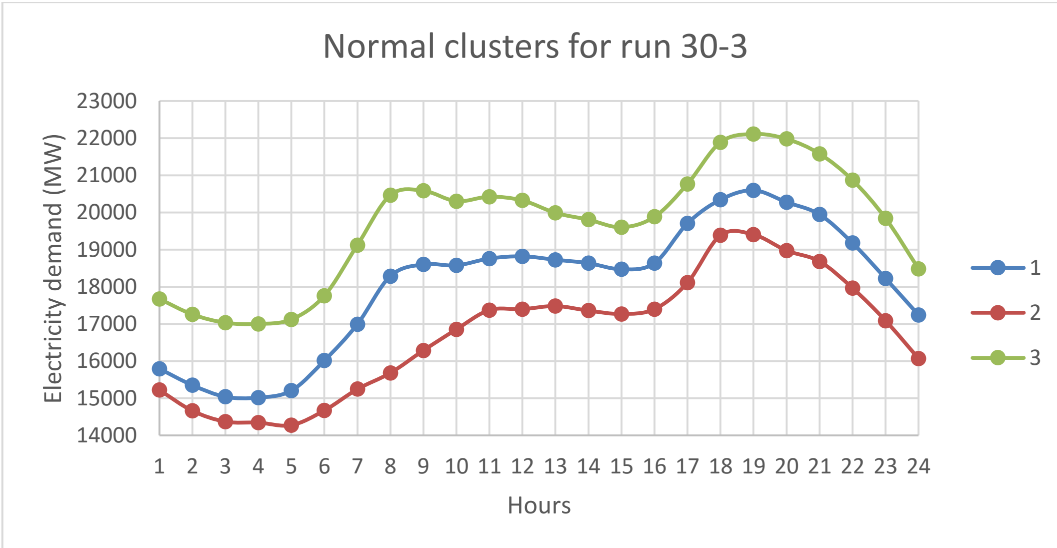

Figure 3-4: Normal clusters for run 30-3

Figure 3-5: Sequence clusters for run 30-3

Figure 3-6 and Figure 3-7 show error histograms of run 30-3 for normal and sequence clustering respectively. The normal clustering has a more even distribution than the sequence clustering. Moreover, the majority falls in bin 2.7 % for normal clustering while for sequence clustering, it is in bin 4.5 %.

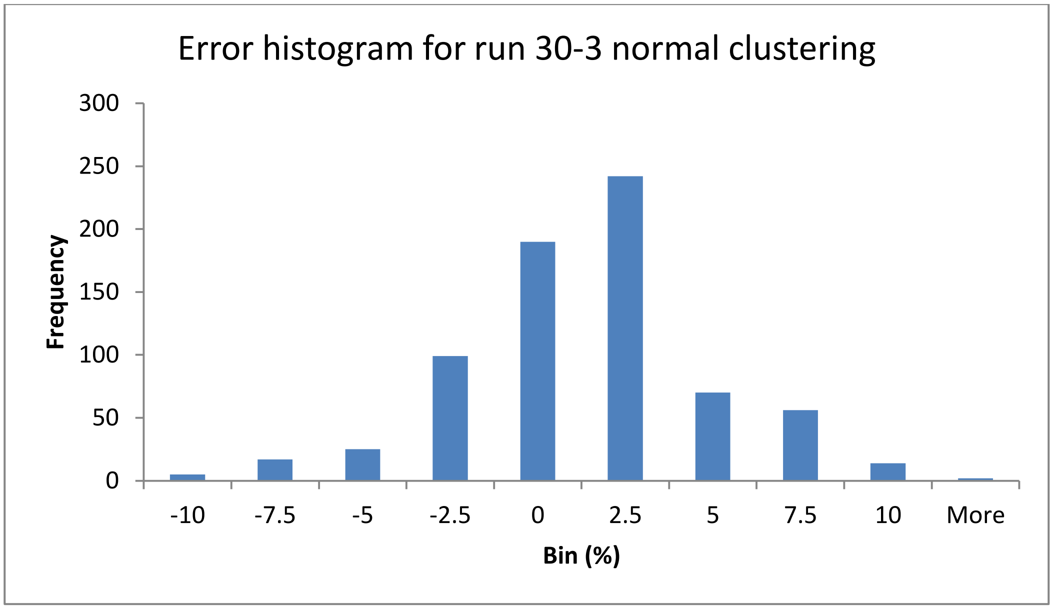

Figure 3-6: Error histogram for run 30-3 normal clustering

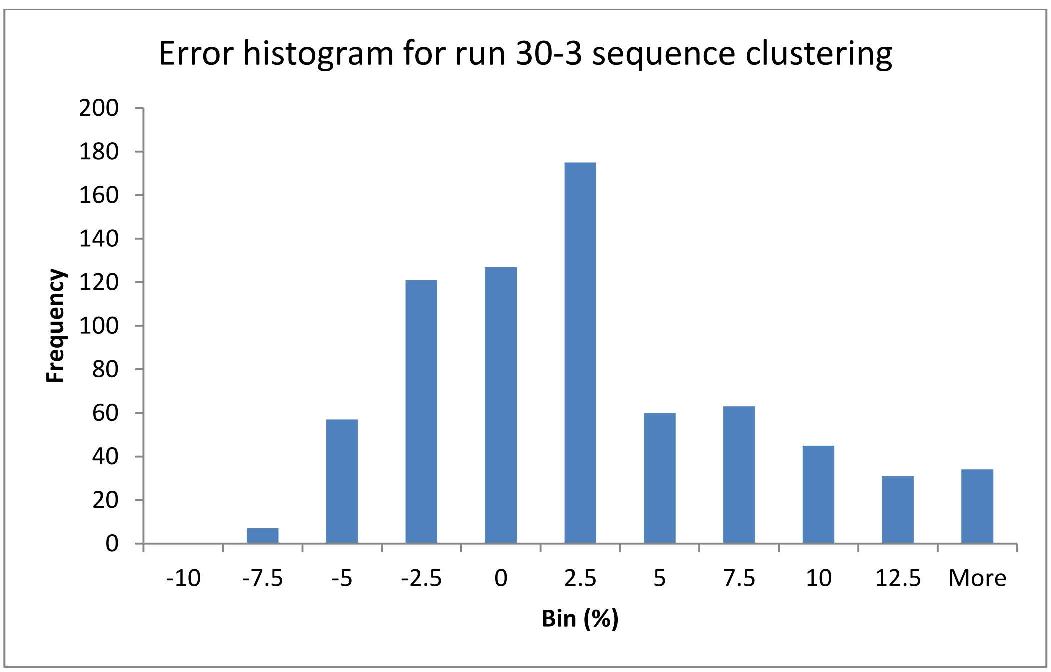

Figure 3-7: Error histogram for run 30-3 sequence clustering

In order to further compare the results of the L_1 criterion with that of the L_2 criterion, we carried out a clustering simulation for the 20-3 cases using both metrics. Although we have listed the advantages and disadvantages of both metrics in Section 3.1 and have also discussed the findings of the literature, the comparison here is aimed at showing that the use of the L_1 metric is legitimate in the case of clustering demand data and that similar results can be obtained. Table 3-3 shows the computational CPU times when using each metric as well as the error averages and error standard deviations. Figure 3-8 shows the obtained clusters by the two criterion. It is clear and as was discussed previously that the computational time using the L_2 metric is much higher than that of the L_1 metric. The L_1 metric maintains linearity of the model at the expense of a slight increase in problem size (since the remodeling of the absolution value term creates additional variables). The solution of the mixed integer linear problem requires in this case less computational time and the solution is guaranteed to be a global optimum. There is much talk that in the case of regression analysis, there can be multiple solutions to an LP (multiple corner points) as compared to the Euclidian distance case which leads to one single solution but in the current application of trajectory clustering this is not the case. Furthermore, the sequence based clustering that is aimed at obtaining clusters to be embedded in scheduling and planning models with the objective of minimizing set-up and changeovers impose additional constraints in the model.

The obtained clusters using both metrics are almost identical. There are minor differences which cannot be seen in Figure 3-8. Only the initial hours of the day seem to differ but Table 3-3 sheds more light on these differences and as quantified by the error average and the error standard deviation, the difference in the obtained clusters is not significant.

Table 3-3: Statistical comparison between the L_1 (IAE) and L_2 (Euclidean) metrics for run 20-3

	L_1 (IAE)	**L_2 (Euclidean)**
CPU time (s)	123	8,211
Error average (%)	0.494	0.467
Error Std (%)	3.728	3.712

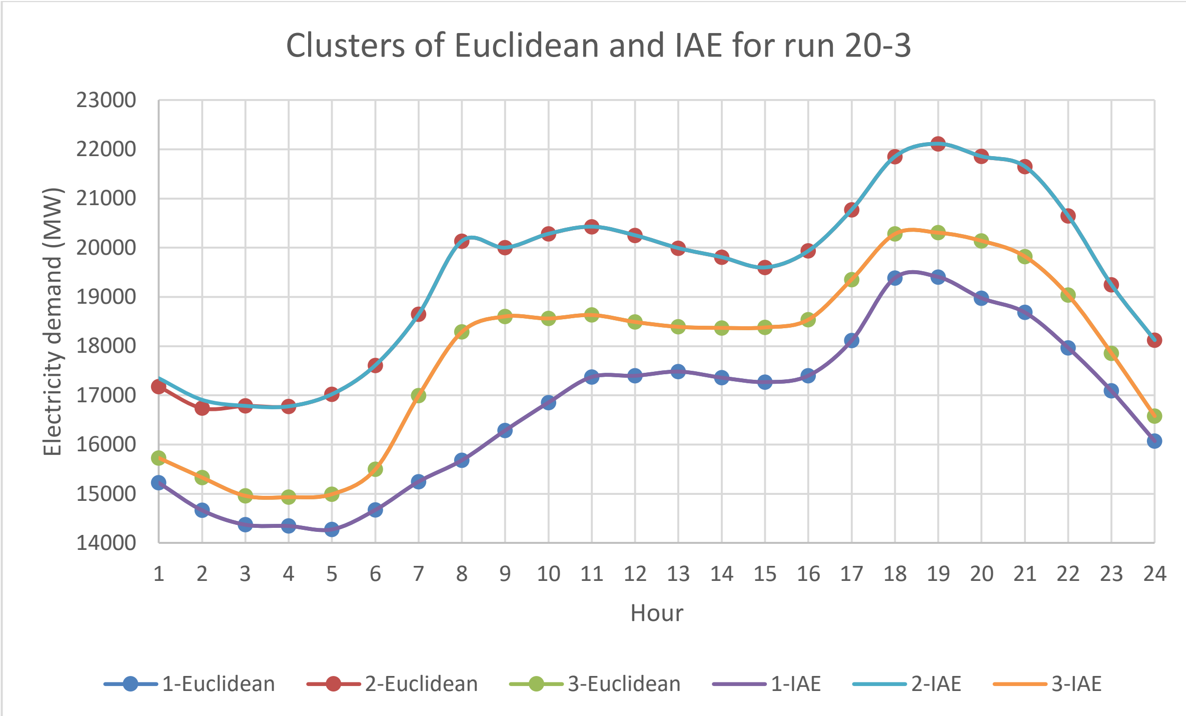

Figure 3-8: Clusters of L_1 (IAE) and L_2 (Euclidian) metrics for the case 20-3

Table 3-4: Details of the obtained clusters using the L_1 (IAE) and L_2 (Euclidean) metrics for run 20-3

Hour	L_1 (IAE) clusters			L_2 (Euclidean) clusters		
	1	2	3	1	2	3
1	15,223	17,347	15,728	15,223	17,177	15,728
2	14,664	16,909	15,330	14,664	16,739	15,330
3	14,371	16,786	14,959	14,371	16,785	14,959
4	14,345	16,777	14,931	14,345	16,776	14,931
5	14,276	17,024	14,991	14,276	17,023	14,991
6	14,671	17,611	15,496	14,671	17,610	15,496
7	15,246	18,647	16,994	15,246	18,646	16,994
8	15,681	20,136	18,289	15,681	20,135	18,289
9	16,287	20,002	18,604	16,287	20,001	18,604
10	16,855	20,283	18,562	16,855	20,282	18,562
11	17,374	20,428	18,632	17,374	20,427	18,632
12	17,398	20,253	18,491	17,398	20,252	18,491
13	17,483	19,992	18,392	17,483	19,991	18,392
14	17,359	19,808	18,371	17,359	19,807	18,371
15	17,270	19,600	18,380	17,270	19,599	18,380
16	17,400	19,935	18,537	17,400	19,934	18,537
17	18,112	20,771	19,354	18,112	20,770	19,354
18	19,386	21,852	20,280	19,386	21,851	20,280
19	19,406	22,113	20,305	19,406	22,112	20,305
20	18,976	21,857	20,140	18,976	21,856	20,140
21	18,685	21,651	19,822	18,685	21,650	19,822
22	17,967	20,650	19,036	17,967	20,649	19,036
23	17,090	19,249	17,857	17,090	19,248	17,857
24	16,070	18,123	16,579	16,070	18,122	16,579

3.4.2 Case Study 2

The aim of this case is to illustrate the wide applications of the proposed clustering algorithm. Solar radiation data of Orlando International Airport was used in this case study [112]. The solar radiation profiles of the first 30 days of the year 2010 are considered (see **Appendix A**, Table A-12). They will be grouped into 3 clusters using both normal and sequence clustering.

Table 3-5 summarizes the results for Case Study 2 while Figure 3-9 and Figure 3-10 show cluster 1 and its corresponding daily solar loads for run 30-3 of normal and sequence clustering respectively. Tables A-13 and A-14 give more details on the actual assignment of days to clusters and the solutions for normal and sequence clustering. The objective function for normal clustering is less than that for sequence clustering. This is normal since sequence clustering imposes the addition restriction that days must fall within a certain sequence. On the other hand, the solution time for normal clustering is greater than that for sequence clustering and this is as discussed for the previous case study due to the integrality gap of both models: the additional sequence constraints are actually helping in terms of computational complexity by leading to a tighter model. The error average for normal clustering is less compared to sequence clustering while the standard deviation is slightly higher. Nevertheless, the mathematical programming models both succeeded in obtaining appropriate clusters that can facilitate the resolution of large multi-scale mathematical programming models as will be discussed further in **Chapter 5** and **Chapter 6**.

The results of both case studies show that the models presented in this chapter are effective in obtaining clusters of trajectories in time-series data and following completely a mathematical programming approach. The case studies considered are however small (at most 30 days) and the computational time was relatively large. Although a 30 day span can cover a wide set of applications in demand scheduling and in particular customer clustering, it will be advantageous if the algorithm is able to handle sets of data in the range of one year or more. This is dealt with in the next chapter.

Table 3-5: Results summary for Case Study 2 (Solar radiation)

	Normal	Sequence
Objective function (Wh/m^2)	2,414	2,725
CPU time (s)	3055	3
Error average (%)	0.316	0.391
Error Std (%)	5.408	4.459

Figure 3-9 and Figure 3-10 show cluster 1 and its corresponding daily solar loads for run 30-3 of normal and sequence clustering respectively. The cluster overlaps with the load curves as in case study 1. Moreover, the cluster always follows the majority. As can be noticed, both clusters almost look alike, which makes the choice between normal and sequence clustering hard to distinguish. However, normal clustering might outperform sequence clustering if a longer horizon is considered.

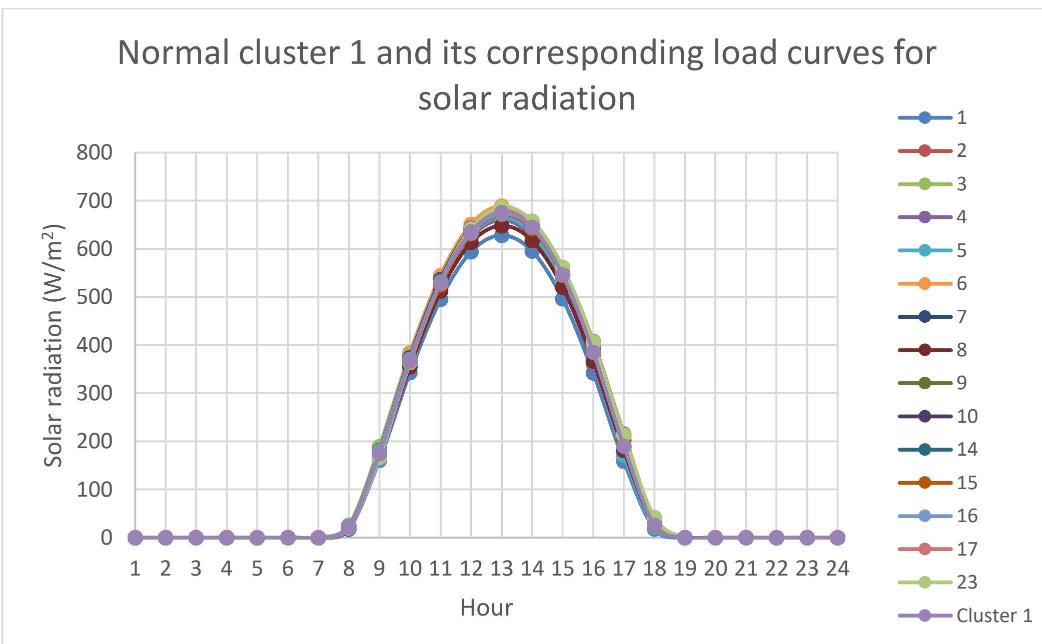

Figure 3-9: Normal cluster 1 and its corresponding load curves for solar radiation

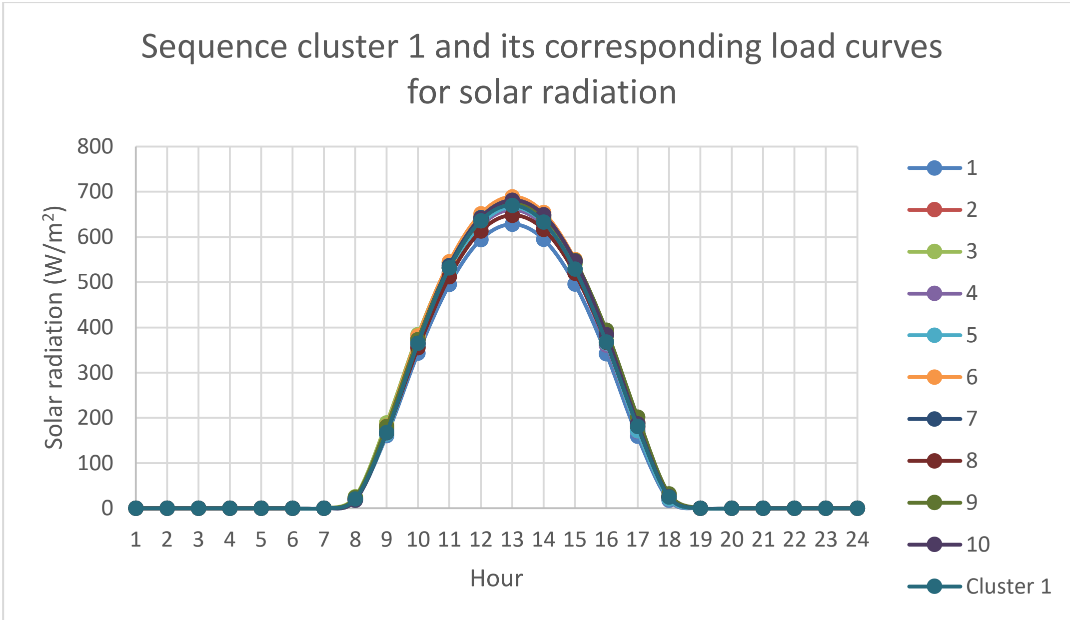

Figure 3-10: Sequence cluster 1 and its corresponding load curves for solar radiation

3.5 Conclusion and recommendations

The clustering algorithm proposed in this chapter possesses a unique feature to provide both normal and sequence clustering using the same algorithmic framework. Despite the model’s simplicity, the computational complexity of this clustering algorithm is very evident as illustrated by the findings of this chapter. Normal clustering and because of less restrictions imposed on the trajectories obtained outperforms in general sequence clustering in term of objective function, average and standard deviation error. Therefore in the applications when sequencing constraints are not needed, normal clustering should be utilized. However, the solution time for sequence clustering is much shorter than that of normal clustering.

One way to deal with the computational complexity is to carefully study the MILP model and reformulate it. It is also possible to apply a traditional decomposition method, like Bender or Lagrangean decomposition. Moreover, one can use hints from parameters tuning outputs to come up with a solution method to tackle such issue. The input parameters in process systems engineering usually consist of multiple attributes while this clustering algorithm handles only one attribute at a time. The scalability of this algorithm would not be a problem since it is MILP-based. However, involving multiple attributes such as simultaneous clustering of electricity demand and solar radiation will result in a multi-objective optimization problem.

This clustering algorithm could be applied to any two dimensional data like the examples used in this chapter (electricity demand and solar radiation). The most suitable applications of this clustering algorithm are long-term scheduling and integrated scheduling and planning problems. The algorithm can also be useful in clustering customer demand in, for example, energy hubs and can also be of tremendous help in forecasting applications (it is easier to forecast clusters than individual demand days). Moreover, little has been done regarding solution accuracy when clustering has been applied. Therefore, one can solve a problem like unit commitment for a certain period. The inputs must be clustered using normal and sequence clustering. The clustered demand from the different clustering algorithms will be used as inputs for the unit commitment model in **Chapter 5**. In this way, we have a clear benchmark to assess the solution accuracy. In addition, we can tell which clustering algorithm will perform better in certain application.

Chapter 4
A mathematical programming based heuristic clustering algorithm for shape-based time series data

4.1 Introduction

The main objective of this chapter is to overcome the computational complexity of the clustering models described in the previous chapter. As was illustrated on the case studies of **Chapter 3**, the computational burden associated with solving the MILP model with the L_1 metric, although much less than that of the L_2 metric, can still represent a drawback in the application of the models for large planning horizons. The case studies presented in the previous chapter span horizons of up to 30 days or 30 different demand patterns. In this chapter we aim at extending the applicability of the presented models for large horizons. The modeling framework maintains linearity and is mathematical programming based. We develop a heuristic algorithm in this chapter that employs the mathematical programming models as its main building block and utilizes an iterative scheme that compares a lower and upper bound solution. Such mathematical programing based heuristics have been utilized in the past and represent suitable solution schemes to tackle large scale mathematical programming models [75,113].

4.2 Proposed heuristics

The structure of the proposed heuristics follows closely the basic procedure of *k*-means algorithm[114]; except that the clusters are obtained through the mathematical programming models of chapter 3. In addition, the *k*-means algorithm is often utilized for one dimensional time-series data; although there are recent versions that are able to deal with trajectories. The *k*-means algorithm starts with initializing *k* partitions randomly, calculates a cluster prototype matrix *M*, assign each object in the data set to the nearest cluster, recalculate the cluster prototype matrix *M* based on the current partition, and repeat the procedure until there is no change if the obtained clusters.

In a similar manner, we propose a heuristic that starts with generating *n* random clusters or scenarios (Figure 4-1). The scenarios can be generated in Excel by randomizing between maximum and minimum of each hour for the entire demand. The procedure then considers each scenario separately and at a first attempt fix the clusters in the MILP model and solve the resulting integer program for day assignment. This gives an upper bound on the solution. Next the day assignment is fixed and an LP model is solved to get a lower bound solution. If the difference between the lower and upper bounds is within an acceptable prespecified range then the solution is saved as current best solution and the next scenario is considered. Otherwise, the procedure for a given scenario is repeated between fixing clusters then fixing day assignment until the lower and upper bounds are within the acceptable tolerance. Then the procedure continues with the next scenario in the list until all scenarios are considered. In the case of normal clustering the model as given by equations 3-1, 3-2, and 3-11 to 3-16 is used while for sequence clustering the model as given by equations 3-1, 3-2, 3-8, and 3-11 to 3-16 is used. The heuristic construction is therefore general and adaptable to either normal or sequence clustering. Furthermore, the Euclidian metric L_2 can also be utilized if desired instead of the L_1 metric by changing Eq. 3-1 appropriately. The computational effort of using the L_2 metric as compared to the L_1 metric will be illustrated on a case study in the next section.

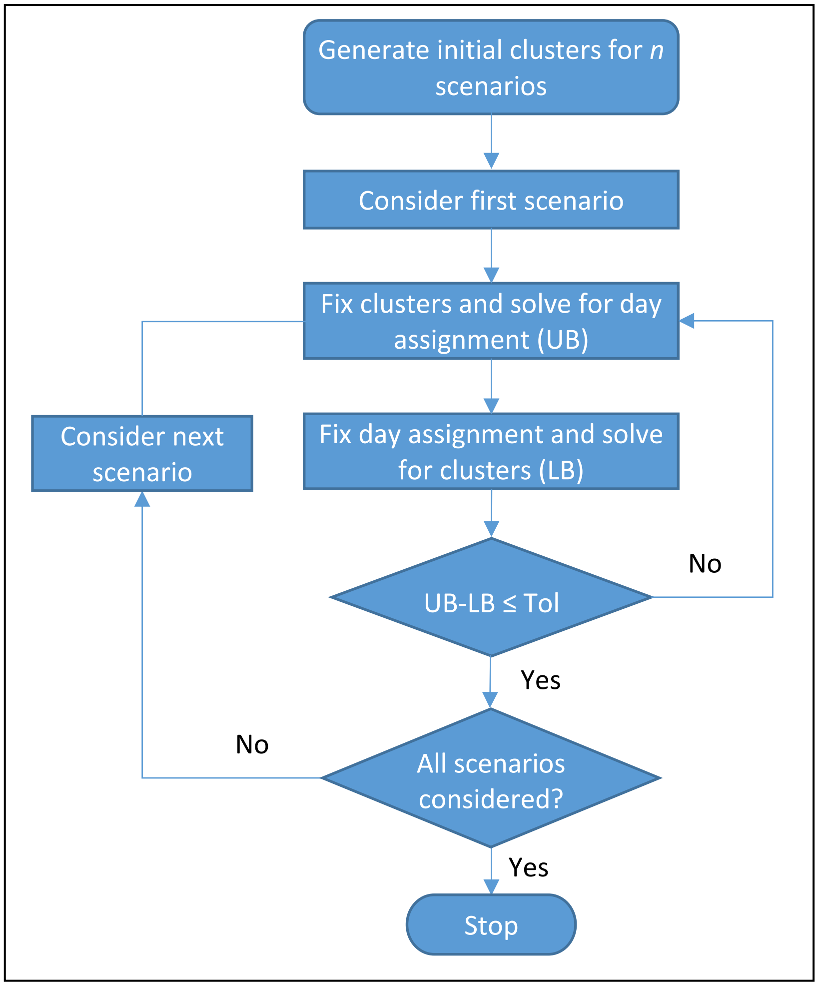

Figure 4-1: Flowchart of proposed heuristics

4.3 Case studies

There are two case studies in this chapter. The first case study is to validate the proposed heuristics while the second considers the whole year electricity demand for clustering. GAMS/CPLEX[110] was used to conduct the runs on Inter(R) Xeon(R) 2.4 GHz (2 processors), 16 GB RAM workstation.

4.3.1 Case Study 1

The goal of case study 1 is to apply the proposed heuristic algorithm and study its ability to reach the optimal solution. The same runs in case study 1 of **Chapter 3** will be used in order to determine how far the solution from the optimal one. The initial list of scenarios for the heuristic was generated in Excel by randomizing between the maximum and minimum of each hour. One hundred scenarios were generated for each run. The algorithm tolerance was set to 10^{-4}.

Table 4-1 and Table 4-2 present result summaries for normal and sequence clustering using the proposed algorithm respectively. The proposed algorithm successfully found the optimal solution for all the runs. However, chance of reaching the optimal solution varied among the runs. The chance of reaching optimal solution was higher for normal clustering. It took a few minutes to solve the model for both normal and sequence clustering.

Table 4-1: Results summary for normal clustering for 30 days using proposed algorithm

	Normal				
Run	20-3	20-4	20-5	25-3	30-3
Best solution found (MWh)	209,317.5	177,328	155,193.5	255,528	311,788
Solution time (mm:ss)	4:01	5:10	6:31	5:09	5:28
Number of scenarios found optimal solution (out of 100)	35	9	4	72	16

Table 4-2: Results summary for sequence clustering for 30 days using the proposed heuristic algorithm

	Sequence				
Run	20-3	20-4	20-5	25-3	30-3
Best solution found (MWh)	353,157	312,150	271,054	453,443	522,376.5
Solution time (mm:ss)	4:19	6:09	9:37	6:06	7:32
Number of scenarios found optimal solution (out of 100)	15	1	1	25	23

4.3.2 Case Study 2

Case Study 2 aims to apply the proposed algorithm for the electricity demand of the whole year, which cannot be tackled using the full scale model. The initial guess was generated in Excel by randomizing between maximum and minimum of each hour for the entire demand. The only difference in the initial guess for sequence clustering is that days are first partitioned based on days to clusters ratio (the fractioned ratio should be rounded down). For example, if we have 30 days and 3 clusters, the ratio would be 10 and this will result in 3 partitioned groups of days. After that, the initial guess for cluster 1 will be generated by randomizing between maximum and minimum of each hour for the first partitioned days. The same applies for cluster 2 and cluster 3. This procedure results in better objective function. The ratio itself could be optimized by a careful study of the demand. Runs for this case study are 4, 5, 6 and 7 clusters with 365 days for each normal and sequence clustering so runs are 8 in total. 25 scenarios were generated per run. It is worth mentioning that parameter tuning was used for sequence clustering to reduce solution time. The algorithm tolerance was 10^{-4}.

Table 4-3 and Table 4-4 show result summaries for Case Study 2. The heuristic was able to obtain good clusters for both normal clustering and sequence clustering. Clearly normal clusters had smaller error% averages and smaller standard errors and this is expected because of the additional restriction on sequence clustering. The heuristic can easily find both types of clusters for any desired application. The tables show that an increase in the number of clusters decreases the objective function, which is in agreement with findings in the previous chapter. In addition, the error average of normal clustering fluctuates as the number of clusters increases while the standard deviation declines. The same trend is observed in case study 1 in the previous chapter. The solution time for sequence clustering is shorter than for normal clustering. Figure 4-2 presents the error histogram for run 365-6 of normal clustering. As can be noticed, the error is normally distributed around bin 0.4 %.

Table 4-3: Summary of the results for normal clustering for 365 days using the heuristic algorithm

	Normal clustering			
Number of clusters	4	5	6	7
Best solution found (MWh)	5,072,038	4,504,404	4,223,551	3,996,387
Average solution time per scenario (min)	4.08	18.48	11.32	14.04
Error average (%)	-0.0150	0.0775	0.0008	0.0344
Error Std (%)	4.9641	4.5263	4.2603	3.9922

Table 4-4: Summary of the results for sequence clustering for 365 days using the heuristic algorithm

	Sequence clustering			
Number of clusters	4	5	6	7
Best solution found (MWh)	7,985,016.5	7,823,143	7,236,448	7,161,205
Average solution time per scenario (min)	1.04	1.68	5.36	8.16
Error average (%)	0.6513	0.7956	0.9314	0.9442
Error Std (%)	7.9932	7.9346	7.5127	7.5335

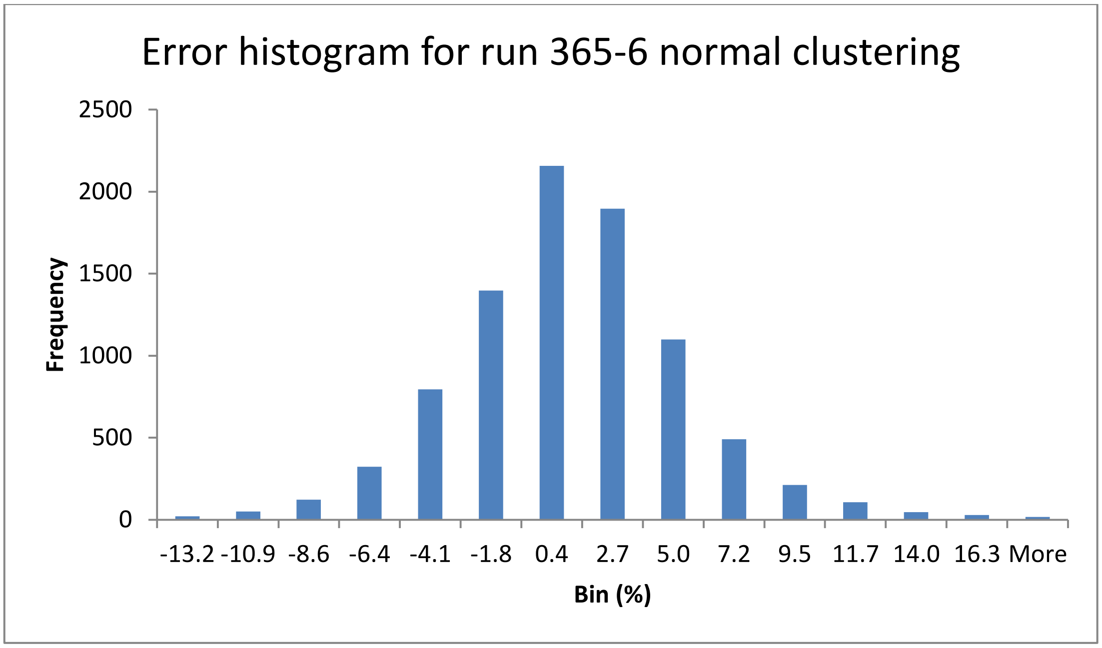

Figure 4-2: Error histogram for run 365-6 normal clustering

Figure 4-3 to Figure 4-10 show clusters and the corresponding day assignments for normal clustering. More details about these clusters and the corresponding assignments are given in **Appendix B**. The initial 4 clusters in run 365-4 do not change much as additional clusters are introduced in other runs. Moreover, clusters 5, 6 and 7 seem to overlap with the first four clusters. A careful look at day assignment figures (**Appendix B**) shows the switch of day assignments from the first four clusters to the new clusters. These observations suggest that the first four clusters seem to be optimal and the algorithm is trying to find a new cluster to improve the objective function resulting in a lower relative error or standard deviation.

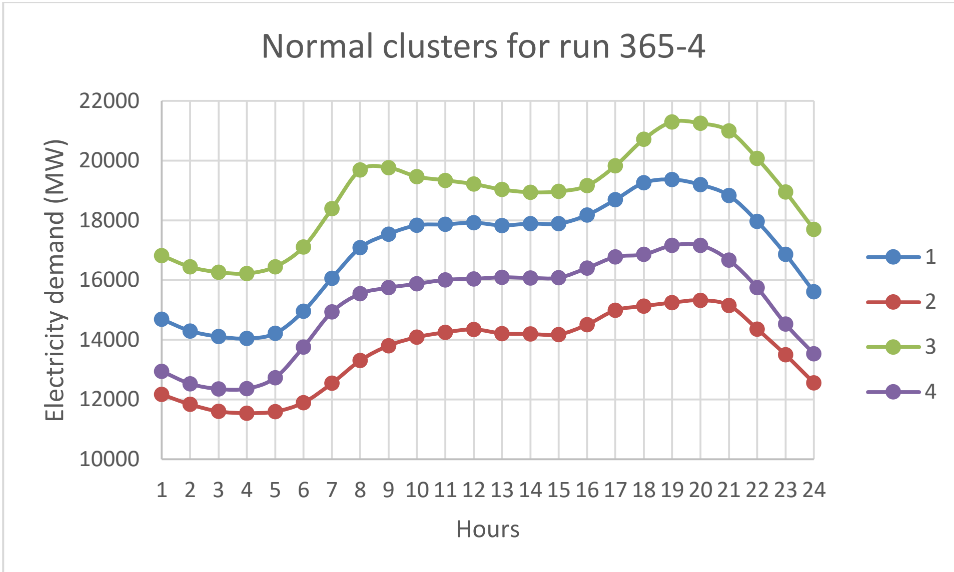

Figure 4-3: Normal clusters for run 365-4

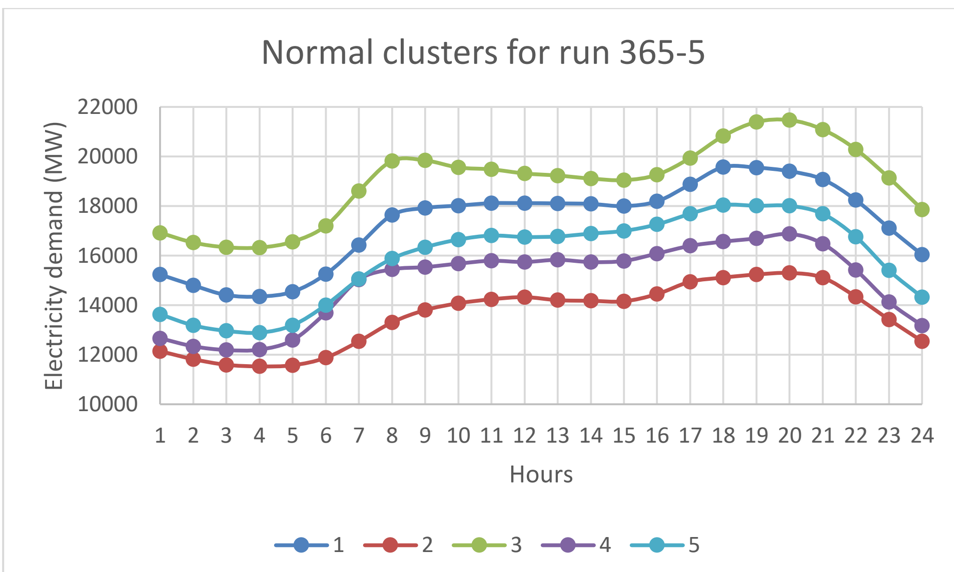

Figure 4-4: Normal clusters for run 365-5

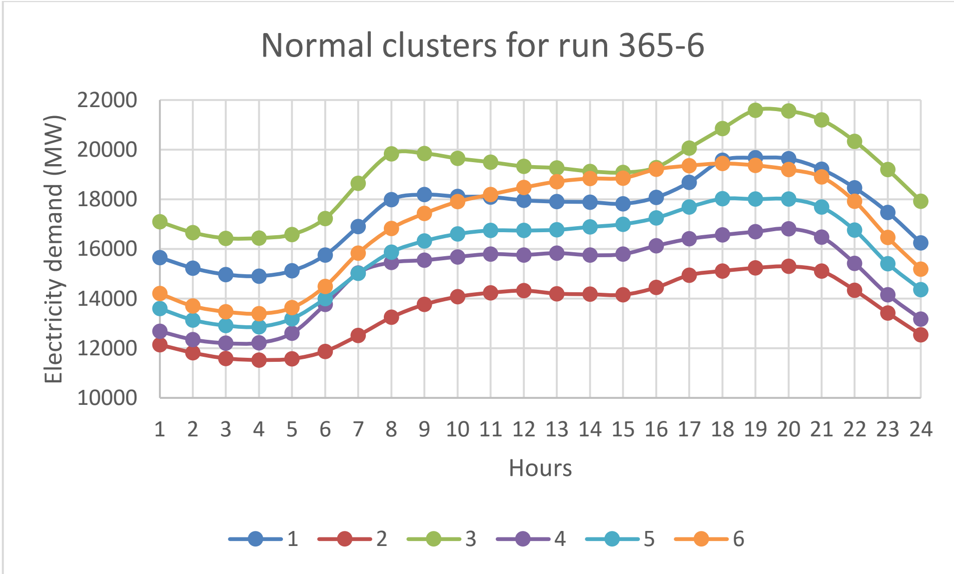

Figure 4-5: Normal clusters for run 365-6

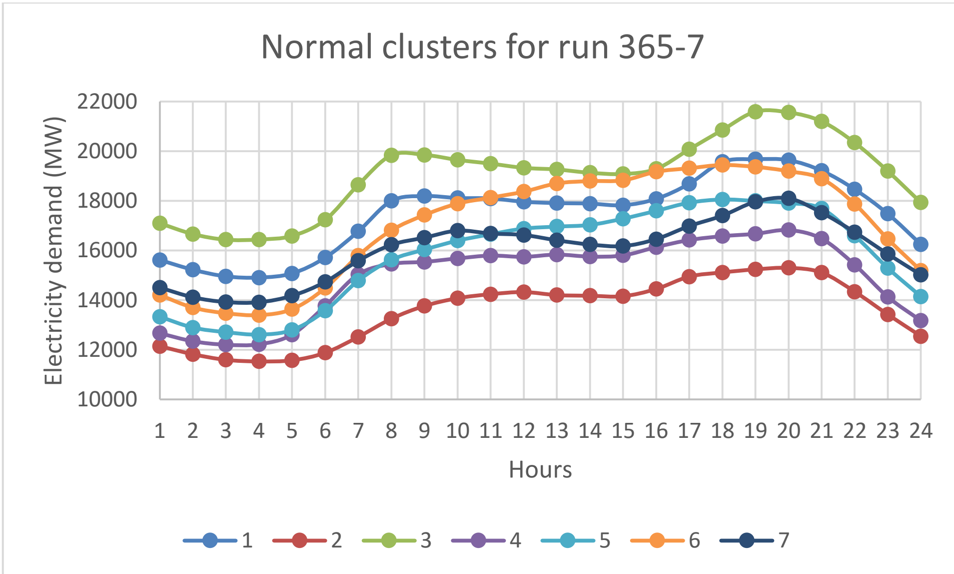

Figure 4-6: Normal clusters for run 365-7

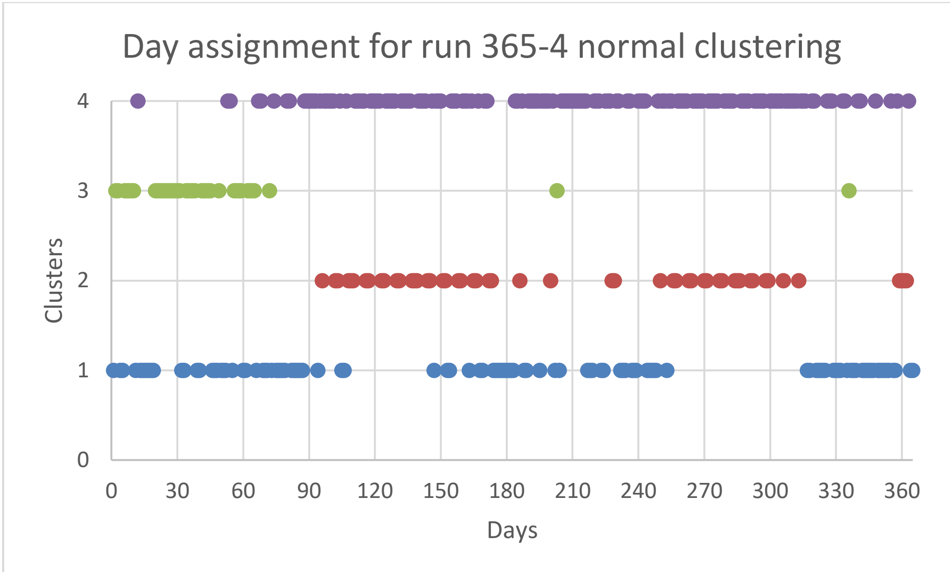

Figure 4-7: Day assignment for run 365-4 normal clustering

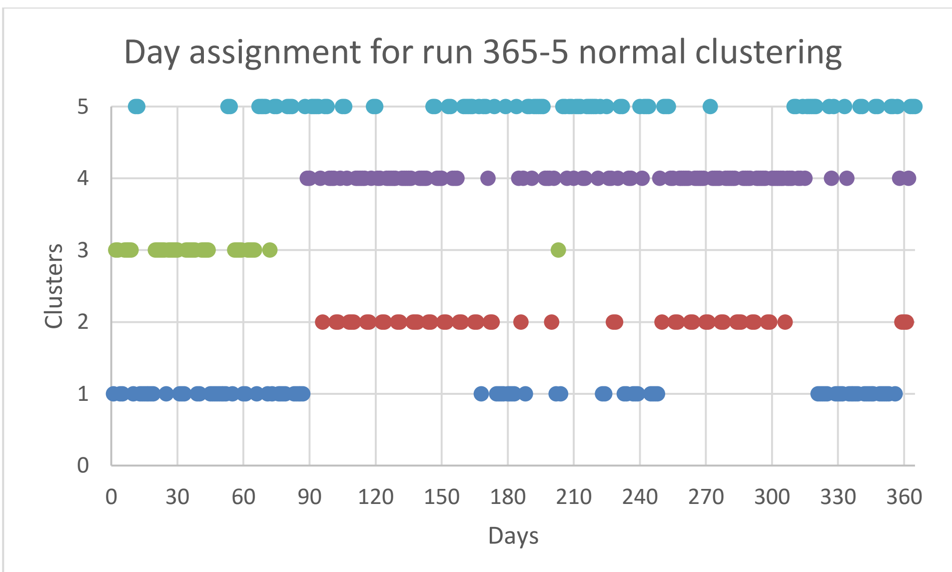

Figure 4-8: Day assignment for run 365-5 normal clustering

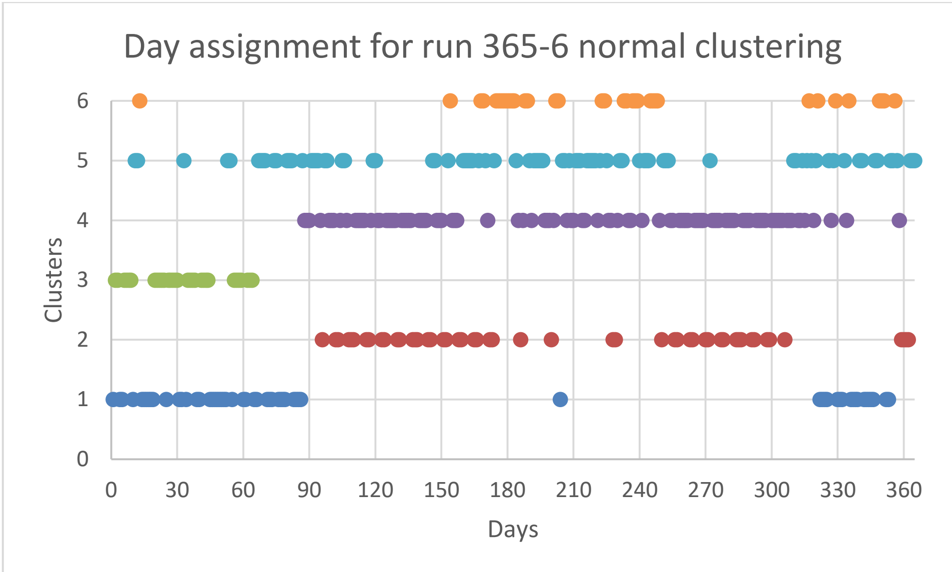

Figure 4-9: Day assignment for run 365-6 normal clustering

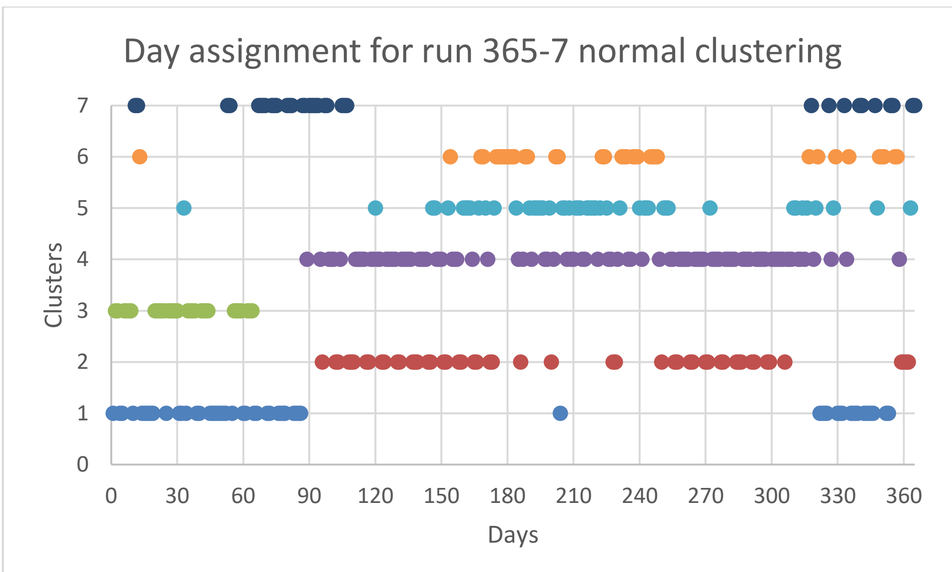

Figure 4-10: Day assignment for run 365-7 normal clustering

Similarly, Figure 4-11 to Figure 4-18 show clusters and day assignments for sequence clustering. The details of these clusters and day assignments are again given in **Appendix B**. As can be seen from the figures, the first and last sequence clusters remain relatively unchanged while the intermediate sequence clusters evolve as the number of cluster increases. The same is observed with day assignments. Notice also how sequence clustering succeeds in getting days in clusters that abide to a certain sequence (i.e. first n_1 days in cluster one, second n_2 days in cluster 2, etc.).

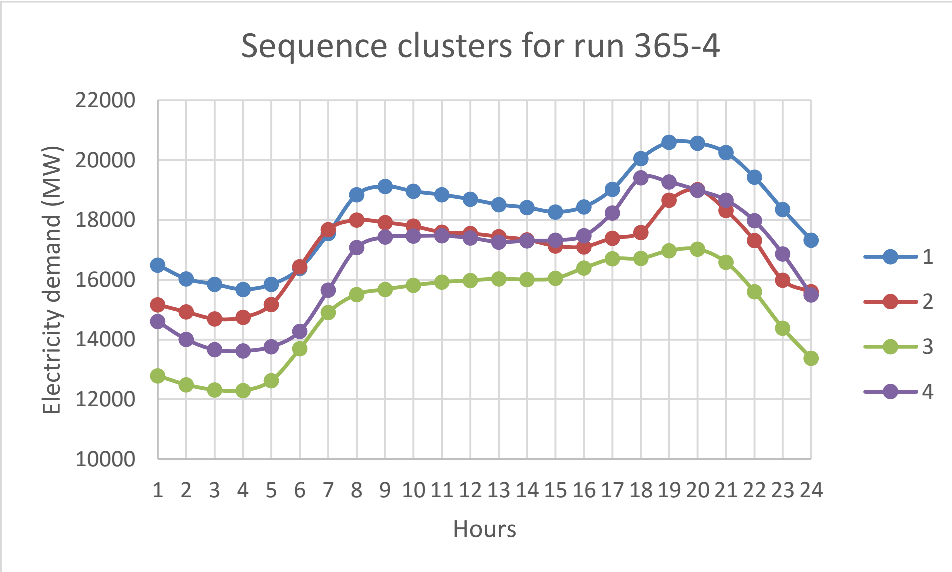

Figure 4-11: Sequence clusters for run 365-4

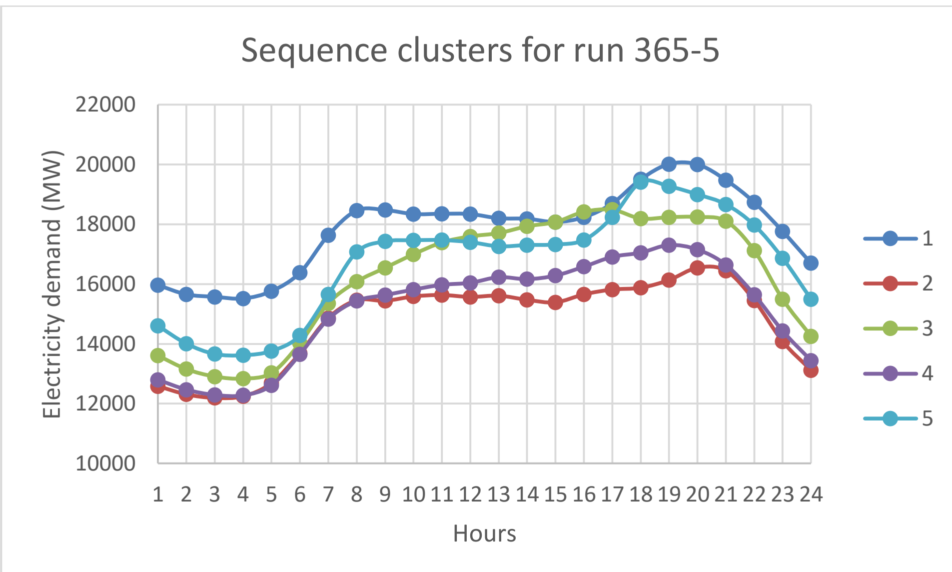

Figure 4-12: Sequence clusters for run 365-5

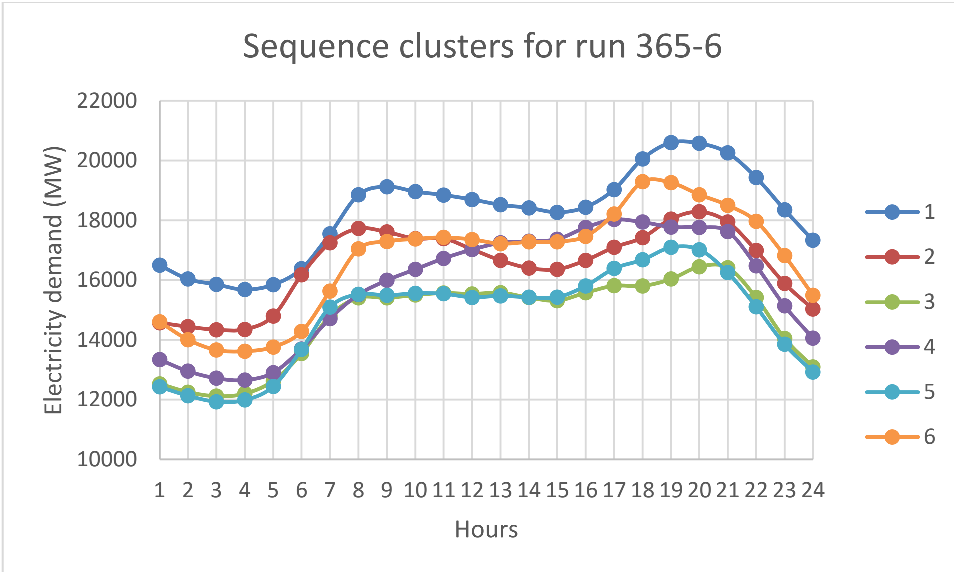

Figure 4-13: Sequence clusters for run 365-6

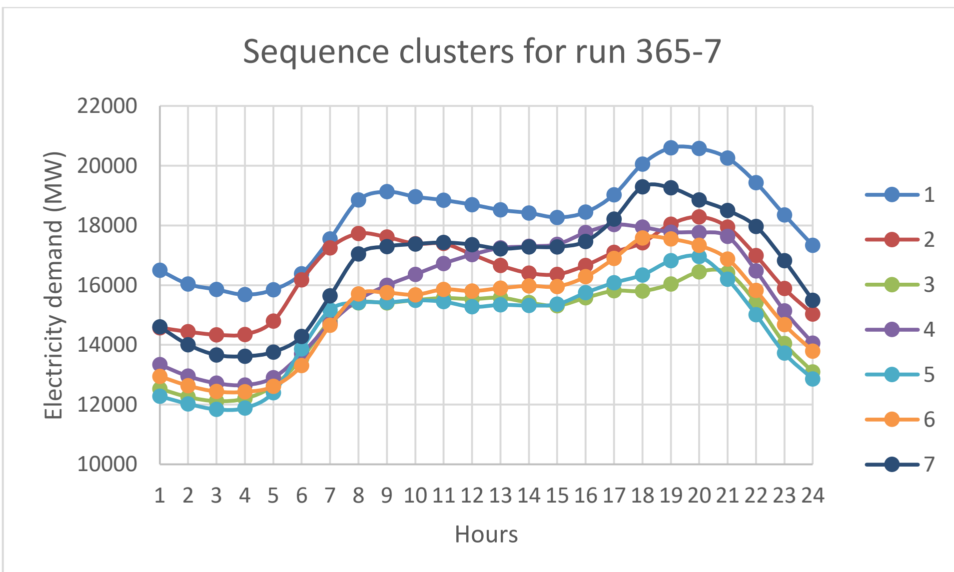

Figure 4-14: Sequence clusters for run 365-7

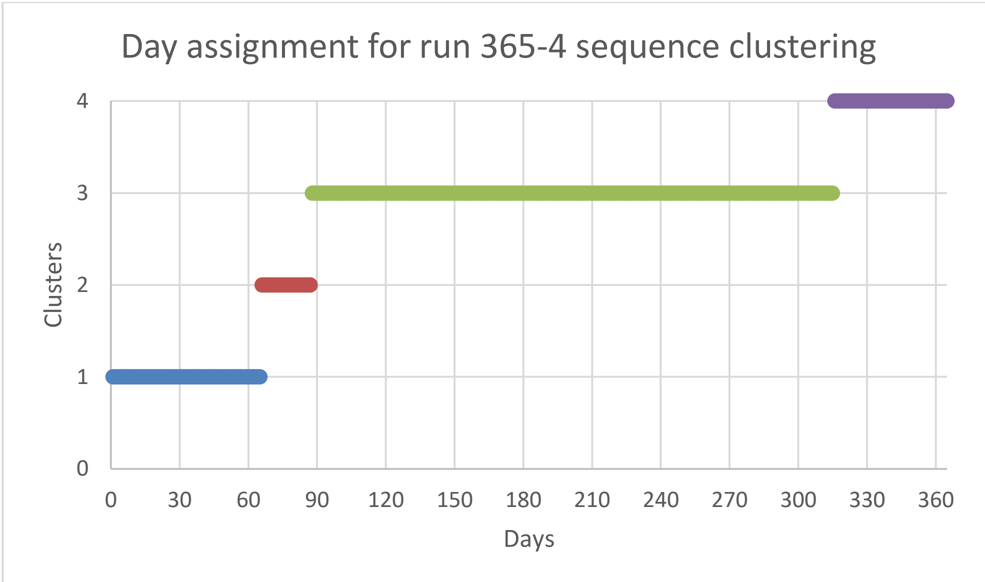

Figure 4-15: Day assignment for run 365-4 sequence clustering

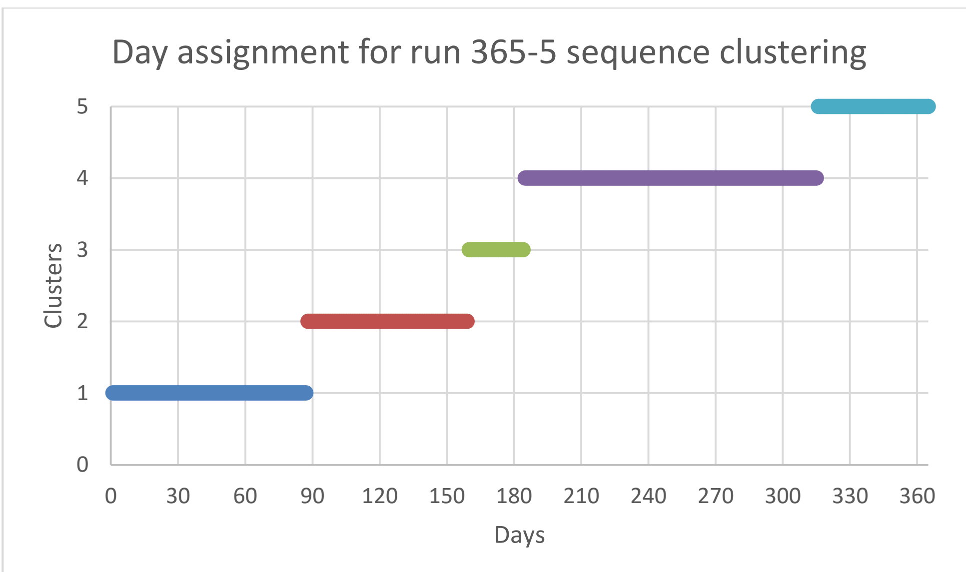

Figure 4-16: Day assignment for run 365-5 sequence clustering

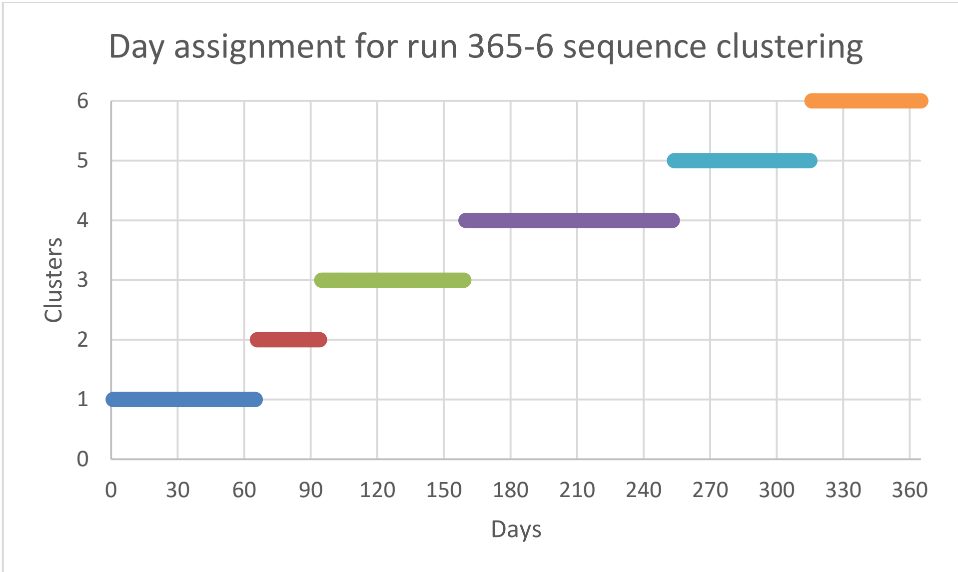

Figure 4-17: Day assignment for run 365-6 sequence clustering

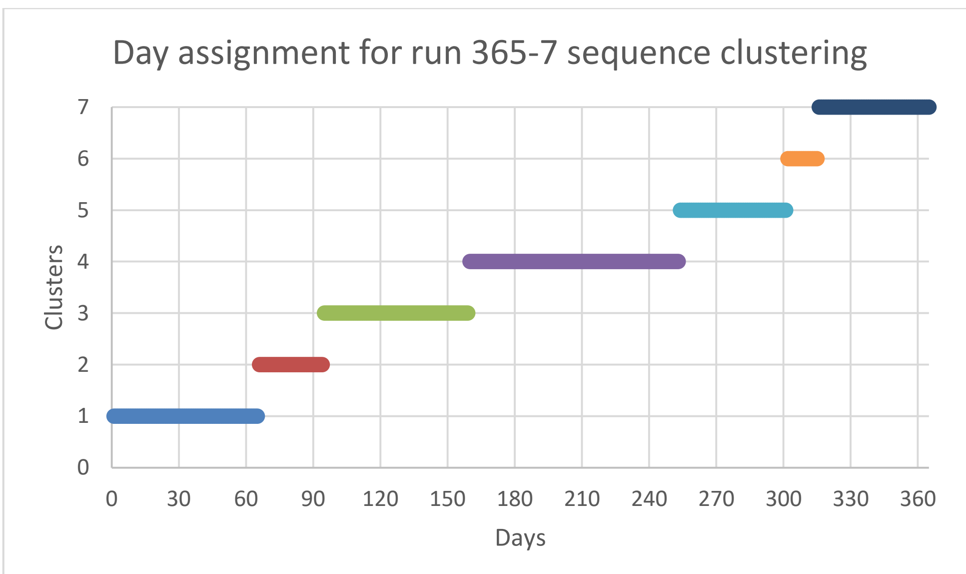

Figure 4-18: Day assignment for run 365-7 sequence clustering

In order to further compare the results of the L_1 criterion with that of the L_2 criterion for the case of a full year of data (365 days), we carried out also a clustering simulation using both L_1 and L_2 metrics. Although we have listed the advantages and disadvantages of both metrics in Section 3.1 where we have discussed the findings of the literature, and also compared the performance of the two metrics for the twenty days case in **Chapter 3**, the comparison here is aimed at showing further that the use of the L_1 metric is legitimate in the case of clustering demand data and that similar results can be obtained at a much less computational cost. For illustration purposes, the results of 365-4 are presented here. The other cases of less or more clusters followed the same trend in terms of results and the same conclusions are drawn as will be discussed for the 365-4 case. Table 4-5 shows in row one the CPU times when using each metric for the case of four clusters. The table also shows the error averages and error standard deviation (rows two and three, respectively). It is clear from Table 4-5 that the computational time using the L_2 metric is a 100 fold higher than that of the L_1 metric.

Figure 4-19 shows the obtained clusters by the two criterion. The obtained clusters using both metrics are almost identical. There are minor differences which cannot be seen in the Figure 4-19 and Table 4-6 sheds more light on these differences and as quantified by the error average and the error standard deviation, the difference in the obtained clusters is not significant.

Table 4-5: Statistical comparison between L_1 (IAE) and L_2 (Euclidean) metrics for run 365-4

	L_1 (IAE)	**L_2 (Euclidean)**
CPU time (s)	6,139	1,224,137
Error average (%)	-0.015	0.034
Error Std (%)	4.964	4.954

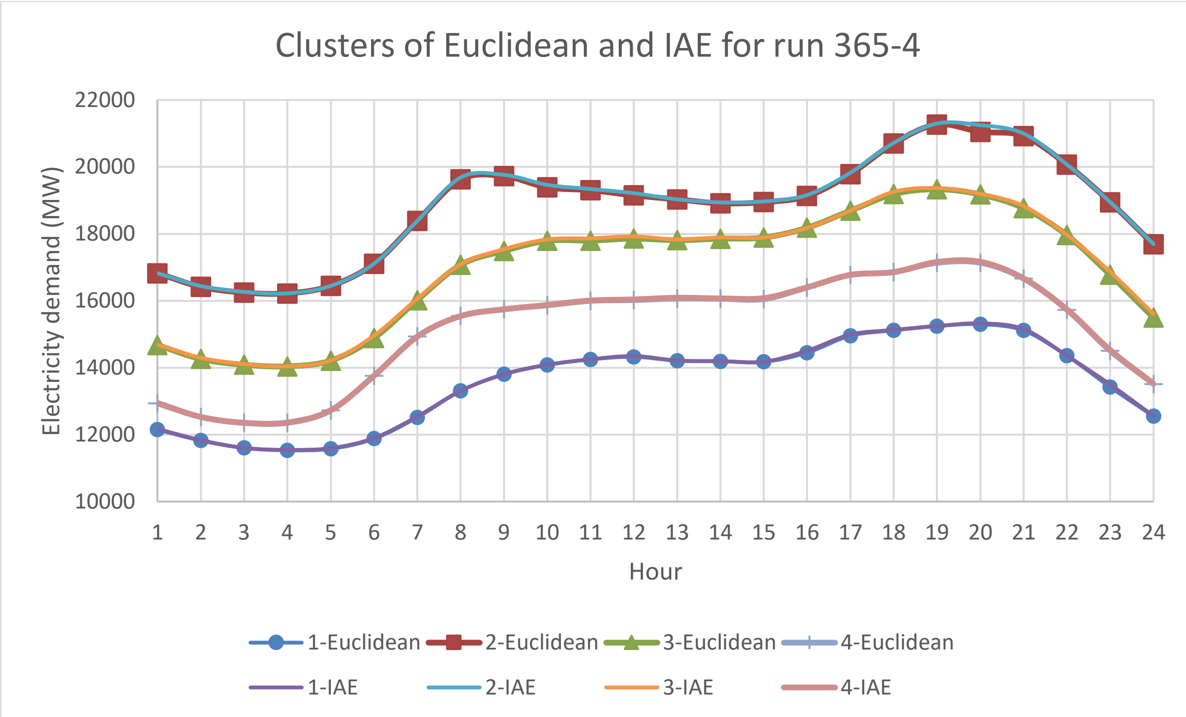

Figure 4-19: Clusters for the 365-4 case using IAE and Euclidean metrics

Table 4-6: Details of the obtained clusters using the L_1 (IAE) and L_2 (Euclidean) metrics for run 365-4

Hour	IAE clusters				Euclidean clusters			
	1	2	3	4	1	2	3	4
1	12,170	16,824	14,689	12,947	12,152	16,824	14,678	12,933
2	11,837	16,441	14,298	12,530	11,824	16,418	14,263	12,529
3	11,604	16,265	14,108	12,355	11,604	16,247	14,090	12,352
4	11,546	16,222	14,043	12,365	11,535	16,222	14,042	12,354
5	11,596	16,447	14,216	12,731	11,578	16,447	14,214	12,723
6	11,898	17,108	14,960	13,760	11,882	17,108	14,891	13,760
7	12,544	18,389	16,064	14,932	12,520	18,389	16,021	14,932
8	13,309	19,685	17,086	15,544	13,309	19,627	17,075	15,544
9	13,805	19,764	17,543	15,746	13,805	19,727	17,490	15,746
10	14,089	19,464	17,834	15,875	14,084	19,389	17,794	15,866
11	14,251	19,341	17,866	16,009	14,251	19,305	17,799	15,999
12	14,350	19,216	17,924	16,043	14,327	19,149	17,858	16,022
13	14,213	19,032	17,832	16,093	14,212	19,032	17,810	16,069
14	14,195	18,941	17,894	16,072	14,195	18,913	17,857	16,063
15	14,179	18,969	17,890	16,080	14,178	18,948	17,890	16,052
16	14,507	19,160	18,179	16,403	14,450	19,129	18,183	16,398
17	14,991	19,826	18,692	16,776	14,959	19,787	18,692	16,772
18	15,130	20,717	19,257	16,858	15,117	20,701	19,187	16,858
19	15,246	21,297	19,368	17,161	15,246	21,268	19,333	17,136
20	15,326	21,249	19,194	17,166	15,302	21,049	19,178	17,137
21	15,154	20,999	18,833	16,673	15,113	20,925	18,766	16,662
22	14,361	20,073	17,967	15,746	14,361	20,070	17,963	15,734
23	13,498	18,950	16,866	14,530	13,425	18,947	16,777	14,509
24	12,558	17,698	15,606	13,529	12,551	17,685	15,498	13,516

4.4 Conclusions

A mathematical programming based heuristic that is able to overcome the computational burden of the clustering model of chapter 3 was proposed in this chapter. The heuristics was developed by mimicking the basic procedure of *k*-means algorithm in the sense of starting with several initial guesses (or scenarios). However the algorithm is mathematical programming based and utilizes the proposed model of chapter 3. The heuristic of this chapter was first applied to cluster the 30 days electricity demands (of Case Study 1, **Chapter 3**) for validation purposes. The heuristic succeeded in finding the optimal solution for all the runs. However, the number of initial guesses leading to an optimal solution varied among the runs. The number of initial guesses leading to the optimal solution was higher for normal clustering than for sequence clustering. It took few minutes to solve the model for both normal and sequence clustering and hence the computational burden we significantly reduced as compared with the original model but without sacrificing any solution quality.

After this, the heuristics was applied to cluster electricity demands for an entire one year. The normal clustering and as was expected outperformed the sequence clustering in term of objective function, error average and standard deviation. This is so because of the additional restrictions that sequence clustering adds to the desired clusters. It was also found that an increase in the number of clusters decreases the objective function, which is in agreement with the findings in the previous chapter. In addition, the error average of normal clustering fluctuates as the number of clusters increases while the standard deviation declines. The same trend was observed in Case Study 1 of the previous chapter. The solution time for sequence clustering was shorter than that for normal clustering. However, the proposed heuristic showed its ability in reducing computational time and maintaining quality of solutions for this case of a whole year of operation.

The first 4 normal clusters did not change much as additional clusters were introduced in other runs, as observed for the first and last sequence clusters. These observations suggest that these clusters seem to be optimal and the algorithm is trying to find a new cluster to improve the objective function resulting in a lower relative error or standard deviation. The issue of an optimal number of clusters could be a future issue in the construction of the heuristic algorithm. In fact, the number of clusters can as well be set as an unknown variable along with the number of clusters and the assignment of days to clusters.

Chapter 5
Application of single attribute clustering to unit commitment model

5.1 Introduction

The main objective of this chapter is to investigate the impact on solution accuracy when clustered demand is applied to a planning model. More specifically, we would like to assess the outputs of normal and sequence clustering against a full planning model that does not employ clustering. The Unit Commitment (UC) model was chosen for this study since it is a well-established model. The UC problem is a medium term decision level problem whose objective is to minimize the operating cost of existing generator units while meeting electricity demands. There are several models for the UC problem available in the literature, ranging from heuristics to mathematical programming based techniques[115]. The unit commitment problem in this chapter is modeled as a Mixed Integer Linear Program (MILP)[116]. The following section presents the UC model formulation.

5.2 Unit commitment model formulation

Consider a set of *I* thermal units to be scheduled over a time horizon *T*. The goal is to minimize the operating cost while meeting the electricity demands and operating within the units' capacities. Eq. 5-1, which represents the objective (cost) function, shows the operating cost, including fuel consumption calculated by a linear function with fixed charges, and fixed start-up and shut-down costs.

$$COST = \sum_{i,t}(a_i * U_{i,t} + b_i * P_{i,t} + CU_{i,t} + CD_{i,t}) \qquad \textbf{(5-1)}$$

where a_i, b_i are coefficients of the fuel cost function of unit i, $U_{i,t}$ is a binary variable representing the on/off status of unit *i* at period t, $CD_{i,t}$ is the shut-down cost of unit *i* in period *t*, and $P_{i,t}$ is the power output of unit *i* in period *t*.

Eq. 5-2 and 5-3 below ensure that the power produced by unit *i* at time *t* is within the generation power limits of that unit.

$$P_{i,t} < P_i^U * U_{i,t} \qquad \forall\, t, i \qquad \textbf{(5-2)}$$

$$P_{i,t} > P_i^L * U_{i,t} \qquad \forall\, t, i \qquad \textbf{(5-3)}$$

Electricity demand D_t will be satisfied at any *t* time by Eq. 5-4 while Eq. 5-5 guarantees the spinning reserve required (R_t) by the available capacity of the active units.

$$\sum_i P_{i,t} \geq D_t \qquad \forall\, t, i \qquad \textbf{(5-4)}$$

$$\sum_i P_i^U U_{i,t} \geq D_t + R_t \qquad \forall\, t, i \qquad \textbf{(5-5)}$$

Eq. 5-6 and Eq. 5-7 specify the online-offline status of unit *i* in its earliest periods of operation which are determined by its initial status and its minimum up (TU_i) and down (TD_i) times. T_i^{ini} is the number of periods that unit *i* has been initially offline (T_i^{ini} < 0) or online (T_i^{ini} > 0).

$$U_{i,t} = 1 \qquad \forall\, i : T_i^{ini} > 0;\ \ t = 1, .., (TU_i - T_i^{ini}) \qquad \textbf{(5-6)}$$

$$U_{i,t} = 0 \qquad \forall\, i : T_i^{ini} < 0;\ \ t = 1, .., (TD_i + T_i^{ini}) \qquad \textbf{(5-7)}$$

Eq. 5-8 and Eq. 5-9 determine unit minimum-up time for the general case and for the first time period respectively. Similarly, Eq. 5-10 and Eq. 5-11 regulate the minimum down-time for a unit.

$$U_{i,t} - U_{i,t-1} \leq U_{i,t+j} \qquad \forall\, i, t > 1, j < TU_i; t + j \ \leq T \qquad \textbf{(5-8)}$$

$$U_{i,1} \leq U_{i,1+j} \qquad \forall\, i: T_i^{ini} < 0,\ j < TU_i \qquad \textbf{(5-9)}$$

$$U_{i,t+j} \leq U_{i,t} - U_{i,t-1} \qquad \forall\, i, t > 1, j < TD_i; t + j \ \leq T \qquad \textbf{(5-10)}$$

$$U_{i,1+j} \ \leq U_{i,1} \qquad \forall\, i: T_i^{ini} > 0,\ j < TD_i \qquad \textbf{(5-11)}$$

Eq. 5-12 and Eq. 5-13 ensure that the ramp rate limits are imposed over unit *i* at time period *t* only if the unit is online in that period and was online also in time period (*t* − 1).

$$P_{i,t} - P_{i,t-1} \leq RU_i U_{i,t-1} \qquad \forall\, i, t: t > 1 \qquad \textbf{(5-12)}$$

$$P_{i,t-1} - P_{i,t} \leq RD_i U_{i,t-1} \qquad \forall\, i, t: t > 1 \qquad \textbf{(5-13)}$$

The start-up cost function is defined as a hot start cost ($CU_{i,t} = Hsc_i$) if downtime ≤ ($TD_i + T_i^{COLD}$) and a cold start cost ($CU_{i,t} = Csc_i$) otherwise. This start-up cost function can be modeled by Eqs. 5-14 to 5-17. Eqs. 5-14 and 5-15 constrain the variable $CU_{i,t}$ to be greater or equal to the hot start cost Hsc_i if unit *i* is started-up at time period *t*. If instead, the unit *i* is turned on at the time period *t* and the downtime at that moment is greater than ($TD_i + T_i^{COLD}$), Eqs. 5-16 and 5-17 force $CU_{i,t}$ to be greater or equal than the cold start cost Csc_i.

$$\left(U_{i,t} - U_{i,t-1}\right) Hsc_i \leq CU_{i,t} \qquad \forall\, i, t: t > 1 \qquad \textbf{(5-14)}$$

$$U_{i,1} Hsc_i \leq CU_{i,t} \qquad \forall\, i : T_i^{ini} < 0 \qquad \textbf{(5-15)}$$

$$\left(U_{i,t} - \sum_{j \leq TD_i + T_i^{COLD}} U_{i,t-j}\right) Csc_i \leq CU_{i,t} \qquad \forall\, i, t: t > TD_i + T_i^{COLD} \qquad \textbf{(5-16)}$$

$$\left(U_{i,t} - \sum_{j<t} U_{i,t-j}\right) Csc_i \leq CU_{i,t}$$

$$\forall\, i : T_i^{ini} < 0, TD_i + T_i^{COLD} \geq \ t > TD_i + T_i^{COLD} + 1 \qquad \textbf{(5-17)}$$

Some units can incur a shut-down cost when they are turned off, which is modeled by Eqs. 5-18 and 5-19.

$$(U_{i,t-1} - U_{i,t})Dc_i \leq CD_{i,t} \qquad \forall\, i,t: t > 1 \qquad \textbf{(5-18)}$$

$$(1 - U_{i,1})Dc_i \leq CD_{i,1} \qquad \forall\, i : T_i^{ini} > 0 \qquad \textbf{(5-19)}$$

5.3 Methodology

The study was conducted on 10 thermal units (see **Appendix C** for the model parameters and details of the units and their capacities)[117]. The original operating cost function was linearized by regression in order to obtain linear functions and still maintain the structure of the model as an MILP (see the regressions of operation costs for all units in **Appendix C**). It is worth mentioning that R^2 value for the linearized cost function was almost 1 for all units within the operating limits. The electricity demand along with clustered demand from **Chapter 4** are used as inputs for this unit commitment model.

In order to examine the effect of model size, the number of units is doubled and tripled so that the number of thermal units in operations become 20 and 30, respectively. The electricity demand should match the units' capacities. Therefore, the electricity demand will be reduced to 5, 10 and 15 % for 10, 20 and 30 units respectively.

In addition, the cost function is multiplied by a parameter N_t as illustrated by Eq. 5-20 to allow comparison between the full scale model and the clustered cases. The parameter N_t represents the number of repetitions (frequency) for corresponding *t* time. The parameter N_t is equal to 1 for the full scale case and equal to the number of days in the clusters for the clustered case. For example, if cluster 1 represents 45 days, N_t will be equal to 45 for a 24 hour horizon of cluster 1. In this way, the cost function for the clustered model is as given by Eq. 5-20 below. The only difference compared to the original non-clustered objective function (Eq. 5-1) is the multiplication by the term N_t.

$$COST = \sum_{i,t}(N_t * (a_i * U_{i,t} + b_i * P_{i,t} + CU_{i,t} + CD_{i,t})) \qquad \textbf{(5-20)}$$

The full scale model has a time horizon of 8760 hours while the clustered cases will have 96, 120, 144 and 168 hours for 4, 5, 6 and 7 clusters, respectively.

5.4 Results and discussion

GAMS/CPLEX [110] was used to conduct this study on an Inter(R) Xeon(R) 2.4 GHz (2 processors), 16 GB RAM workstation. Table 5-1 and Table 5-2 present result summaries of normal and sequence clustering for various number of units, respectively. The application of clustering (both normal and sequence) shows a clear advantage in terms of solution time compared to the full scale case. The solution times of normal and sequence cases for 10 and 20 units are very close to each other. However, it takes far less time to solve the model for the normal clustering compared to sequence one for the 30 units' case. More details about the model output for the different runs conducted are given in **Appendix C**.

The objective functions for the clustered cases is very close to the optimal non clustered model. Figure 5-1 to Figure 5-3 show the objective function values in error percentage (in comparison with the optimal non clustered solution) of 10, 20, and 30 units respectively for all normal and sequence clusters. The error range is within ± 0.5 % for all cases. Consistent trends are observed across these figures. The first observation is that normal clustering is underestimated while sequence clustering is overestimated. The 4 sequence clusters case is the closet to the optimal indicating that four clusters is the optimal representation of the demand curve for electricity. This is in accordance with what is expected since usually electricity demand is seasonal and is often clustered into the four well known seasons. Therefore this validates further the clustering algorithm presented in this work. The clustering algorithm uses the data to come up with the clusters and it probably can be improved further through the incorporation of a priori knowledge about the demand data and the optimal number of clusters that should be employed. Using more than four clusters seems to increase the error gap as illustrated. This suggests that increasing the number of clusters does not improve the solution quality for the case of electricity data. It also indicates that there is an optimal number of normal or sequence clusters regardless of cluster quality.

Table 5-1: Results summary of normal clustering for various number of units

Number of units		Optimal	Number of clusters (Normal)			
			4	5	6	7
10	CPU time (s)	2,228	3	5	5	6
	Objective function ($)	1.37 x10^8	1.37 x10^8	1.37 x10^8	1.37 x10^8	1.37 x10^8
20	CPU time (s)	33,580	9	14	12	22
	Objective function ($)	2.73 x10^8	2.72 x10^8	2.73 x10^8	2.73 x10^8	2.73 x10^8
30	CPU time (s)	99,280	63	28	129	37
	Objective function ($)	4.09 x10^8	4.08 x10^8	4.08 x10^8	4.08 x10^8	4.08 x10^8

Table 5-2: Results summary of sequence clustering for various number of units

Number of units		Optimal	Number of clusters (Sequence)			
			4	5	6	7
10	CPU time (s)	2,228	3	5	6	9
	Objective function ($)	1.37x10^8	1.37 x10^8	1.37 x10^8	1.38 x10^8	1.38 x10^8
20	CPU time (s)	33,580	11	13	15	26
	Objective function ($)	2.73 x10^8	2.73 x10^8	2.74 x10^8	2.74 x10^8	2.74 x10^8
30	CPU time (s)	99,280	177	1830	1720	979
	Objective function ($)	4.09 x10^8	4.09 x10^8	4.10 x10^8	4.11 x10^8	4.11 x10^8

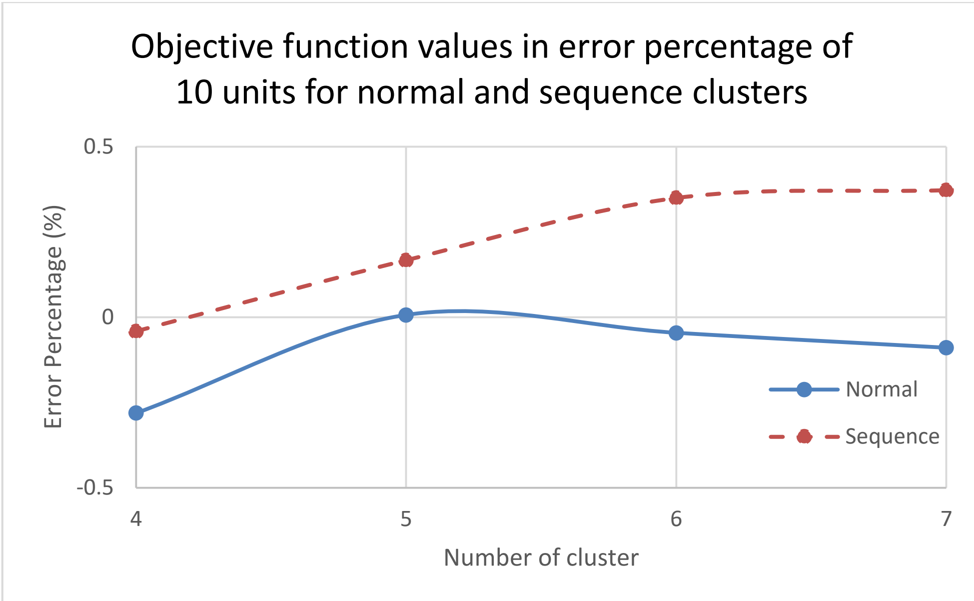

Figure 5-1: Objective function values in error percentage of 10 units for normal and sequence clusters

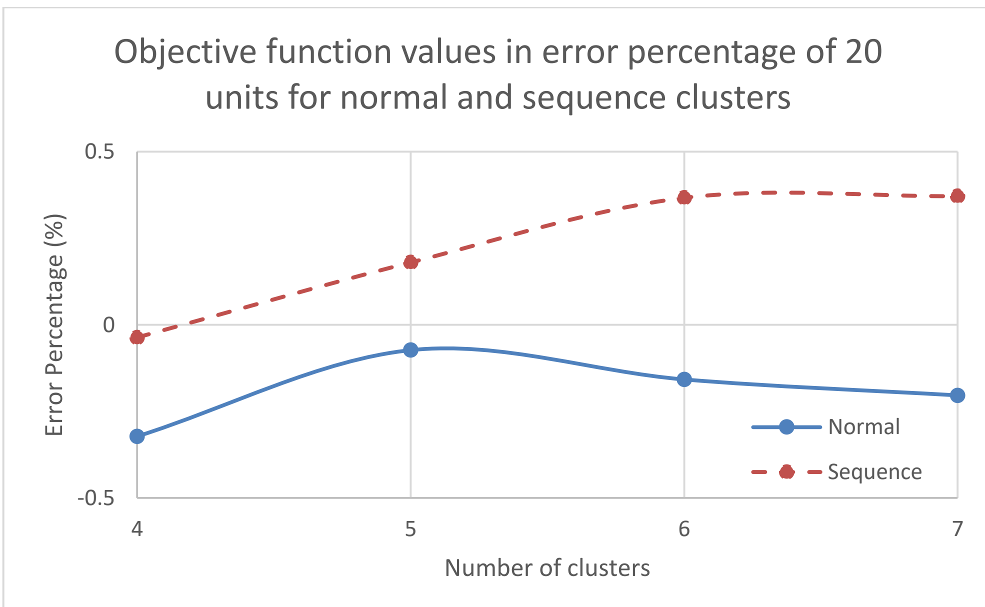

Figure 5-2: Objective function values in error percentage of 20 units for normal and sequence clusters

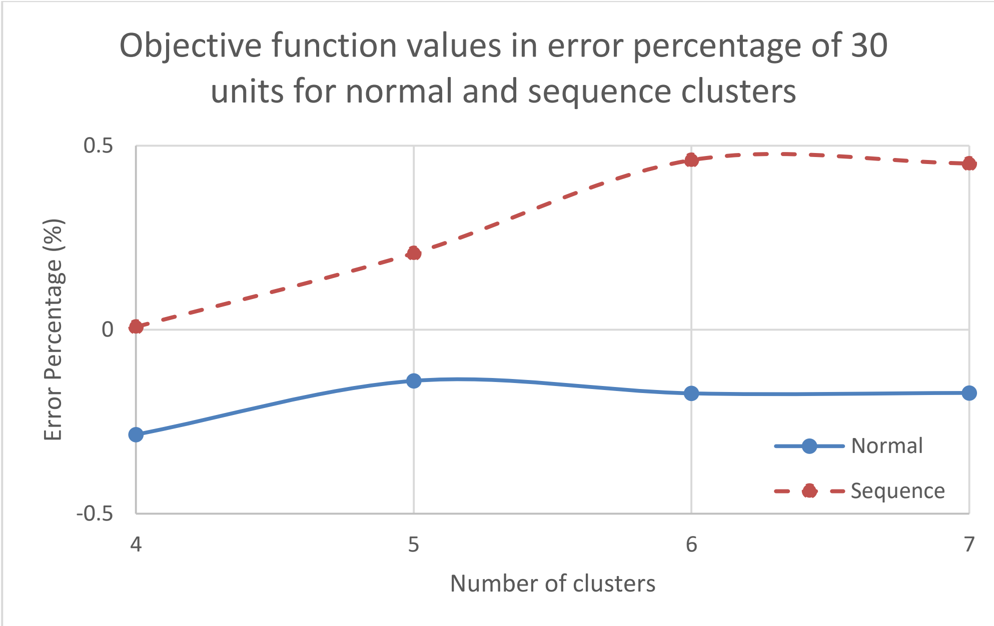

Figure 5-3: Objective function values in error percentage of 30 units for normal and sequence clusters

The parameter N_t is multiplied by Variable $P_{i,t}$ and $U_{i,t}$ once the solution is obtained. Similar to the objective function, this allows comparison between the full space model and the clustered cases. It determines the online number of hours and produced power of each unit. Table 5-3 and Table 5-4 show the produced power of normal and sequence clustering respectively for 10 units along with their optimal values (see **Appendix C** for more details). The results of clustered cases are very close to the optimal case. Normal clustering predicts the units responsible to meet the demands for the optimal solution. Similarly, Tables 5-5 and 5-6 show the online number of hours for normal and sequence clustering respectively for 10 units along with optimal non-clustered values.

Table 5-3: Produced power (MWh) of optimal and normal clustering for 10 units

		Normal clusters			
Units	**Optimal**	**4**	**5**	**6**	**7**
1	3.98×10^6	3.97×10^6 (-0.13%)	3.97×10^6 (-0.09%)	3.97×10^6 (-0.08%)	3.97×10^6 (-0.12%)
2	2.86×10^6	2.85×10^6 (-0.31%)	2.85×10^6 (-0.40%)	2.85×10^6 (-0.54%)	2.86×10^6 (-0.22%)
4	3.75×10^4	5.76×10^4 (54%)	5.36×10^4 (43%)	4.35×10^4 (16%)	4.35×10^4 (16%)
5	8.55×10^4	5.27×10^4 (-38%)	7.88×10^4 (-7.9%)	8.88×10^4 (3.9%)	7.63×10^4 (-11%)
6	2.65×10^4	3.81×10^4 (44%)	2.62×10^4 (-0.9%)	2.62×10^4 (-1.2%)	3.26×10^4 (23%)
8	1.09×10^3	4.10×10^2 (-62%)	3.50×10^2 (-68%)	0	8.60×10^2 (-21%)

Table 5-4: Produced power (MWh) of optimal and sequence clustering for 10 units

		Sequence clusters			
Units	**Optimal**	**4**	**5**	**6**	**7**
1	3.98×10^6	3.97×10^6 (-0.08%)	3.98×10^6 (0.02%)	3.98×10^6 (0.11%)	3.98×10^6 (0.10%)
2	2.86×10^6	2.91×10^6 (1.56%)	2.92×10^6 (1.94%)	2.92×10^6 (2.12%)	2.92×10^6 (2.16%)
4	3.75×10^4	0	0	0	0
5	8.55×10^4	6.92×10^4 (-19%)	7.38×10^4 (-14%)	6.98×10^4 (-18%)	6.98×10^4 (-18%)
6	2.65×10^4	4.66×10^4 (76%)	3.69×10^4 (39%)	4.66×10^4 (76%)	4.71×10^4 (78%)
8	1.09×10^3	0	7.20×10^2 (-34%)	9.40×10^2 (-14%)	9.40×10^2 (-14%)

Table 5-5: Online number of hours for optimal and normal clustering for 10 units

		Normal clusters			
Units	**Optimal**	**4**	**5**	**6**	**7**
1	8,760	8,760 (0%)	8,760 (0%)	8,760 (0%)	8,760 (0%)
2	8,760	8,760 (0%)	8,760 (0%)	8,760 (0%)	8,760 (0%)
4	289	451 (56.1%)	420 (45.3%)	341 (18.0%)	341 (18.0%)
5	2,459	1,470 (-40.2%)	2,549 (3.7%)	2,673 (8.7%)	2,176 (-11.5%)
6	1,303	1,906 (46.3%)	1,312 (0.7%)	1,301 (-0.2%)	1,621 (24.4%)
8	109	41 (-62.4%)	35 (-67.9%)	0	86 (-21.1%)

Table 5-6: Online number of hours for optimal and sequence clustering for 10 units

		Sequence clusters			
Units	**Optimal**	**4**	**5**	**6**	**7**
1	8,760	8,760 (0%)	8,760 (0%)	8,760 (0%)	8,760 (0%)
2	8,760	8,760 (0%)	8,760 (0%)	8,760 (0%)	8,760 (0%)
4	289	0	0	0	0
5	2,459	1,652 (-32.8%)	2,241 (-8.9%)	1,723 (-29.9%)	1,723 (-29.9%)
6	1,303	2,330 (78.8%)	1,845 (41.6%)	2,328 (78.7%)	2,356 (80.8%)
8	109	0	72 (-33.9%)	94 (-13.8%)	94 (-13.8%)

The sequence of time horizon for normal clustering is a very important factor as it affects the solution time and quality. For example, there are 720 (6!) ways to arrange the time horizon of 6 normal clusters by switching between cluster loads. The ramp rate might eliminate some combinations because of infeasibilities. Figure 5-4 shows 3 possible combinations for time horizon of the 6 normal clusters case for illustration purposes. All of them comply with ramp rate requirements. The number of units is set to 30 to examine the impact. Table 5-7 presents the computational summary. As can been noticed, the sequence has a major impact on the solution time. The effect on solution quality is insignificant as the results suggest.

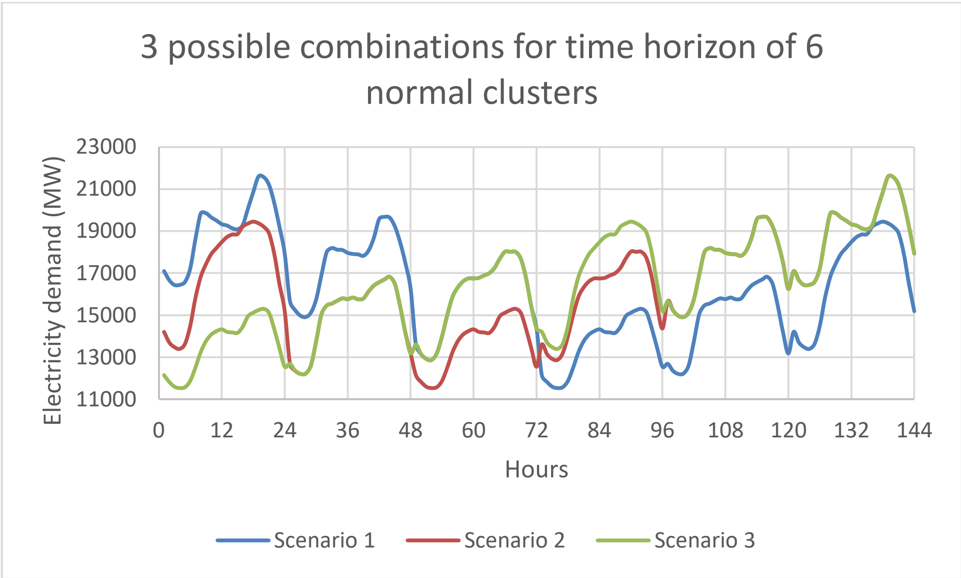

Figure 5-4: Three possible combinations for time horizon of 6 normal clusters

Table 5-7: Computational summary of 3 possible time horizon of 6 normal clusters using 30 units

	Scenario 1	Scenario 2	Scenario 3
Solution time (s)	164	1,127	54,652
Objective function ($)	4.08 $\times10^8$	4.08 $\times10^8$	4.09 $\times10^8$
Error (%)	-0.173	-0.130	-0.022

5.5 Conclusion

This chapter has evaluated the solution quality when clustered demand is applied to a planning model. Unit commitment model was chosen as the planning model for this study. The clustered electricity demand from the previous chapter was used as input to the model. The results show a great advantage in terms of solution time for clustered cases compared the original optimal solution (full scale case). Furthermore, increasing model size has a minor impact on the solution error. In addition, the results suggest high solution quality can be achieved with a smaller number of clusters.

The objective function values for clustered cases are very close to the original optimal (full scale case) solution. The error range is within ± 0.5 % for all cases considered. Normal clustering produced solutions that slightly underestimate the optimal non-clustered solution while sequence clustering produced solutions that overestimate the original optimal solution but this is case specific and the trend and occurrence cannot be generalized. The 4 sequence clusters case is the closest to the original optimal solution while the 5 clusters case is the closest to for normal clustering. The error gap increases as the number of clusters increases, thus suggesting that increasing the number of clusters does not improve solution quality. It also indicates that there is an optimal number of normal or sequence clusters regardless of cluster quality.

As indicated, the sequence of time horizon for normal clustering is very important due to its effects on the solution time and quality. For illustration purposes, we showed 3 possible combinations for time horizon of the 6 normal clusters case. All of them comply with ramp rate requirements. The number of units was set to 30 to examine the impact. As has been noticed, the sequence has a major impact on the solution time. The effect on solution quality is insignificant as the results suggest. The sequence of time horizon for normal clustering can be challenging as there are many ways to obtain arrangements.

Chapter 6
Shape-based Time Series Mathematical Programming Clustering Model for Multiple Attributes

6.1 Introduction

The input parameters in process systems engineering usually consist of multiple attributes such as the simultaneous demand for electricity and heat. Therefore, the aim of this chapter is to extend the clustering model formulated in **Chapter 3** to take into account multiple attributes at the same time. The weighting method was chosen as a multi-objective optimization approach[118]. The following shows the typical model formulation for multi-objective optimization using weighting method

$$\min z = \sum_a w_a * f_a(x) \qquad \textbf{(6-1)}$$

$$\text{s.t.} \qquad x \in S \qquad \textbf{(6-2)}$$

where f_a is the objective function for attribute a, w_a is the weight factor for attribute a, $w_a \geq 0$, $\sum_a w_a = 1$, and S is the feasible region. Figure 6-1 shows an illustration of a Pareto front for a bi-objective problem. μ_1 and μ_2 are values of objective functions 1 and 2 respectively. The Pareto frontier is constructed by applying different combinations of weight factors. The utopia point (μ^u) corresponds to the optimal values of objective functions 1 and 2 (μ^{1*} and μ^{2*}). However, the utopia point is usually infeasible as illustrated by the figure. Therefore, the best solution is the closest to the utopia point.

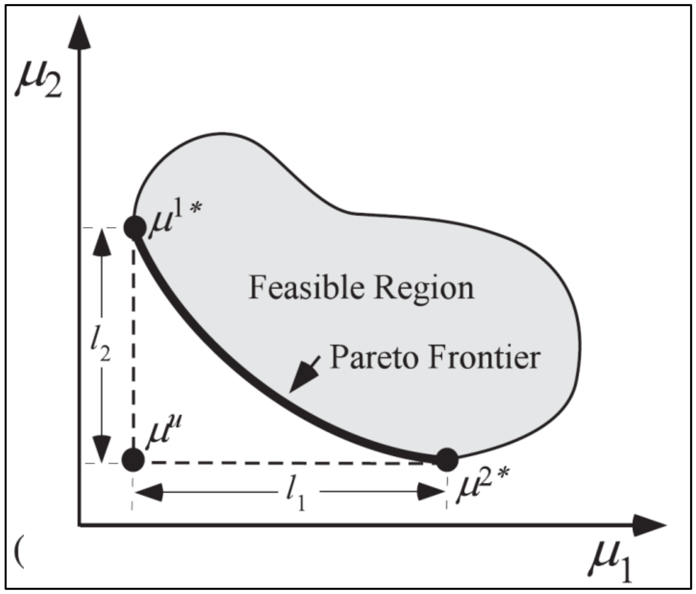

Figure 6-1: Illustration for Pareto frontier [119]

6.2 Model formulation

The formulation below presents the clustering model for multiple attributes. The differences between this formulation and the one in **Chapter 3** are the introduction of index a (that represents the different attributes) and the application of the weighting method to handle the multi-objective issue. Therefore, the model for normal clustering will be Eqs.6-3 to 6-11 while for sequence clustering, it will be Eqs.6-3 to 6-12.

$$\min z = \sum_a w_a * IAE_a \qquad \textbf{(6-3)}$$

s.t. $$\sum_{c=1}^{C} x_{d,c} = 1 \qquad \forall\ d \qquad \textbf{(6-4)}$$

$$IAE_a = \frac{\Delta}{2} * \sum_{d=1}^{D} \sum_{h=1}^{H-1} I_{a,d,h} + I_{a,d,h+1} \qquad \forall\ a \qquad \textbf{(6-5)}$$

$$I_{a,d,h} \geq L_{a,d,h} * x_{d,c} - P_{a,h,d,c} \qquad \forall\ a,h,d,c \qquad \textbf{(6-6)}$$

$$I_{a,d,h} \geq P_{a,h,d,c} - L_{a,d,h} * x_{d,c} \qquad \forall\ a,h,d,c \qquad \textbf{(6-7)}$$

$$Cl_{a,c,h} - L_{a,h}^{U} * \left(1 - x_{d,c}\right) \leq P_{a,h,d,c} \qquad \forall\ a,h,d,c \qquad \textbf{(6-8)}$$

$$L_{a,h}^{L} * x_{d,c} \leq P_{a,h,d,c} \qquad \forall\ a,h,d,c \qquad \textbf{(6-9)}$$

$$Cl_{a,c,h} - L_{a,h}^{L} * \left(1 - x_{d,c}\right) \geq P_{a,h,d,c} \qquad \forall\ a,h,d,c \qquad \textbf{(6-10)}$$

$$L_{a,h}^{U} * x_{d,c} \geq P_{a,h,d,c} \qquad \forall\ a,h,d,c \qquad \textbf{(6-11)}$$

$$x_{d+1,c} \leq x_{d,c} + x_{d,c-1} \qquad \forall\ d\ ,c \qquad \textbf{(6-12)}$$

The following defines variables and parameters of the model:

$I_{a,d,h}$ = Absolute difference between load curve L and clustered curve C for hour h in day d day for attribute a

$Cl_{a,c,h}$ = Demand for hour h in cluster c and attribute a

$P_{a,h,d,c}$ = Relaxation variable

$x_{d,c}$ = Assignment variable of load for day d joining cluster c and it is equal to one if such assignment takes place and equal to 0 otherwise.

$L_{a,d,h}$ = Demand load of attribute a for h hour in day d

$L^L_{a,h}$ = Lower bound of attribute a load for hour h

$L^U_{a,h}$ = Upper bound of attribute a load for hour h

Eq. 6-3 above represents the objective function as a weighted function between the performance criteria of the different attributes a under consideration. $w_a$in the equation represents the weight factors for the attributes a with the additional restrictions that: $w_a \geq 0$, and $\sum_a w_a = 1$. Eq. 6-4 is the day assignment constraint that requires that each day of the year is assigned to a cluster C of curves. Eq. 6-5 is a numerical evaluation of the norm L_1 using the trapezoidal rule for IAE between loads L and cluster curves C. As mentioned in **Chapter 3** other quadrature integration schemes could also be employed by accounting for the number of segments in an appropriate manner. For instance, for an odd number of segments, one can apply the trapezoidal rule for the first segment and Simpson's 1/3 rule for the remaining segments. However, we chose to employ the trapezoidal rule for illustration purposes and for the sake of simplicity. The intervals considered are small enough (in hours) compared to the overall horizon (year) and the approximation of IAE using the trapezoidal rule will be adequate. As indicated in **Chapter 3**, the model construction is flexible in terms of which performance criteria is used. Utilizing the L_2 norm instead is straightforward and requires the use of the Euclidean distance in Eq. 6-5. Eq. 6-6 to 6-10 are analogous to Eqs 3-11 to 3-16 and are obtained from the linearization of bilinear terms in the original model but now an additional index that takes into account attribute a is included. Similarly, Eq. 6-12 is analogous to Eq. 3-8 and is needed only for the case of sequence clustering.

The above model is a mathematical representation of clustering trajectories of time series data of different attributes and is aimed to achieve clusters through the minimization of the L_1 norm. The model can be used for normal clustering (Eqs. 6-3 to 6-11) or sequence clustering (Eqs. 6-3 to 6-12). This formulation provides a unique platform to provide normal and sequence clustering for problems with multiple attributes since it utilizes the same algorithmic structure.

6.3 Proposed heuristics for multiple attributes

Since the computational complexity is evident for the clustering model of the previous section, we will extend the proposed heuristic algorithm of **Chapter 4** to handle multiple attributes. Figure 6-2 shows a flowchart of the proposed heuristic algorithm for multiple attributes. Here the heuristics will be executed for every weight factor combination. The following explains the procedures. First step of the proposed heuristic starts with generating *n* random clusters or scenarios (Figure 6-2). The scenarios can be generated in Excel (similar to what we did in **Chapter 4**) by randomizing between maximum and minimum of each hour and for each attribute for the entire demand curves. The procedure then considers each weight factor and starts with the first scenario. At a first attempt the clusters are fixed in the MILP model and the resulting integer program is solved for day assignment. This gives an upper bound on the solution. Next the day assignment is fixed and an LP model is solved to get a lower bound solution. If the difference between the lower and upper bounds is within an acceptable prespecified range then the solution is saved as current best solution and the next scenario is considered. Otherwise, the procedure for a given scenario is repeated between fixing clusters and then fixing day assignment until the lower and upper bounds are within the acceptable tolerance. When all scenarios for a given weight factor are considered, the procedure goes to the next weight factor and the steps are repeated for until all weight factors are considered.

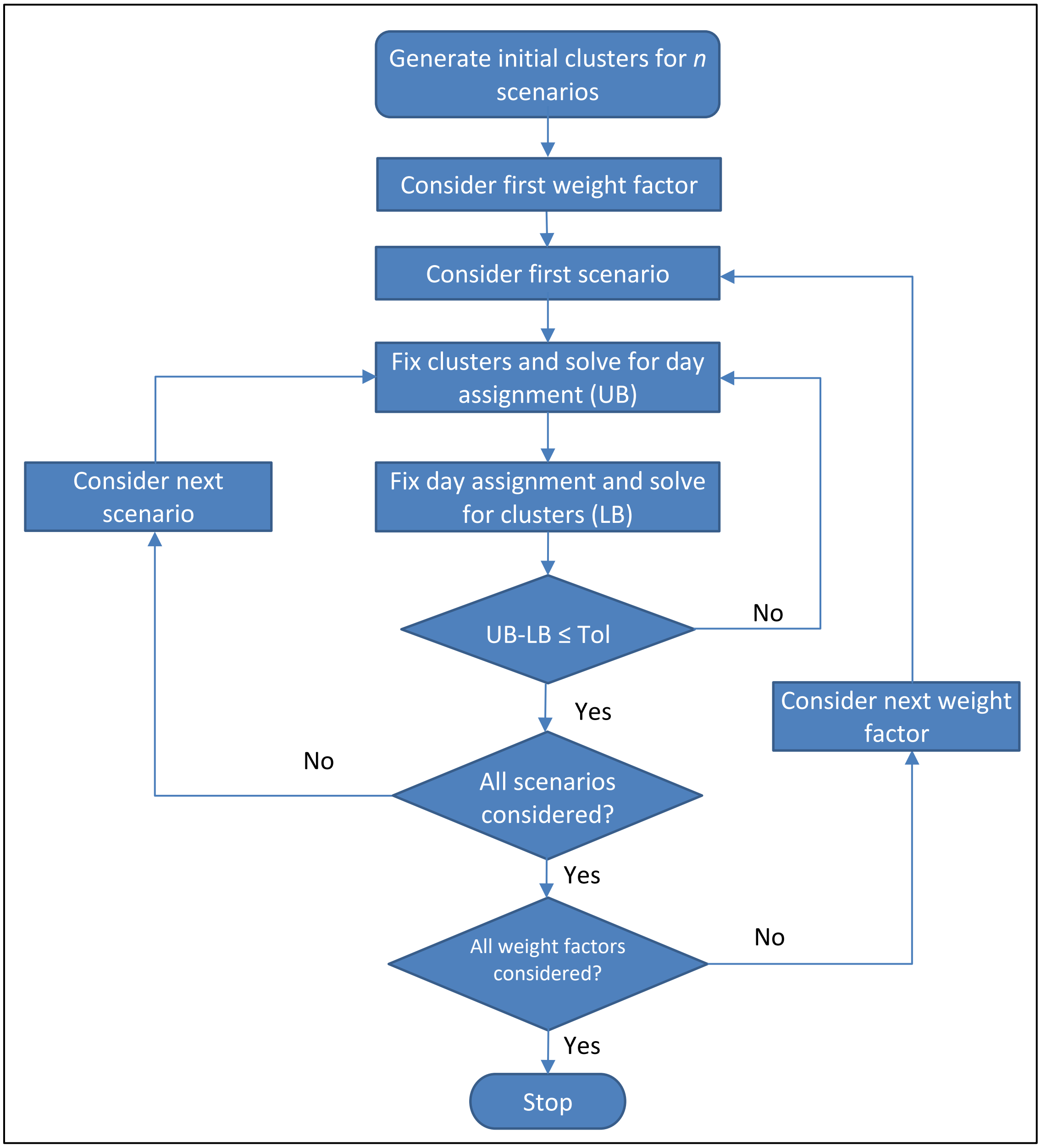

Figure 6-2: Flowchart of proposed heuristics for multiple attributes

6.4 Case Studies

The aim of this section is to assess the computational performance and outputs of the multiple attribute clustering algorithm derived in the previous section. Similar to **Chapter 4**, the initial guess was generated in Excel by randomizing between maximum and minimum of each hour and attribute for the entire demand. The same days to clusters ratios, described in **Chapter 4**, were used to generate the initial guess for sequence clustering. Runs for this case study are 4, 5, and 6 clusters with 365 days for each normal and sequence clustering, so runs are 6 in total. Twenty five scenarios were generated per run. It is worth mentioning that parameter tuning was used for sequence clustering to reduce solution time. The algorithm tolerance was set to 10^{-4}.

Hourly heat and electricity demands for an energy hub system will be used in this study [120]. Figure 6-3 and Figure 6-4 show the demands for heat and electricity respectively. GAMS/CPLEX [110] was used to conduct the runs on Inter(R) Xeon(R) 2.4 GHz (2 processors), 16 GB RAM workstation. Table 6-1 shows the 8 weight factor combinations to determine the Pareto front. The priority between heat and electricity varies among the weight factor combinations. Weight factor 1 leans towards heat whereas for electricity, it is weight factor 8.

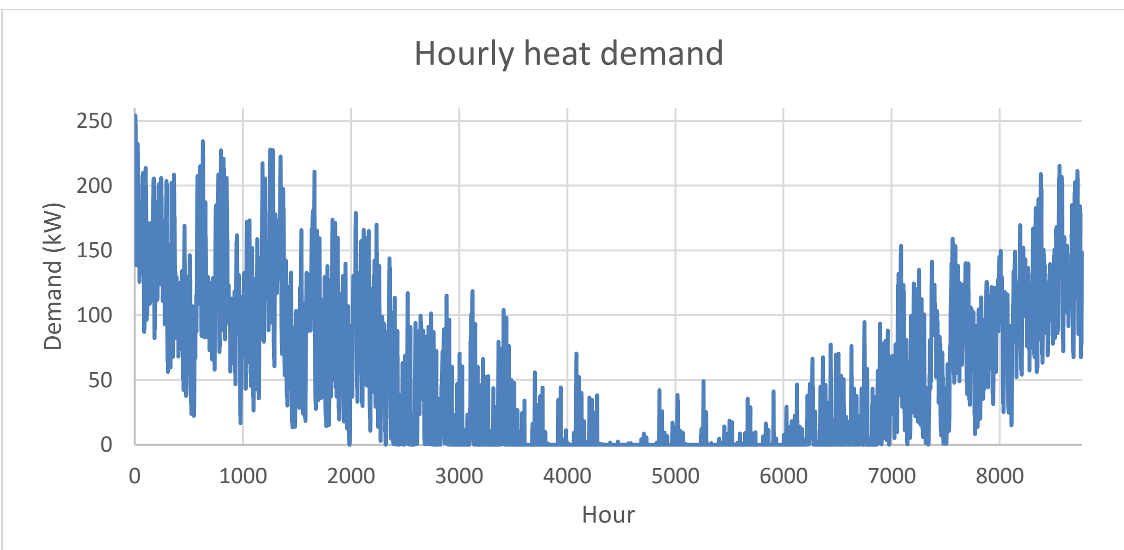

Figure 6-3: Hourly heat demand

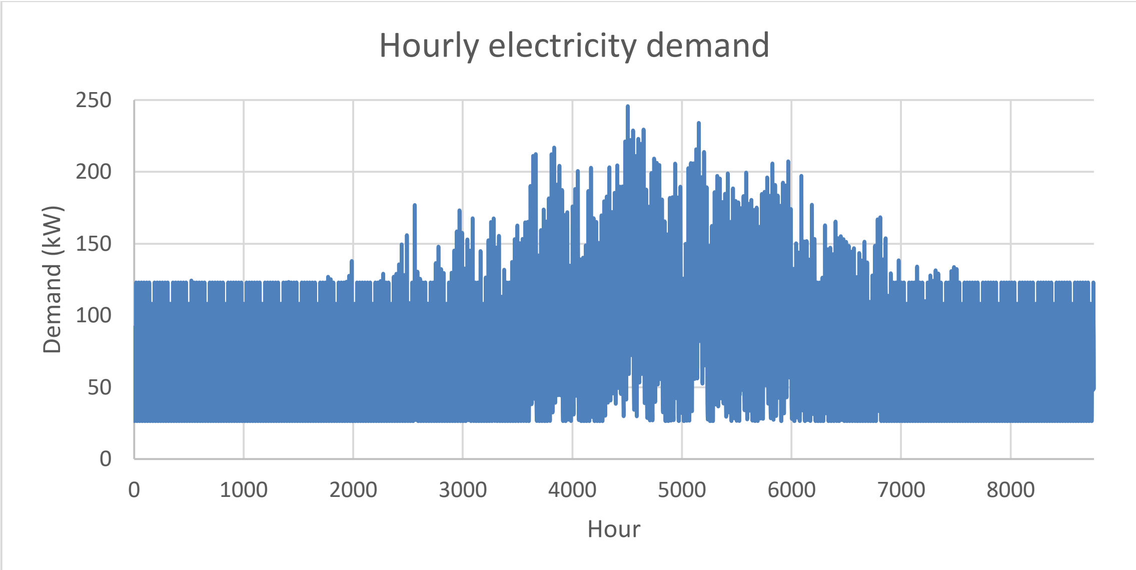

Figure 6-4: Hourly electricity demand

Table 6-1: Weight factors

	Electricity	Heat
1	0.2	0.8
2	0.3	0.7
3	0.4	0.6
4	0.5	0.5
5	0.6	0.4
6	0.7	0.3
7	0.8	0.2
8	0.9	0.1

Table 6-2 shows the solution time for all runs. The solution time for sequence clustering is shorter than for normal clustering even if they are in the same order of magnitude. Figure 6-5 and Figure 6-6 present the Pareto frontiers for normal and sequence clustering, respectively. As one can notice, the Pareto frontiers are captured for all runs with the weight factor combinations considered in this study. A better objective function is achieved when the number of clusters increases for both normal and sequence clustering.

Table 6-2: Solution time for all runs

	Normal clustering			Sequence clustering		
	4	5	6	4	5	6
Average solution time per scenario (min)	5.90	6.35	10.63	1.71	2.83	7.74

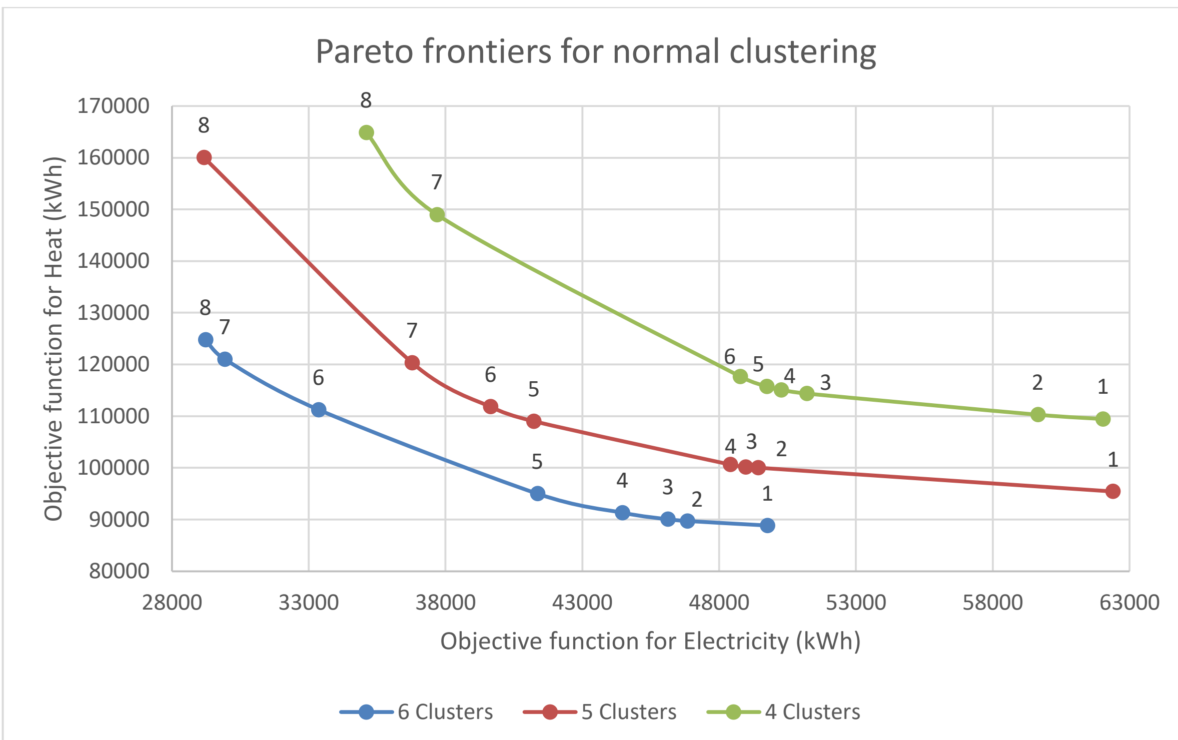

Figure 6-5: Pareto frontiers for normal clustering

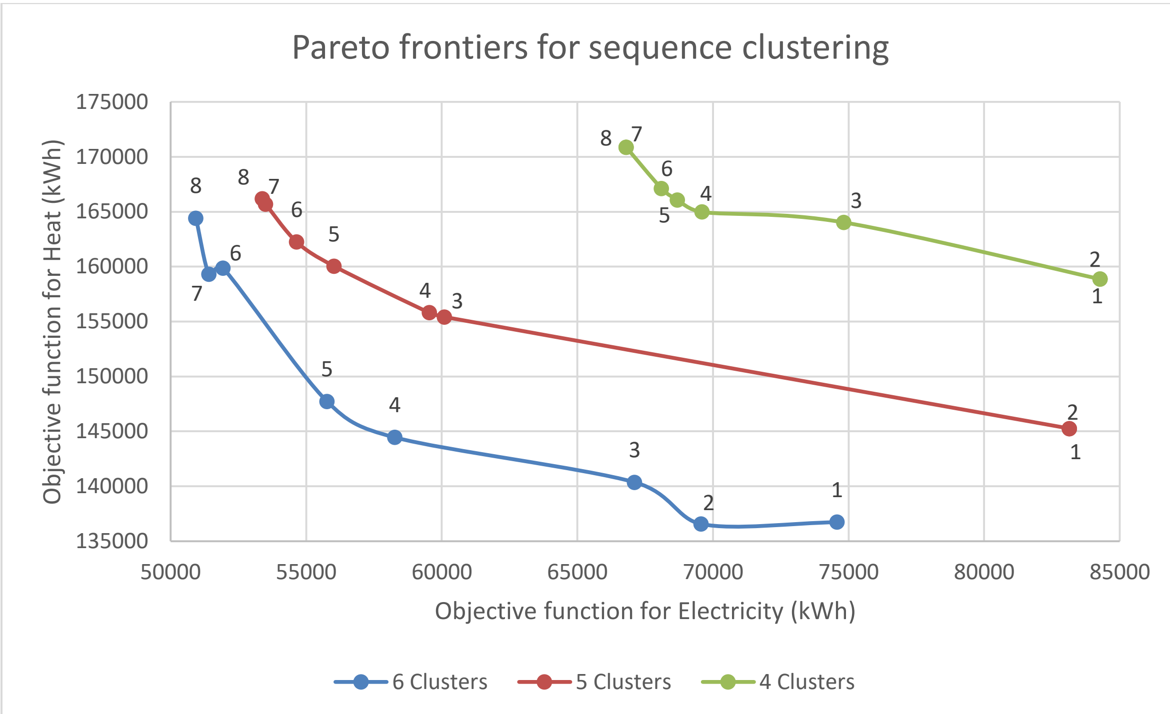

Figure 6-6: Pareto frontiers for sequence clustering

Table 6-3 and Table 6-4 show the results of 5 clusters for normal and sequence clustering respectively (see **Appendix D** for the results of all runs). The same relative error presented in the previous chapters was applied to carry out the average and standard deviation calculations. The results for normal clustering were better than those for sequence clustering in terms of objective function, error average and standard deviation. This is expected as discussed in previous chapters since sequence clustering imposes the extra sequence requirements that might be needed in certain process operations to minimize set-ups. As can be seen, the results vary as the weight factor changes. The heat demand contains zero values for certain periods. For these particular instances, the relative error calculation is not conducted since we cannot divide by zero. However, the error average and standard deviation of the heat demand are amplified. This comes from the huge fluctuation in the heat demand. Although the demand range is between 0 and 250 kW, the calculation of relative error is still difficult. For example, if the demand value is 0.1 kW and the cluster value is 1 kW, the relative error will be 900 %. The error average and standard deviation for the electricity demand are not that severe compared to the heat. In addition, they are in the same order of magnitude as those discussed in **Chapter 4**.

Table 6-3: Computational statistics Normal 365-5

Weight	Electricity			Heat		
	Average (%)	Std (%)	IAE (MWh)	Average (%)	Std (%)	IAE (MWh)
1	1.07	18.30	62.4	549	17,766	95
2	1.12	15.67	49.4	426	12,935	100
3	1.11	15.57	49.0	426	13,099	100
4	1.07	15.44	48.4	429	13,100	101
5	0.62	13.36	41.2	533	16,688	109
6	0.45	12.73	39.6	537	15,893	112
7	0.26	11.51	36.8	726	20,728	120
8	-0.08	9.40	29.2	1,013	25,552	160

Table 6-4: Computational statistics Sequence 365-5

Weight	Electricity			Heat		
	Average (%)	Std (%)	IAE (kWh)	Average (%)	Std (%)	IAE (kWh)
1	-0.55	20.65	83.2	759	24,238	145
2	-0.55	20.65	83.2	759	24,238	145
3	0.63	17.73	60.1	982	28,268	155
4	0.64	17.52	59.5	963	27,832	156
5	0.32	16.45	56.0	1,055	34,650	160
6	0.36	15.87	54.7	929	28,813	162
7	0.27	15.49	53.5	928	28,768	166
8	0.28	15.45	53.4	1,040	32,355	166

Figure 6-7 to Figure 6-12 show the clusters and day assignments of normal and sequence clustering for weight factors 1 and 8 and for 4, 5, and 6 clusters. The priority of weight factor 1 is for the heat demand whereas it is weight factor 8 for the electricity demand. The weight factor has a major effect on clusters.

The cluster quality is enhanced as the number of clusters increases. The flexibility of normal clustering has a major advantage over the sequence. There are many clusters of electricity demand, especially sequence clusters, overlapping with each other. They cannot be merged since they correspond to different days and the clusters of heat demand for these days are different.

6.5 Conclusion

This chapter proposed an extension of the clustering model, developed in the **Chapter 3,** to incorporate multiple attributes simultaneously. Different attributes have different scales or units, which renders the problem a multi-objective optimization problem. The weighting method approach was applied to deal with such a problem. In addition, the heuristic algorithm, used in **Chapter 4**, was extended to consider all weight factors combinations.

Hourly electricity and heat demands for one year were used in this study. The Pareto frontiers were captured for all runs with the weight factor combinations considered. The results show that a better objective function is achieved when the number of clusters increases for both normal and sequence clustering. The normal clustering results are found to be better the sequence clustering in terms of objective function, error average and standard deviation. The statistical analysis of the heat demand was challenging as suggested by the results. This is due to the huge fluctuation in the heat demand. Moreover, calculations of relative error were troublesome for the demand that was close zero.

The flexibility of normal clustering has a major advantage over the sequence. There are many clusters of electricity demand, especially sequence clusters, overlapping with each other. They cannot be merged since they correspond to different days and the clusters of heat demand for these days are different. Therefore, for applications that do not require sequencing it is advantageous to use normal clustering to minimize the computational effort and be able to deal with large scale models.

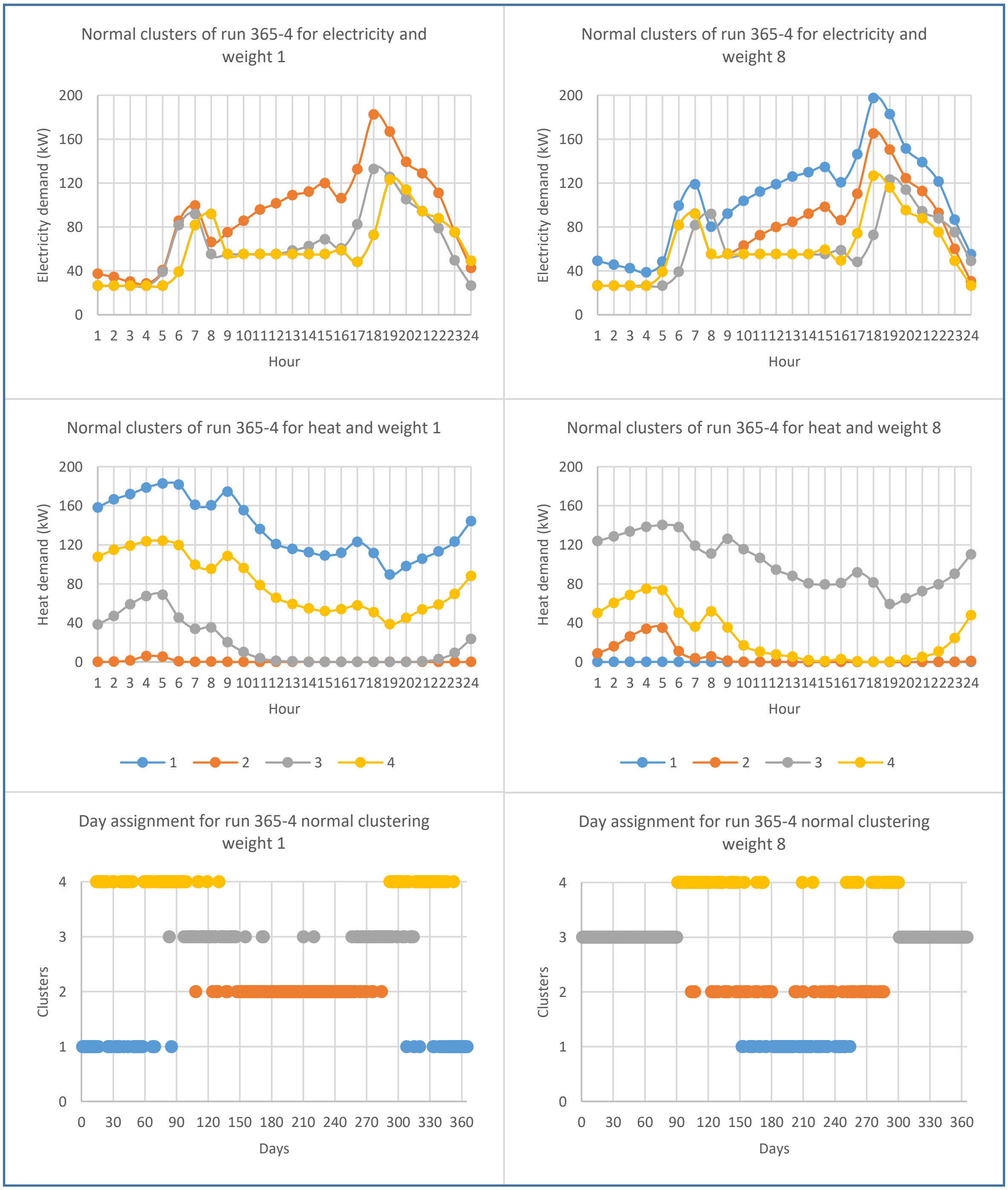

Figure 6-7: Normal clusters of run 365-4 for heat and electricity with day assignment for weight 1 and 8

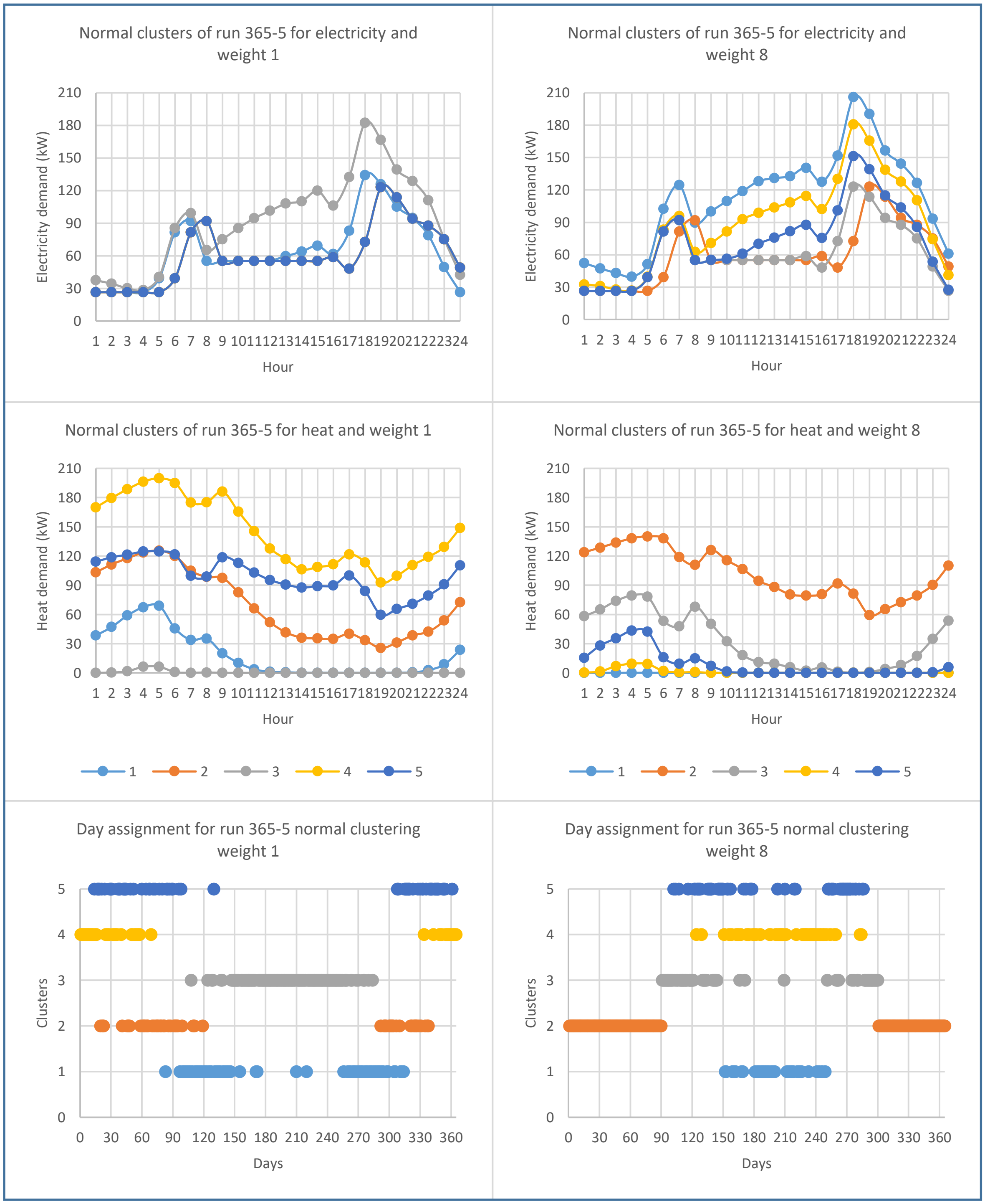

Figure 6-8: Normal clusters of run 365-5 for heat and electricity with day assignment for weight 1 and 8

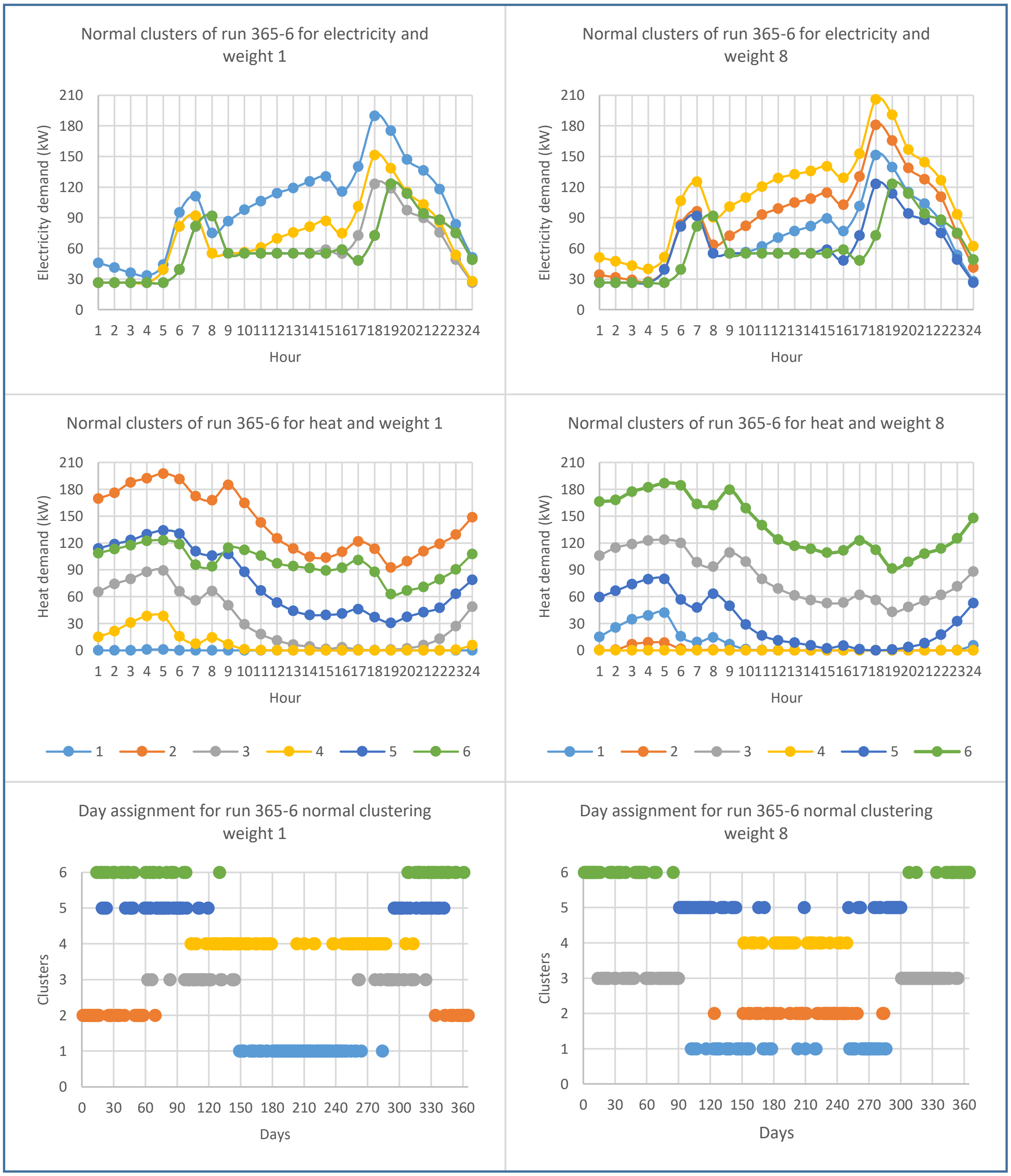

Figure 6-9: Normal clusters of run 365-6 for heat and electricity with day assignment for weight 1 and 8

Figure 6-10: Sequence clusters of run 365-4 for heat and electricity with day assignment for weight 1 and 8

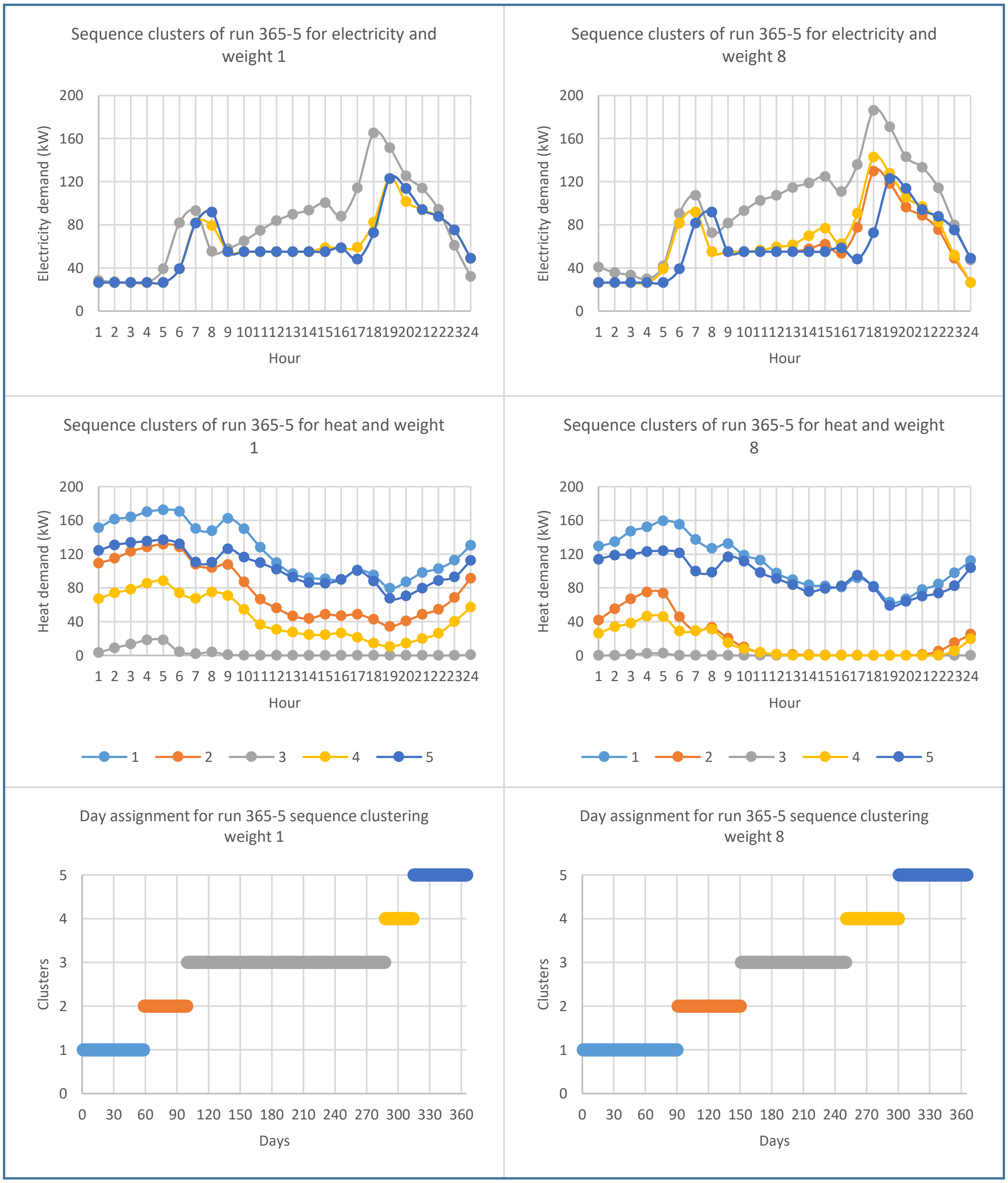

Figure 6-11: Sequence clusters of run 365-5 for heat and electricity with day assignment for weight 1 and 8

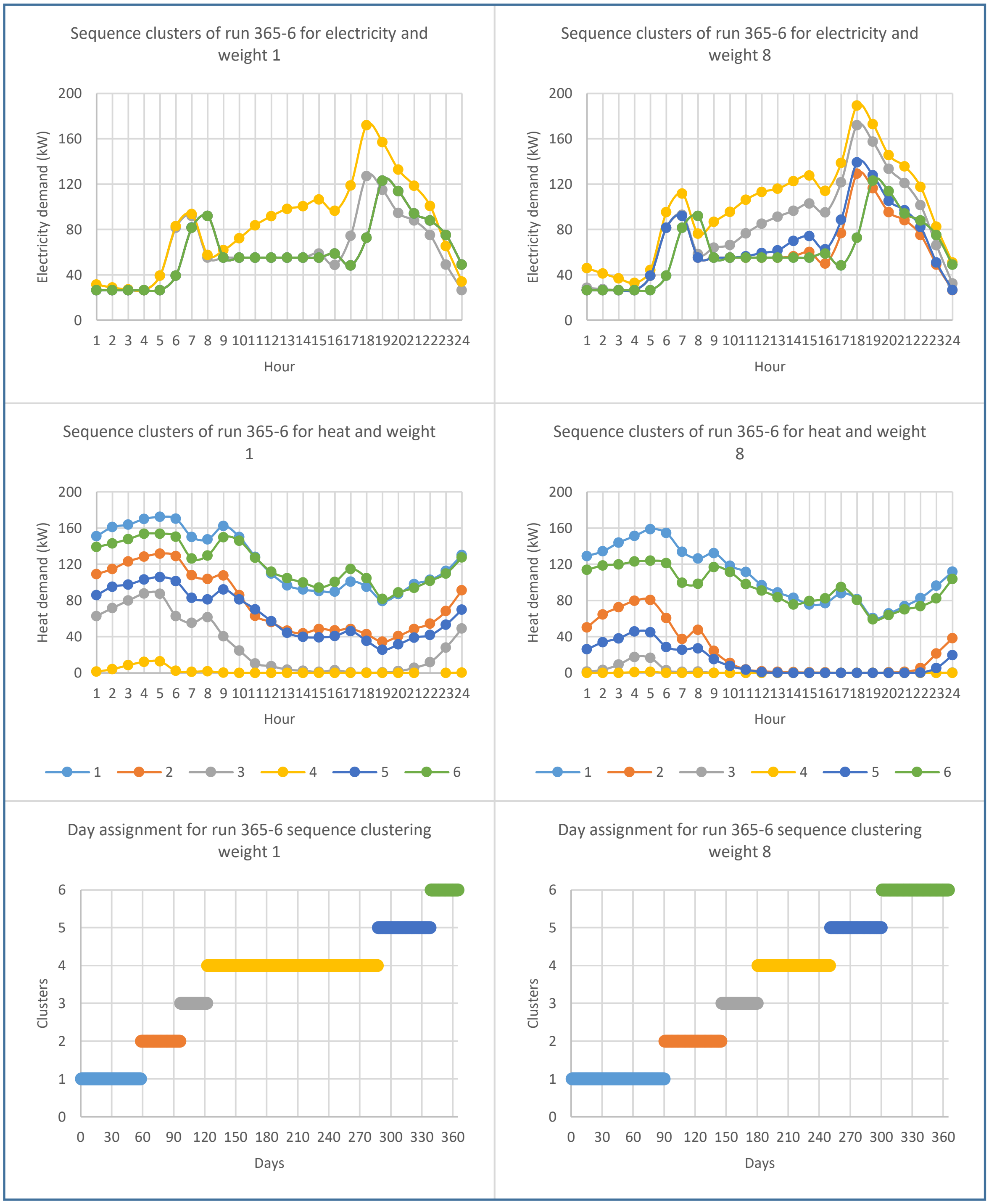

Figure 6-12: Sequence clusters of run 365-6 for heat and electricity with day assignment for weight 1 and 8

Chapter 7
Application of multiple attribute clustering to energy hubs

7.1 Introduction

The aim of this chapter is to apply the clustering heuristic of the previous chapter on a case study that involves multiple attributes and to investigate the impact on solution accuracy. We have already established in the previous chapters that clustering reduces the computational burden significantly and without reducing solution quality. An energy hub model was chosen for this study since it usually has multiple demand attributes. Both normal and sequential clustering will be evaluated on this energy hub model.

The energy hub problem is a medium term decision level intended to minimize the operating cost of existing units while meeting demands. There are several models for the energy hub problem available in the literature, ranging from heuristics to mathematical programming. The energy hub problem in this chapter is modeled as a linear programming (LP)[120] model. The following section presents the energy hub model formulation.

7.2 Energy hub model formulation

The goal of the present energy hub system is to minimize the operating cost while meeting the electricity and heat demands and operating within the units' capacities. Figure 7-1 shows the schematic of the energy hub system. It consists of a boiler and combined heat and power (CHP) unit with the choice of purchasing electricity from the grid. Natural gas is the fuel for both the boiler and CHP. As illustrated, the electricity demand is met by CHP and the grid whereas the heat demand is met by the boiler and CHP.

Eq. 7-1 represents the objective (cost) function. It is basically the operating cost, which includes fuel (gas) consumption, operation and maintenance, and grid expenses.

$$Cost = \sum_{h,d}\left(ELEC_{d,h}^{CHP} * OM_{CHP} + HEAT_{d,h}^{Boiler} * OM_{boiler} + \left(NG_{d,h}^{CHP} + NG_{d,h}^{Boiler}\right) * Price_{NG} + ELEC_{d,h}^{Grid} * Price_{h}^{Grid}\right) \quad \textbf{(7-1)}$$

The following defines variables and parameters of the model:

$ELEC_{d,h}^{Grid}$ = Electricity from grid in d day and h hour (kW)

$ELEC_{d,h}^{CHP}$ = Electricity from CHP in d day and h hour (kW)

$HEAT_{d,h}^{Boiler}$ = Heat from boiler in d day and h hour (kW)

$HEAT_{d,h}^{CHP}$ = Heat from CHP in d day and h hour (kW)

$NG_{d,h}^{CHP}$ = Natural gas for CHP in d day and h hour (m^3)

$Price_h^{Grid}$ = Hourly electricity price from grid ($/kWh)

$Price_{NG}$ = Natural gas price (0.325 $/$m^3$)

OM_{boiler} = Operation and maintenance cost for boiler (0.027 $/kWh)

OM_{CHP} = Operation and maintenance cost for CHP (0.016 $/kWh)

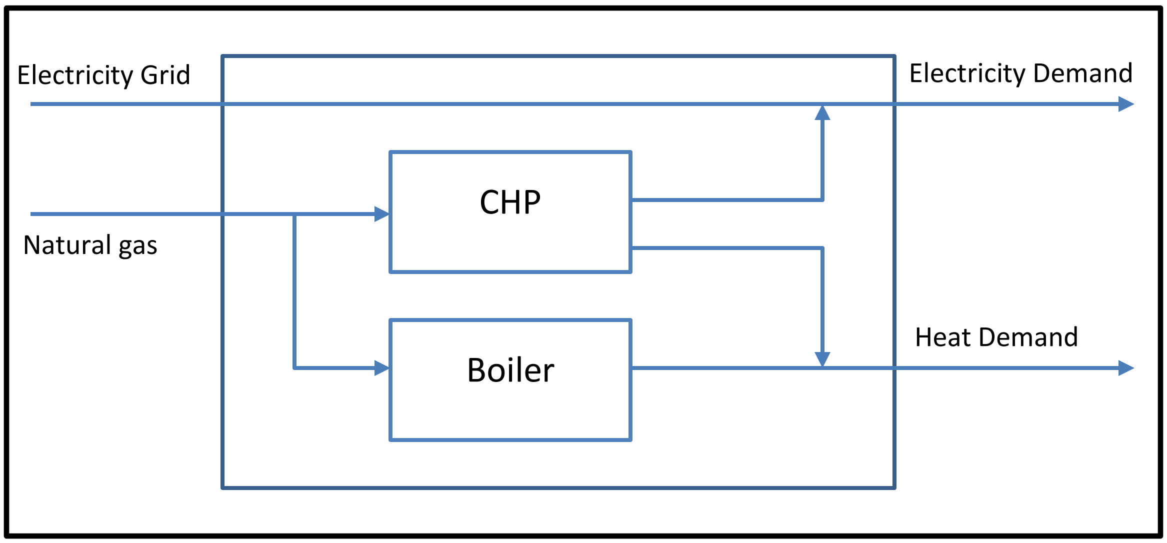

Figure 7-1: Schematic for the energy hub system

Electricity and heat demands will be satisfied at any hour *h* of day *d* by Eq. 7-2 and Eq. 7-3 respectively

$$L_{d,h}^{elec} = ELEC_{d,h}^{Grid} + ELEC_{d,h}^{CHP} \quad \forall\, h, d \qquad \textbf{(7-2)}$$

$$L_{d,h}^{heat} = HEAT_{d,h}^{Boiler} + HEAT_{d,h}^{CHP} \quad \forall\, h, d \qquad \textbf{(7-3)}$$

where $L_{d,h}^{elec}$ and $L_{d,h}^{heat}$ are Hourly electricity and heat demands respectively (kW).

Eq. 7-4 and 7-5 ensure that the power produced by CHP and the boiler at any time, respectively, is within generation power limits.

$$ELEC_{d,h}^{CHP} < Max_{CHP} \quad \forall\, h, d \qquad \textbf{(7-4)}$$

$$HEAT_{d,h}^{Boiler} < Max_{boiler} \quad \forall\, h, d \qquad \textbf{(7-5)}$$

The following equations show the calculations for electricity produced by CHP and heat generated by the boiler and CHP

$$ELEC_{d,h}^{CHP} = NG_{d,h}^{CHP} * \eta_{CHP}^{elec} * b \qquad \forall\, h,d \qquad \textbf{(7-6)}$$

$$HEAT_{d,h}^{Boiler} = NG_{d,h}^{Boiler} * \eta_{boiler}^{heat} * b \qquad \forall\, h,d \qquad \textbf{(7-7)}$$

$$HEAT_{d,h}^{CHP} = NG_{d,h}^{CHP} * \eta_{CHP}^{heat} * b \qquad \forall\, h,d \qquad \textbf{(7-8)}$$

where:

η_{CHP}^{elec} =Electrical efficiency for CHP (0.346)

η_{CHP}^{heat} =Thermal efficiency for CHP (0.44)

η_{boiler}^{heat} =Thermal efficiency for boiler (0.9)

The hourly price of electricity from the grid ($Price_{h}^{Grid}$) which can be found in **Appendix E**. The parameter b is a unit conversion factor for natural gas (10.7 kW/m^3).

7.3 Case Study

The electricity and heat demands along with clustered demands from **Chapter 6** will be used as inputs for this energy hub model. Similar to **Chapter 5**, the cost function is multiplied by parameter N_d as illustrated by Eq. 7-9 to allow comparison between full scale and clustered cases. The parameter N_d represents the number of repetitions (frequency) for corresponding *d* day. The parameter N_t is equal to 1 for full scale case and equal to number of days for cluster case. For example, if cluster 1 represents 45 days, N_d of cluster 1will be equal to 45.

$$Cost = \sum_{h,d}(N_d * (ELEC_{d,h}^{CHP} * OM_{CHP} + HEAT_{d,h}^{Boiler} * OM_{boiler} + (NG_{d,h}^{CHP} + NG_{d,h}^{Boiler}) * Price_{NG} + ELEC_{d,h}^{Grid} * Price_{h}^{Grid})) \qquad \textbf{(7-9)}$$

The full scale model will consider hourly loads of the heat and electricity demands for 365 days whereas the clustered cases will be hourly loads of 4, 5, and 6 clusters (clusters will be considered as days).

GAMS/CPLEX [110] was used to implement this case study on Inter(R) Xeon(R) 2.4 GHz (2 processors), 16 GB RAM workstation. Since the model was LP, it took a few seconds to solve the full scale case, which made it difficult to illustrate the advantage of clustering application in terms of reducing the solution time for this particular example. However, the reduction in computational time through the use of clustering has been established in the previous chapter. Here we focus on solution quality instead.

Table 7-1 presents the objective function values of the full scale case (optimal) and all runs. For a better assessment, Figure 7-2 shows the values of objective function along with the relative error in comparison with the optimal case. All clustered cases are underestimated in terms of the objective function value. Normal clustering is closer to the optimal case than sequence clustering. The error average of objective function is -1.7 % for normal clustering while for sequence clustering, it is -4.2 %. Increasing the number of clusters enhances the solution quality for both normal and sequence clustering as it closes the gap between the optimal and clustered cases' solutions. Varying the weight factors does not have a drastic effect on the values of objective function. This might be because of a similar symmetry in the heat and electricity demands.

Table 7-1: Objective function values in thousands of dollars for all runs with the optimal case of full scale

Weight Factor	Optimal	Number of clusters (Normal)			Number of clusters (Sequence)		
		4	5	6	4	5	6
1	77.1	75.8	76.0	76.5	72.6	73.2	74.3
2	77.1	76.0	75.9	76.3	72.6	73.2	74.7
3	77.1	75.9	75.9	76.4	73.1	74.2	74.7
4	77.1	75.8	75.9	76.3	73.5	74.4	74.8
5	77.1	75.7	76.1	76.3	73.6	73.9	74.6
6	77.1	75.7	75.8	76.1	73.8	73.9	74.4
7	77.1	74.7	75.7	76.2	73.6	73.9	74.2
8	77.1	74.4	74.6	75.9	73.6	74.2	73.6

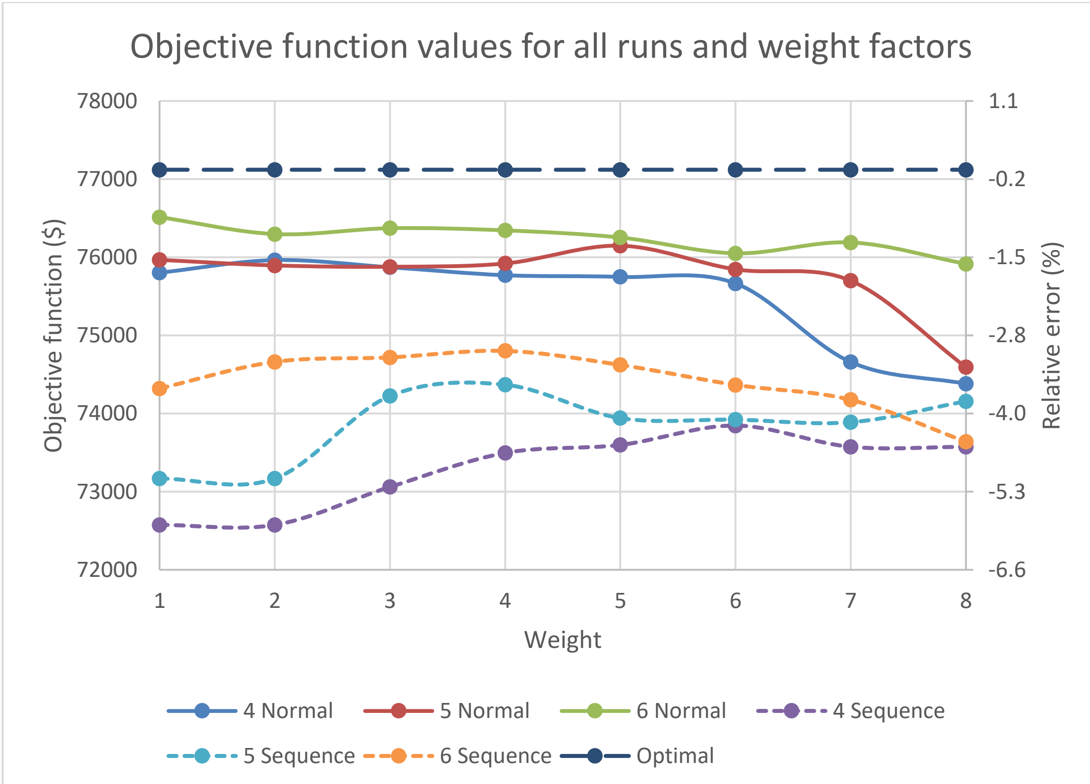

Figure 7-2: Objective function values for all runs and weight factors

In order to examine the effect of increasing the number of clusters, Figure 7-3 and Figure 7-4 show energy production for normal and sequence clustering cases of weight factors 1 & 8 respectively (see **Appendix E** for the results of all runs and for more details). Increasing the number of clusters improves the solution quality as it closes the gap between the optimal non-clustered cases and the clustered cases. In addition, the results of weight factor 1 are much closer to the optimal non-clustered case. Moreover, normal clustering shows a better solution quality than sequence clustering.

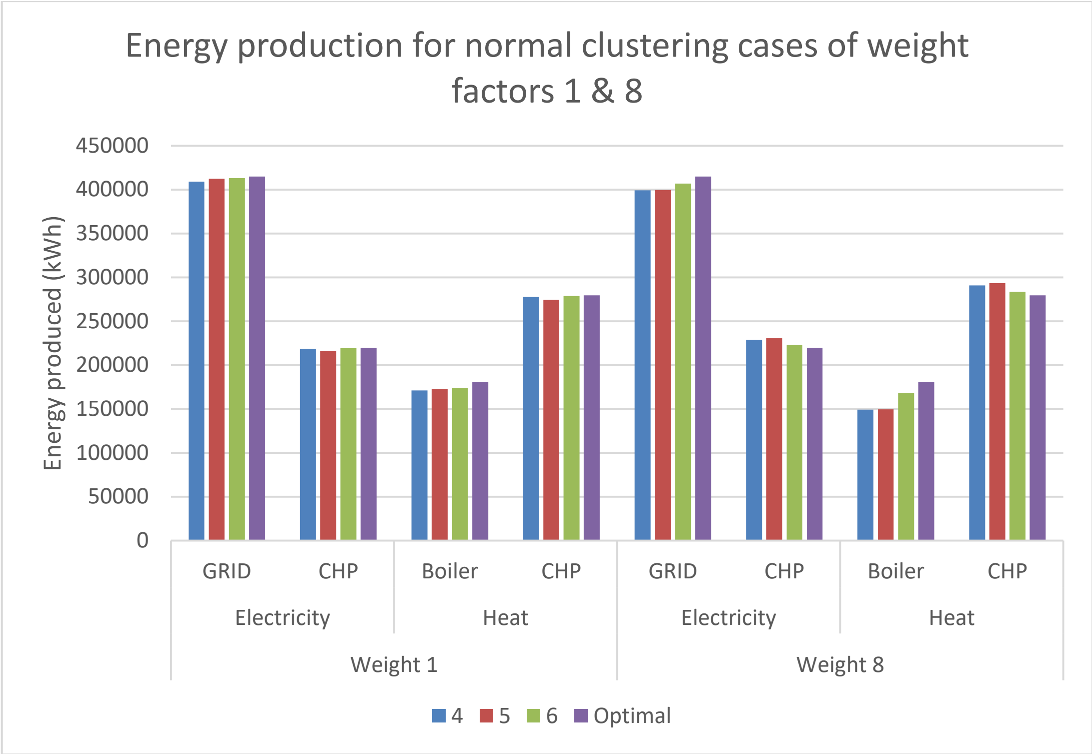

Figure 7-3: Energy production for normal clustering cases of weight factors 1 & 8

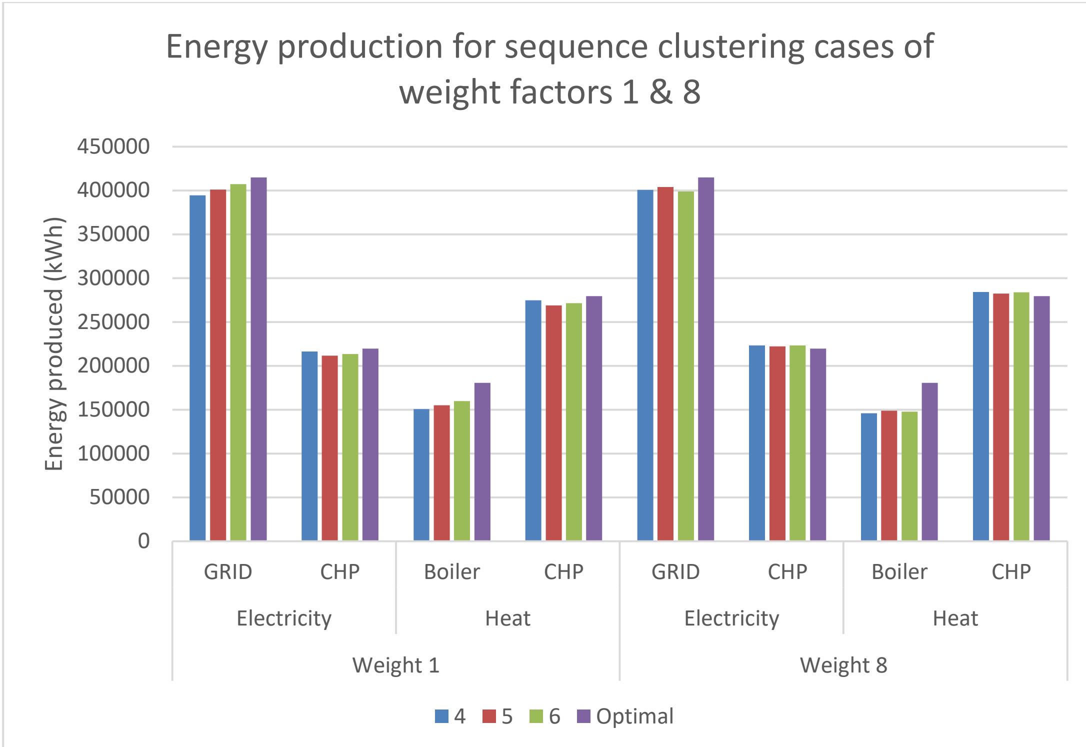

Figure 7-4: Energy production for sequence clustering cases of weight factors 1 & 8

In order to examine the effect of weight factors, Figure 7-5 and Figure 7-6 show Energy production for all weight factors of 5 normal and sequence clusters cases respectively. Varying the weight factors has a gradual effect on the solution quality as the priority switches from electricity to heat. This might be because the demands of electricity and heat have the same symmetry over the whole horizon. As mentioned, the results of weight factor 1 are much closer to the optimal non-clustered case.

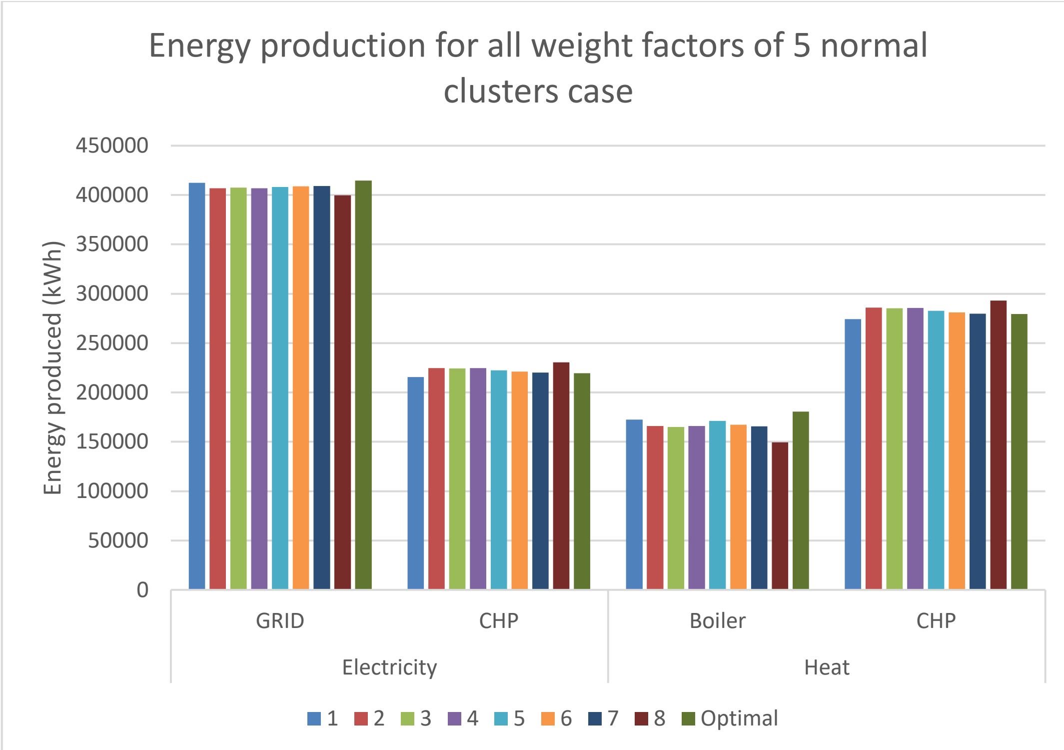

Figure 7-5: Energy production for all weight factors of 5 normal clusters case

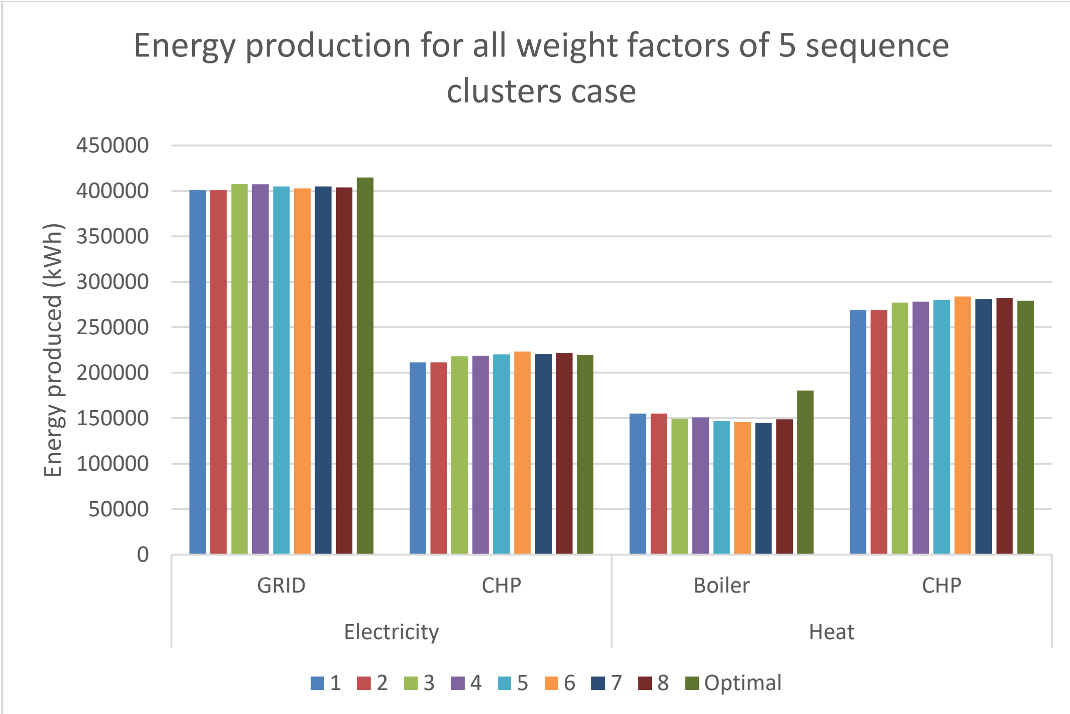

Figure 7-6: Energy production for all weight factors of 5 sequence clusters case

7.4 Conclusion

In this chapter, we investigated the impact on solution accuracy when multiple clustered demands are applied to a planning model. The energy hub model was chosen for this study. The goal of this energy hub system is to minimize the operating cost while meeting the electricity and heat demands and operating within the units' capacities. It consists of a boiler and combined heat and power (CHP) with the choice of purchasing electricity from the grid. Natural gas is the fuel for both the boiler and CHP. The electricity demand is met by CHP and the grid whereas the heat demand is met by the boiler and CHP.

All clustered cases are underestimated in terms of the objective function value. Normal clustering is closer to the optimal case than sequence clustering. The error average of objective function is -1.7 % for normal clustering while for sequence clustering, it is -4.2 %. Increasing the number of clusters enhances the solution quality for both normal and sequence clustering as it closes the gap between the optimal and clustered cases' solutions. Varying the weight factors does not have a drastic effect on the values of the objective function. This might be because of a similar symmetry in the heat and electricity demands. In addition, the results of weight factor 1, prioritizing electricity demand, are much closer to the optimal case. Moreover, normal clustering gave better solution qualities than sequence clustering.

Chapter 8
Conclusion and future work

This thesis was aimed at tackling the integrated supply chain model covering strategic, tactical, and operational decisions through the use of a clustering approach. Since utilizing a shorter time period (e.g. hours) to obtain optimal decisions leads to larger intractable integrated models, the thesis aimed at reducing model size through representing the yearly days by "typical" days that are representative of the year of operation. Clustering in this context focuses on classifying periods of operation in different groups with similar demand patterns and characteristics. Clustering of demand patterns has been little studied and is a complex problem because it has a multi-dimensional facet that deals with shape (i.e. trajectory) of the hourly demand curves and also often times has different attributes (e.g. simultaneous demand for electricity and heat). The thesis presents an important contribution to the clustering literature and illustrates the reduction of computational effort of integrated supply chain models. A mathematical programming based approach was undertaken and the clustering problem was first formulated as a Mixed Integer Nonlinear Program (MINLP) and then appropriate linearization schemes were employed to obtain an exact linearization of the model and render it to a Mixed Integer Linear Program (MILP). This clustering algorithm possesses a unique feature to yield normal and sequence clustering using the same similarity measure. The algorithm could be applied to any two dimensional data like the examples used in this thesis (solar radiation, heat and electricity demand). Despite the model's simplicity, the computational complexity of this clustering algorithm was very evident as illustrated by the findings. Normal clustering and because of the lesser restrictions on sequencing constraints lead to a better objective function, average and standard deviation error than the results of sequence clustering. However, the solution time for sequence clustering was much shorter than for normal clustering. It is concluded that if sequencing restrictions are not important then normal clustering should be utilized. However in cases where it is desired to minimize the number of set-ups and to have less complicated operations through continuous minimum run lengths then sequence clustering should be employed.

The computational burden associated with solving the MILP model with the L_1 metric was found to be much less than that of the L_2 metric. The L_1 criterion is less sensitive to the presence of outliers and is a robust alternative to the L_2 criteria for problems exhibiting large disturbances or incomplete data. The L_1 criterion has been shown in this thesis to have an important advantage over the L_2 criterion in terms of computational effort since the resulting model can be formulated as a linear optimization problem.

However, the computational burden associated with solving the MILP model with the L_1 metric, can still represent a drawback in the application of the models for large planning horizons. To overcome this computational complexity, a heuristic algorithm that employs the mathematical programming models as its main building block and utilizes an iterative scheme that compares a lower and upper bound solution was developed. The heuristic was first applied to cluster 30 day electricity demands for validation purposes. The heuristic succeeded in finding the optimal solution for all the runs. However, the chance of reaching optimal solution varied among the runs. The chance of reaching optimal solution was higher for normal clustering. It took a few minutes to solve the model for both normal and sequence clustering.

The heuristic was then applied to cluster electricity demands for one year. Normal clustering lead to a better objective function, error average and standard deviation than sequence clustering. Increasing the number of clusters decreased the objective function, which is in agreement with findings of the original exact model. In addition, the error average of normal clustering fluctuated as the number of clusters increased while the standard deviation declined. The same trend was observed as with the original mathematical programming model. The solution time for sequence clustering was shorter than for normal clustering.

In order to investigate the impact on solution accuracy when clustered demand is applied to a planning model, the clustered electricity demand was used as input to the unit commitment model. The results showed a great advantage in terms of solution time for clustered cases compared to the full scale solution Furthermore, increasing model size had a minor impact on solution accuracy. In addition, simulation runs suggest that high solution quality can be achieved with a smaller number of clusters. The minimum number of clusters that leads to an acceptable computational effort should therefore be adopted.

The objective function values for clustered cases are very close to the optimal (full scale case). The error range is within ± 0.5 % for all cases. The error gap increased as the number of clusters increased, thus suggesting that increasing the number of clusters does not improve solution quality. This also indicated that there is an optimal number of normal or sequence clusters regardless of cluster quality. The sequence of time horizon for normal clustering can be challenging as there are many ways to arrange it. Results show that the sequence has a major impact on the solution time whereas the effect on solution quality is insignificant.

The thesis proposed an extension of the single attribute clustering model, developed in the **Chapter** to incorporate multiple attributes ,**3** through a multi-objective optimization approach since different attributes have different scales or units. The weighting method approach was applied to deal with such a problem. In addition, the heuristic algorithm used in **Chapter 4**, was extended to consider all weight factors combinations.

Hourly electricity and heat demands for one year were used in this study. The Pareto frontiers were captured for all runs with the weight factor combinations considered in this study. The results show that a better objective function is achieved when the number of clusters increases for both normal and sequence clustering. The normal clustering outperforms the sequence clustering in terms of objective function, error average and standard deviation. The statistical analysis of the heat demand was challenging as suggested by the results, due to the huge fluctuation in the heat demand. Moreover, calculations of relative error were troublesome for the demand that was close zero.

The flexibility of normal clustering has a major advantage over the sequence one. There are many clusters of electricity demand, especially sequence clusters, overlapping with each other. They cannot be merged since they correspond to different days and the clusters of heat demand for these days are different. Therefore, for applications that do not require sequencing it is advantageous to use normal clustering to minimize the computational effort and be able to deal with large scale models. The clustered electricity and heat demands were used as inputs to the energy hub model in order to evaluate the solution quality when multiple clustered demands are applied to a planning model. The goal of this energy hub system is to minimize the operating cost while meeting the electricity and heat demands and operating within the units' capacities. It consists of a boiler and combined heat and power (CHP) with the choice of purchasing electricity from the grid. Natural gas is the fuel for both the boiler and CHP. The electricity demand is met by CHP and the grid whereas the heat demand is met by the boiler and CHP.

All clustered cases are underestimated in terms of the objective function value. Normal clustering is closer to the optimal case than sequence clustering. The error average of objective function is -1.7 % for normal clustering while for sequence clustering, it is -4.2 %. Increasing the number of clusters enhances the solution quality for both normal and sequence clustering as it closes the gap between the optimal and clustered cases' solutions. Varying the weight factors does not have a drastic effect on the values of objective function. This might be because of a similar symmetry in the heat and electricity demands. In addition, results of weight factors 1, prioritizing electricity demand, are much closer to the optimal case. Moreover, normal clustering outperformed sequence clustering in terms of solution quality.

The clustering algorithms developed in this thesis are able to deal with any two dimensional data sets (electricity demand or solar radiation) but also simultaneous demand patterns (e.g. simultaneous demand for heat and power). The most suitable applications of the clustering algorithm are long-term scheduling and integrated scheduling and planning problems. The algorithms can also be useful in clustering customer demand in, for example, energy hubs and can also be of tremendous help in forecasting applications (i.e. it is easier to forecast clusters than individual demand days).

The following are the recommendations for future works.

- Reformulate the clustering model: Since the model is MILP, it is also possible to apply a traditional decomposition method, like Bender or Lagrangean decomposition. Moreover, one can use hints from parameters tuning outputs to come up with a solution method to tackle such an issue.
- Determine the optimal number of clusters for certain classes of applications.
- Examine the effect of the days to clusters ratio for sequence clustering for the heuristics applications.
- Implement different numerical evaluations for the objective function of the clustering model.
- Apply different multi-objective approaches to handle more than two attributes.
- Apply this clustering algorithm to different integrated supply chain models and evaluate the solution accuracy.
- Study different other applications of the developed clustering algorithms such as clustering customer demand and forecasting applications.

Bibliography

(1) Papageorgiou, L. G. Supply Chain Optimisation for the Process Industries: Advances and Opportunities. *Comput. Chem. Eng.* **2009**, *33* (12), 1931.

(2) Grossmann, I. E. Advances in Mathematical Programming Models for Enterprise-Wide Optimization. *Comput. Chem. Eng.* **2012**, *47*, 2.

(3) Maravelias, C. T.; Sung, C. Integration of Production Planning and Scheduling: Overview, Challenges and Opportunities. *Comput. Chem. Eng.* **2009**, *33* (12), 1919.

(4) Weinan, E. *Principles of Multiscale Modeling*; Princeton, USA, 2011.

(5) Karabasov, S.; Nerukh, D.; Hoekstra, A.; Chopard, B.; Coveney, P. V. Multiscale Modelling: Approaches and Challenges. *Philos. Trans. R. Soc. A Math. Phys. Eng. Sci.* **2014**, *372* (2021).

(6) Ge, W.; Wang, L.; Wang, W.; Gao, J.; Zhang, J.; Zhao, H.; Chen, F.; Ren, Y.; Liu, X.; Lu, B.; et al. MULTI-SCALE MODELING. In *Chemical Engineering And Chemical Process Technology*; Pohorecki, R., Bridgwater, J., Molzahn, M., Gani, R., Gallegos, C., Eds.; Eolss Publishers Co. Ltd.,: Singapore, 2010; pp 43–78.

(7) XI, Y.; WILLAM, K.; FRANGOPOL, D. M. *Journal of Engineering Mechanics.*; American Society of Civil Engineers, 1983; Vol. 126.

(8) Ingram, G. D.; Cameron, I. T.; Hangos, K. M. Classification and Analysis of Integrating Frameworks in Multiscale Modelling. *Chem. Eng. Sci.* **2004**, *59* (11), 2171.

(9) Li, J.; Kwauk, M. Exploring Complex Systems in Chemical Engineering—the Multi-Scale Methodology. *Chem. Eng. Sci.* **2003**, *58* (3), 521.

(10) Hughes, T. J. R.; Feijóo, G. R.; Mazzei, L.; Quincy, J.-B. The Variational Multiscale Method—a Paradigm for Computational Mechanics. *Comput. Methods Appl. Mech. Eng.* **1998**, *166* (1), 3.

(11) Li, J.; Zhang, J.; Ge, W.; Liu, X. Multi-Scale Methodology for Complex Systems. *Chem. Eng. Sci.* **2004**, *59* (8), 1687.

(12) Oh, H.; Karimi, I. Regulatory Factors and Capacity-Expansion Planning in Global Chemical Supply Chains. *Ind. Eng. Chem. Res.* **2004**, *43* (13), 3364.

(13) Ryu, J. Multiperiod Planning Strategies with Simultaneous Consideration of Demand Fluctuations and Capacity Expansion. *Ind. Eng. Chem. Res.* **2006**, *45* (19), 6622.

(14) Wagner, H. M.; Whitin, T. M. Dynamic Version of the Economic Lot Size Model. *Manage. Sci.* **1958**, *5* (1), 89.

(15) Veinott, A. F.; Wagner, H. M. Optimal Capacity Scheduling-I. *Oper. Res.* **1962**, *10* (4), 518.

(16) Veinott, A.; Wagner, H. Optimal Capacity Scheduling-II. *Oper. Res.* **1962**, *10* (4), 533.

(17) Barchi, R. H.; Sparrow, F. T.; Vemuganti, R. R. Production, Inventory and Capacity Expansion Scheduling with Integer Variables. *Manage. Sci.* **1975**, *21* (7), 783.

(18) Himmelblau, D. M.; Bickel, T. C. Optimal Expansion of a Hydrodesulfurization Process. *Comput. Chem. Eng.* **1980**, *4* (1), 101.

(19) Grossmann, I. E.; Santibanez, J. Applications of Mixed-Integer Linear Programming in Process Synthesis. *Comput. Chem. Eng.* **1980**, *4* (4), 205.

(20) Shimizu, Y.; Takamatsu, T. Application of Mixed-Integer Linear Programming in Multiterm Expansion Planning under Multiobjectives. *Comput. Chem. Eng.* **1985**, *9* (4), 367.

(21) Hiller, R. S.; Shapiro, J. F. Optimal Capacity Expansion Planning When There Are Learning Effects. *Manage. Sci.* **1986**, *32* (9), 1153.

(22) Jiménez G, A.; Rudd, D. F. Use of a Recursive Mixed-Integer Programming Model to Detect an Optimal Integration Sequence for the Mexican Petrochemical Industry. *Comput. Chem. Eng.* **1987**, *11* (3), 291.

(23) Sahinidis, N.; Grossmann, I.; Fornari, R. E.; Chathrathi, M. Optimization Model for Long Range Planning in the Chemical Industry. *Comput. Chem. Eng.* **1989**, *13* (9), 1049.

(24) Sahinidis, N.; Grossmann, I. Reformulation of the Multiperiod MILP Model for Capacity Expansion of Chemical Processes. *Oper. Res.* **1992**, *40* (1).

(25) Li, S.; Tirupati, D. Dynamic Capacity Expansion Problem with Multiple Products: Technology Selection and Timing of Capacity Additions. *Oper. Res.* **1994**, *42* (5), 958.

(26) Lee, H.-K.; Lee, I.-B.; Reklaitis, G. . Capacity Expansion Problem of Multisite Batch Plants with Production and Distribution. *Comput. Chem. Eng.* **2000**, *24* (2–7), 1597.

(27) Hugo, A.; Pistikopoulos, E. N. Environmentally Conscious Long-Range Planning and Design of Supply Chain Networks. *J. Clean. Prod.* **2005**, *13* (15), 1471.

(28) Bashiri, M.; Badri, H.; Talebi, J. A New Approach to Tactical and Strategic Planning in Production–distribution Networks. *Appl. Math. Model.* **2012**, *36* (4), 1703.

(29) Mariel, K.; Minner, S. Strategic Capacity Planning in Automotive Production Networks under Duties and Duty Drawbacks. *Int. J. Prod. Econ.* **2015**, *170*, 687.

(30) Manne, A. S. Capacity Expansion and Probabilistic Growth. *Econometrica* **1961**, *29* (4), 632.

(31) Giglio, R. J. Stochastic Capacity Models. *Manage. Sci.* **1970**, *17* (3), 174.

(32) Hazell, P. B. R. A Linear Alternative to Quadratic and Semivariance Programming for Farm Planning under Uncertainty. *Am. J. Agric. Econ.* **1971**, *53* (1), 53.

(33) Leondes, C. T.; Nandi, R. K. Capacity Expansion in Convex Cost Networks with Uncertain Demand. *Oper. Res.* **1975**, *23* (6), 1172.

(34) Chao, H. Peak Load Pricing and Capacity Planning with Demand and Supply Uncertainty. *Bell J. Econ.* **1983**, *14* (1), 179.

(35) Sherali, H. D.; Soyster, A. L.; Murphy, F. H.; Sen, S. Intertemporal Allocation of Capital Costs in Electric Utility Capacity Expansion Planning Under Uncertainty. *Manage. Sci.* **1984**, *30* (1), 1.

(36) Modiano, E. M. Derived Demand and Capacity Planning Under Uncertainty. *Oper. Res.* **1987**, *35* (2), 185.

(37) Dantzig, G. B. Planning under Uncertainty Using Parallel Computing. *Ann. Oper. Res.* **1988**, *14* (1), 1.

(38) Eppen, G. D.; Martin, R. K.; Schrage, L. OR Practice—A Scenario Approach to Capacity Planning. *Oper. Res.* **1989**, *37* (4), 517.

(39) Paraskevopoulos, D.; Karakitsos, E.; Rustem, B. Robust Capacity Planning under Uncertainty. *Manage. Sci.* **1991**, *37* (7), 787.

(40) Liu, M. L.; Sahinidis, N. V. Computational Trends and Effects of Approximations in an MILP Model for Process Planning. *Ind. Eng. Chem. Res.* **1995**, *34* (5), 1662.

(41) Liu, M. L.; Sahinidis, N. V. Optimization in Process Planning under Uncertainty. *Ind. Eng. Chem. Res.* **1996**, *35* (11), 4154.

(42) Maravelias, C. T.; Grossmann, I. E. Simultaneous Planning for New Product Development and Batch Manufacturing Facilities. *Ind. Eng. Chem. Res.* **2001**, *40* (26), 6147.

(43) MirHassani, S. A.; Lucas, C.; Mitra, G.; Messina, E.; Poojari, C. A. Computational Solution of Capacity Planning Models under Uncertainty. *Parallel Comput.* **2000**, *26* (5), 511.

(44) Chen, C.-L.; Lee, W.-C. Multi-Objective Optimization of Multi-Echelon Supply Chain Networks with Uncertain Product Demands and Prices. *Comput. Chem. Eng.* **2004**, *28* (6), 1131.

(45) Santoso, T.; Ahmed, S.; Goetschalckx, M.; Shapiro, A. A Stochastic Programming Approach for Supply Chain Network Design under Uncertainty. *Eur. J. Oper. Res.* **2005**, *167* (1), 96.

(46) Mohammadi Bidhandi, H.; Mohd Yusuff, R. Integrated Supply Chain Planning under Uncertainty Using an Improved Stochastic Approach. *Appl. Math. Model.* **2011**, *35* (6), 2618.

(47) Baghalian, A.; Rezapour, S.; Farahani, R. Z. Robust Supply Chain Network Design with Service Level against Disruptions and Demand Uncertainties: A Real-Life Case. *Eur. J. Oper. Res.* **2013**, *227* (1), 199.

(48) Hatefi, S. M.; Jolai, F. Robust and Reliable Forward–reverse Logistics Network Design under Demand Uncertainty and Facility Disruptions. *Appl. Math. Model.* **2014**, *38* (9), 2630.

(49) Chen, L.-M.; Liu, Y. E.; Yang, S.-J. S. Robust Supply Chain Strategies for Recovering from Unanticipated Disasters. *Transp. Res. Part E Logist. Transp. Rev.* **2015**, *77*, 198.

(50) Moon, C.; Lee, Y. H.; Jeong, C. S.; Yun, Y. Integrated Process Planning and Scheduling in a Supply Chain. *Comput. Ind. Eng.* **2008**, *54* (4), 1048.

(51) Grossmann, I. Enterprise-Wide Optimization: A New Frontier in Process Systems Engineering. *AIChE J.* **2005**, *51* (7), 1846.

(52) MORAD, N.; ZALZALA, A. M. S. Genetic Algorithms in Integrated Process Planning and Scheduling. *J. Intell. Manuf.* **1999**, *10* (2), 169.

(53) Zentner, M. G.; Pekny, J. F.; Reklaitis, G. V.; Gupta, J. N. D. Practical Considerations in Using Model-Based Optimization for the Scheduling and Planning of Batch/semicontinuous Processes. *J. Process Control* **1994**, *4* (4), 259.

(54) Li, Z.; Ierapetritou, M. G. Production Planning and Scheduling Integration through Augmented Lagrangian Optimization. *Comput. Chem. Eng.* **2010**, *34* (6), 996.

(55) Dimitriadis, a. D.; Shah, N.; Pantelides, C. C. RTN-Based Rolling Horizon Algorithms for Medium Term Scheduling of Multipurpose Plants. *Computers & Chemical Engineering*. 1997,

pp S1061–S1066.

(56) Li, Z.; Ierapetritou, M. G. Rolling Horizon Based Planning and Scheduling Integration with Production Capacity Consideration. *Chem. Eng. Sci.* **2010**, *65* (22), 5887.

(57) Lin, X.; Floudas, C. a; Modi, S.; Juhasz, N. M. Continuous-Time Optimization Approach for Medium-Range Production Scheduling of a Multiproduct Batch Plant. *Ind. Eng. Chem. Res.* **2002**, *41* (16), 3884.

(58) Sung, C.; Maravelias, C. T. A Projection-Based Method for Production Planning of Multiproduct Facilities. *AIChE J.* **2009**, *55* (10), 2614.

(59) van den Heever, S. a.; Grossmann, I. E. A Strategy for the Integration of Production Planning and Reactive Scheduling in the Optimization of a Hydrogen Supply Network. *Comput. Chem. Eng.* **2003**, *27* (12), 1813.

(60) Erdirik-Dogan, M.; Grossmann, I. E. Simultaneous Planning and Scheduling of Single-Stage Multi-Product Continuous Plants with Parallel Lines. *Comput. Chem. Eng.* **2008**, *32* (11), 2664.

(61) Dogan, M. E.; Grossmann, I. E. A Decomposition Method for the Simultaneous Planning and Scheduling of Single-Stage Continuous Multiproduct Plants. *Ind. Eng. Chem. Res.* **2006**, *45* (1), 299.

(62) Bassett, M. H.; Pekny, J. F.; Reklaitis, G. V. Decomposition Techniques for the Solution of Large-Scale Scheduling Problems. *AIChE J.* **1996**, *42* (12), 3373.

(63) Kallrath, J. Planning and Scheduling in the Process Industry. *OR Spectr.* **2002**, *24* (3), 219.

(64) Ciré, A.; Coban, E.; Hooker, J. Mixed Integer Programming vs. Logic-Based Benders Decomposition for Planning and Scheduling. *... OR Tech. Constraint Program. ...* **2013**, 325.

(65) Timpe, C. Solving Planning and Scheduling Problems with Combined Integer and Constraint Programming. *OR Spectr.* **2002**, *24* (4), 431.

(66) Hooker, J. N. Planning and Scheduling by Logic-Based Benders Decomposition. *Oper. Res.* **2007**, *55* (3), 588.

(67) Warren Liao, T. Clustering of Time Series Data—a Survey. *Pattern Recognit.* **2005**, *38* (11), 1857.

(68) Steinbach, M.; Ertöz, L.; Kumar, V. The Challenges of Clustering High Dimensional Data. In *New Directions in Statistical Physics: Econophysics, Bioinformatics, and Pattern Recognition*; Wille, L. T., Ed.; Springer Berlin Heidelberg: Berlin, Heidelberg, 2004; pp 273–309.

(69) Jain, A. K. Data Clustering: 50 Years beyond K-Means. *Pattern Recognit. Lett.* **2010**, *31* (8), 651.

(70) Köhn, H.-F.; Hubert, L. J. Hierarchical Cluster Analysis. In *Wiley StatsRef: Statistics Reference Online*; John Wiley & Sons, Ltd, 2014.

(71) Kaufman, L.; Rousseeuw, P. J. *Finding Groups in Data: An Introduction to Cluster Analysis*; John Wiley & Sons, Inc.: Hoboken, NJ, USA, 2005.

(72) Rao, M. Cluster Analysis and Mathematical Programming. *J. Am. Stat. Assoc.* **1971**, *66* (335), 622.

(73) Kusiak, A. Analysis of Integer Programming Formulations of Clustering Problems. *Image Vis. Comput.* **1984**, *2* (1), 35.

(74) Koontz, W. L. G.; Narendra, P. M.; Fukunaga, K. A Branch and Bound Clustering Algorithm. *IEEE Trans. Comput.* **1975**, *C-24* (9).

(75) Sağlam, B.; Salman, F. S.; Sayın, S.; Türkay, M. A Mixed-Integer Programming Approach to the Clustering Problem with an Application in Customer Segmentation. *Eur. J. Oper. Res.* **2006**, *173* (3), 866.

(76) Balachandra, P.; Chandru, V. Modelling Electricity Demand with Representative Load Curves. *Energy* **1999**, *24* (3), 219.

(77) Balachandra, P.; Chandru, V. Supply Demand Matching in Resource Constrained Electricity Systems. *Energy Convers. Manag.* **2003**, *44* (3), 411.

(78) Fazlollahi, S.; Becker, G.; Maréchal, F. Multi-Objectives, Multi-Period Optimization of District Energy Systems: III. Distribution Networks. *Comput. Chem. Eng.* **2014**, *66*, 82.

(79) Pochet, Y.; Warichet, F. A Tighter Continuous Time Formulation for the Cyclic Scheduling of a Mixed Plant. *Comput. Chem. Eng.* **2008**, *32* (11), 2723.

(80) Wu, D.; Ierapetritou, M. Cyclic Short-Term Scheduling of Multiproduct Batch Plants Using Continuous-Time Representation. *Comput. Chem. Eng.* **2004**, *28* (11), 2271.

(81) Sahinidis, N.; Grossmann, I. MINLP Model for Cyclic Multiproduct Scheduling on Continuous Parallel Lines. *Comput. Chem. Eng.* **1991**, *15* (2), 85.

(82) Pinto, J.; Grossmann, I. Optimal Cyclic Scheduling of Multistage Continuous Multiproduct Plants. *Comput. Chem. Eng.* **1994**, *18* (9), 797.

(83) Sokal, R. R.; Rohlf, F. J. An Experiment in Taxonomic Judgment. *Syst. Bot.* **1980**, *5* (4), 341.

(84) Gordon, A. D. Constructing Dissimilarity Measures. *J. Classif.* **1990**, *7* (2), 257.

(85) FLEISS, J. L.; ZUBIN, J. ON THE METHODS AND THEORY OF CLUSTERING. *Multivariate Behav. Res.* **1969**, *4* (2), 235.

(86) Everitt, B. S.; Landau, S.; Leese, M.; Landau, D. S.; Leese, D. M. *Cluster Analysis*; John Wiley & Sons, Incorporated: Hoboken, UNKNOWN, 2010.

(87) Mahalanobis, P. C. On the Generalized Distance in Statistics. In *Proceedings of the National Institute of Sciences*; Calcutta, India, 1936; pp 49–55.

(88) Art, D.; Gnanadesikan, R.; Kettenring, J. R. Data-Based Metrics for Cluster Analysis. *Util. Math. A* **1982**, *21*, 75.

(89) De Soete, G. Optimal Variable Weighting for Ultrametric and Additive Tree Clustering. *Qual. Quant.* **1986**, *20* (2), 169.

(90) Brusco, M. J.; Cradit, J. D. A Variable-Selection Heuristic for K-Means Clustering. *Psychometrika* **2001**, *66* (2), 249.

(91) Brusco, M. J.; Köhn, H.-F. Optimal Partitioning of a Data Set Based on the P-Median Model. *Psychometrika* **2007**, *73* (1), 89.

(92) Marton, C. H.; Elkamel, a.; Duever, T. a. An Order-Specific Clustering Algorithm for the Determination of Representative Demand Curves. *Comput. Chem. Eng.* **2008**, *32* (6), 1365.

(93) Qiang, X.; Zhou, B.; Jiang, Q. Alternative Formulations for Unit Commitment. In *2012 Power Engineering and Automation Conference*; IEEE, 2012; pp 1–6.

(94) Bektaş, S.; Şişman, Y. The Comparison of L_1 and L_2-Norm Minimization Methods. *Int. J. Phys.* **2010**, *5* (11), 1721.

(95) Sabo, K. Center-Based L_1-clustering Method. *Int. J. Appl. Math. Comput. Sci.* **2014**, *24* (1), 151.

(96) Lyu, Q.; Lin, Z.; She, Y.; Zhang, C. A Comparison of Typical l_p Minimization Algorithms. *Neurocomputing* **2013**, *119*, 413.

(97) Green, R.; Staffell, I.; Vasilakos, N. Divide and Conquer? *k*-Means Clustering of Demand Data Allows Rapid and Accurate Simulations of the British Electricity System. *IEEE Trans. Eng. Manag.* **2014**, *61* (2), 251.

(98) Chelmis, C.; Kolte, J.; Prasanna, V. K. Big Data Analytics for Demand Response: Clustering over Space and Time. In *2015 IEEE International Conference on Big Data (Big Data)*; IEEE, 2015; pp 2223–2232.

(99) Tominc, P.; Tominc, L. Some Aspects of Differences between L_1 and L_2 Criteria in the Linear Switching Regression. **2000**.

(100) Hirano, S.; Tsumoto, S. Empirical Comparison of Clustering Methods for Long Time-Series Databases. In *Active Mining*; 2005; pp 268–286.

(101) Papadimitriou, S.; Sun, J.; Faloutsos, C. Dimensionality Reduction and Forecasting on Streams. In *Data Streams*; Springer US: Boston, MA, 2007; pp 261–288.

(102) Wilpon, J.; Rabiner, L. A Modified K-Means Clustering Algorithm for Use in Isolated Work Recognition. *IEEE Trans. Acoust.* **1985**, *33* (3), 587.

(103) Singh, S. K. *Process Control: Concepts, Dynamics and Applications*; PHI Learning Pvt. Ltd., 2010.

(104) Vinod, H. D. Integer Programming and the Theory of Grouping. *J. Am. Stat. Assoc.* **1969**, *64* (326), 506.

(105) Mangasarian, O. L. Absolute Value Equation Solution via Dual Complementarity. *Optim. Lett.* **2013**, *7* (4), 625.

(106) Torres, F. E. Linearization of Mixed-Integer Products. *Math. Program.* **1990**, *49* (1–3), 427.

(107) Mirzaesmaeeli, H.; Elkamel, A.; Douglas, P. L.; Croiset, E.; Gupta, M. A Multi-Period Optimization Model for Energy Planning with CO(2) Emission Consideration. *J. Environ. Manage.* **2010**, *91* (5), 1063.

(108) Hourly Ontario and Market Demands, 2002-2014 http://www.ieso.ca/Pages/Power-Data/Data-Directory.aspx (accessed Feb 15, 2015).

(109) Bakker, M.; van Duist, H.; van Schagen, K.; Vreeburg, J.; Rietveld, L. Improving the Performance of Water Demand Forecasting Models by Using Weather Input. *Procedia Eng.* **2014**, *70*, 93.

(110) Corporation, G. D. General Algebraic Modeling System (GAMS) Release 23.3.3. 2009.

(111) GAMS - The Solver Manuals, GAMS Release 23.3.3. Washington, DC, USA 2009.

(112) National Solar Radiation Data Base, Hourly Solar data http://rredc.nrel.gov/solar/old_data/nsrdb/1991-2010/ (accessed Apr 12, 2016).

(113) Üney, F.; Türkay, M. A Mixed-Integer Programming Approach to Multi-Class Data Classification Problem. *Eur. J. Oper. Res.* **2006**, *173* (3), 910.

(114) Xu, R.; Wunsch, D. C. *Clustering*; John Wiley & Sons, Inc.: Hoboken, NJ, USA, 2008; Vol. 1.

(115) Padhy, N. P. Unit Commitment—A Bibliographical Survey. *IEEE Trans. Power Syst.* **2004**, *19* (2), 1196.

(116) Marcovecchio, M. G.; Novais, A. Q.; Grossmann, I. E. Deterministic Optimization of the Thermal Unit Commitment Problem: A Branch and Cut Search. *Comput. Chem. Eng.* **2014**, *67*, 53.

(117) Kazarlis, S. A.; Bakirtizis, A. G.; Petridis, V. A Genetic Algorithm Solution to the Unit Commitment Problem. *Power Syst. IEEE …* **1996**, *11* (February), 83.

(118) *Multiobjective Optimization: Interactive and Evolutionary Approaches*; Branke, J., Deb, K., Miettinen, K., Słowiński, R., Eds.; Lecture Notes in Computer Science; Springer Berlin Heidelberg: Berlin, Heidelberg, 2008; Vol. 5252.

(119) Messac, A.; Ismail-Yahaya, A.; Mattson, C. A. The Normalized Normal Constraint Method for Generating the Pareto Frontier. *Struct. Multidiscip. Optim.* **2003**, *25* (2), 86.

(120) Maroufmashat, A.; Elkamel, A.; Fowler, M.; Sattari, S.; Roshandel, R.; Hajimiragha, A.; Walker, S.; Entchev, E. Modeling and Optimization of a Network of Energy Hubs to Improve Economic and Emission Considerations. *Energy* **2015**, *93*, 2546.

Appendix A

Supplementary data for chapter 3

Table A-1: Hourly electricity demand in MW for first 30 days of 2014 (case study 1)

	1	2	3	4	5	6	7	8	9	10	11	12
1	15862	15462	15182	14925	14807	14934	15246	15510	15675	16018	16479	16950
2	15778	15629	15524	15472	15730	16345	17659	18728	19438	19980	20440	20696
3	17878	17383	17163	17102	17165	17611	18647	19627	20002	20422	20428	20253
4	17011	16507	16043	15874	15853	16120	16344	16996	17411	17583	17929	17924
5	15497	14922	14692	14566	14538	14705	15055	15555	15983	16599	17255	17393
6	15471	14919	14672	14449	14649	15481	16692	18207	18478	18562	18762	18821
7	17347	16909	16786	16777	17042	17760	19030	20467	20767	20283	20515	20594
8	17675	17259	17034	16944	17024	17710	19123	20397	20503	20304	20042	19895
9	17117	16673	16352	16397	16509	17266	18547	20136	19832	19371	19136	18917
10	16428	16161	15739	15602	15747	16544	17749	19176	19266	19070	19209	19103
11	14881	14370	14090	14033	13873	14073	14569	15278	16101	16828	17392	17484
12	14363	13865	13549	13414	13381	13572	13933	14581	15296	15875	16292	16351
13	14370	13957	13800	13822	13926	14501	15880	17592	17639	17422	17374	17398
14	14487	14057	13952	13848	13953	14671	16157	17689	17879	17588	17442	17257
15	14972	14501	14256	14295	14391	15260	16555	17941	18323	18163	18290	18430
16	15796	15353	14959	14931	15133	15675	17075	18488	18604	18459	18299	18244
17	15728	15330	15040	15018	14991	15496	16994	18289	18644	18581	18632	18491
18	15223	14664	14371	14345	14276	14602	15215	16059	16746	17324	17554	17528
19	15353	14817	14499	14442	14545	14827	15266	15681	16287	16855	17236	17451
20	15487	15049	14919	14951	15206	16015	17280	18980	19241	19476	19584	19452
21	17413	17093	16967	17001	17123	17896	19144	20877	20717	20423	20384	20326
22	18268	17955	17865	17922	18045	18577	19724	21276	21093	20905	20716	20620
23	18221	17821	17594	17447	17662	18198	19591	20985	21023	20781	20697	20515
24	18243	17741	17656	17436	17649	18172	19621	20818	20644	20345	20470	20567
25	17136	16633	16283	16029	15956	16279	16715	17419	18150	18577	18899	19020
26	16582	16175	16133	16056	16176	16370	16868	17456	18039	18343	18541	18684
27	16458	16028	15972	16062	16211	16949	18311	19685	19843	19360	19341	19286
28	17807	17332	17317	17271	17554	18153	19434	20632	20650	20233	19972	19659
29	17764	17527	17329	17258	17436	17955	19237	20660	20586	20376	20426	20340
30	17365	16966	16615	16546	16720	17345	18765	19975	20044	19514	19258	19149

	13	14	15	16	17	18	19	20	21	22	23	24
1	17048	17040	16957	17226	18112	19519	19406	18976	18685	18275	17480	16458
2	20718	20651	20576	20543	21276	22476	22320	21981	21933	21156	19946	18756
3	19992	19808	19590	19652	20447	21852	22113	21729	21268	20505	19249	17931
4	17827	17652	17584	17632	18466	19957	19699	18952	18604	17967	17153	16239
5	17535	17428	17357	17756	18691	20056	19888	19257	18853	18151	17090	16070
6	18760	18865	18982	19266	20263	21736	21834	21686	21356	20454	19328	18001
7	20647	20586	20373	20575	21390	22507	22774	22438	22120	21100	19966	18734
8	19612	19705	19600	19935	20771	21794	21735	21564	21202	20291	19189	17896
9	18680	18637	18472	18916	19911	21346	21419	21049	20846	20066	18724	17428
10	18953	18893	18661	18521	19013	19827	19637	19064	18649	18063	17086	15808
11	17483	17359	17128	16984	17472	18295	18169	17610	17201	16703	15950	15096
12	16442	16357	16244	16373	17182	18406	18632	18289	17856	17132	16184	15289
13	17410	17376	17407	17585	18389	19386	19323	19145	18802	17923	16690	15384
14	17004	16957	16863	17163	18103	19336	19805	19506	19179	18383	17110	15875
15	18363	18371	18380	18638	19708	20346	20330	20278	20085	19308	17887	16579
16	18272	18234	18195	18537	19354	20280	20305	20140	19822	19036	17857	16700
17	18392	18186	17990	18191	18570	19602	19607	19253	19016	18240	17292	16121
18	17487	17232	17270	17400	17996	18982	19333	18818	18434	17899	16943	16099
19	17775	17891	18000	18083	18727	19621	19651	19097	18704	18045	17118	16154
20	19384	19326	19082	19315	20377	21527	21868	21857	21651	20650	19239	18123
21	19973	19986	19845	19885	20862	22209	22552	22268	22194	21492	20193	19049
22	20354	20233	20049	20416	21303	22508	22737	22566	22192	21376	20128	18923
23	20411	20207	20062	20255	20984	22055	22489	22423	22170	21482	20318	19083
24	20385	20430	20268	20202	20782	21891	21966	21723	21332	20699	19373	18267
25	18897	18635	18465	18466	18865	20164	20598	20003	19408	18952	18226	17331
26	18729	18794	18993	19354	20074	21138	21045	20510	19947	19178	18270	17241
27	19368	19413	19345	19606	20394	21591	22056	22025	21534	20872	19844	18582
28	19582	19526	19597	19880	20753	21903	22383	22283	21937	21184	19908	18486
29	20012	19742	19604	19741	20514	21508	22061	21955	21582	20841	19355	18159
30	19032	19112	19236	19441	19942	20701	21297	20957	20531	19679	18575	17565

Table A-2: Solution of $x_{d,x}$ for run 20-3 normal and sequence clustering

Day	Normal 1	Normal 2	Normal 3	Sequence 1	Sequence 2	Sequence 3
1	1			1		
2		1			1	
3		1			1	
4	1				1	
5	1				1	
6			1		1	
7		1			1	
8		1			1	
9		1			1	
10			1		1	
11	1					1
12	1					1
13	1					1
14	1					1
15			1			1
16			1			1
17			1			1
18	1					1
19	1					1
20		1				1

Table A-3: Solution of $Cl_{c,h}$ in MW for run 20-3 normal and sequence clustering

Hour	Normal			Sequence		
	1	2	3	1	2	3
1	15,223	17,347	15,728	15,862	17,011	14,972
2	14,664	16,909	15,330	15,462	16,507	14,501
3	14,371	16,786	14,959	15,182	16,043	14,256
4	14,345	16,777	14,931	14,925	15,874	14,295
5	14,276	17,024	14,991	14,807	15,853	14,276
6	14,671	17,611	15,496	14,934	16,544	14,671
7	15,246	18,647	16,994	15,246	17,749	15,880
8	15,681	20,136	18,289	15,510	19,176	17,592
9	16,287	20,002	18,604	15,675	19,438	17,639
10	16,855	20,283	18,562	16,018	19,371	17,422
11	17,374	20,428	18,632	16,479	19,209	17,442
12	17,398	20,253	18,491	16,950	19,103	17,484
13	17,483	19,992	18,392	17,048	18,953	17,487
14	17,359	19,808	18,371	17,040	18,893	17,376
15	17,270	19,600	18,380	16,957	18,982	17,407
16	17,400	19,935	18,537	17,226	19,266	17,585
17	18,112	20,771	19,354	18,112	20,263	18,389
18	19,386	21,852	20,280	19,519	21,736	19,386
19	19,406	22,113	20,305	19,406	21,735	19,607
20	18,976	21,857	20,140	18,976	21,564	19,145
21	18,685	21,651	19,822	18,685	21,202	18,802
22	17,967	20,650	19,036	18,275	20,291	18,045
23	17,090	19,249	17,857	17,480	19,189	17,110
24	16,070	18,123	16,579	16,458	17,896	16,099

Table A-4: Solution of $x_{d,x}$ for run 25-3 normal and sequence clustering

Day	Normal			Sequence		
	1	2	3	1	2	3
1		1		1		
2			1	1		
3			1	1		
4		1		1		
5		1		1		
6	1			1		
7			1	1		
8			1	1		
9	1			1		
10	1			1		
11		1			1	
12		1			1	
13		1			1	
14		1			1	
15	1				1	
16	1				1	
17	1				1	
18		1			1	
19		1			1	
20	1					1
21			1			1
22			1			1
23			1			1
24			1			1
25	1					1

Table A-5: Solution of $Cl_{c,h}$ in MW for run 25-3 normal clustering

Hour	Normal			Sequence		
	1	2	3	1	2	3
1	15,728	15,223	17,675	17,011	14,972	18,221
2	15,330	14,664	17,259	16,507	14,501	17,741
3	14,959	14,371	17,034	16,043	14,256	17,594
4	14,951	14,345	17,001	15,874	14,295	17,436
5	15,133	14,276	17,123	15,853	14,276	17,649
6	15,675	14,671	17,760	16,544	14,671	18,172
7	16,994	15,246	19,123	17,749	15,880	19,591
8	18,289	15,681	20,467	19,176	17,592	20,877
9	18,604	16,287	20,644	19,438	17,639	20,717
10	18,577	16,855	20,345	19,371	17,422	20,423
11	18,762	17,374	20,440	19,209	17,442	20,470
12	18,821	17,398	20,515	19,103	17,484	20,515
13	18,680	17,483	20,354	18,953	17,487	20,354
14	18,635	17,359	20,207	18,893	17,376	20,207
15	18,465	17,270	20,049	18,982	17,407	20,049
16	18,537	17,400	20,202	19,266	17,585	20,202
17	19,354	18,112	20,862	20,263	18,389	20,862
18	20,280	19,386	22,055	21,736	19,386	22,055
19	20,330	19,406	22,320	21,735	19,607	22,489
20	20,140	18,976	21,981	21,564	19,145	22,268
21	19,822	18,685	21,933	21,202	18,802	22,170
22	19,036	17,967	21,100	20,291	18,045	21,376
23	17,887	17,090	19,946	19,189	17,110	20,128
24	16,700	16,070	18,734	17,896	16,099	18,923

Table A-6: Solution of $x_{d,x}$ for run 30-3 normal and sequence clustering

	Normal			Sequence		
Day	**1**	**2**	**3**	**1**	**2**	**3**
1		1		1		
2			1	1		
3			1	1		
4		1		1		
5		1		1		
6	1			1		
7			1	1		
8			1	1		
9	1			1		
10	1			1		
11		1			1	
12		1			1	
13		1			1	
14		1			1	
15	1				1	
16	1				1	
17	1				1	
18		1			1	
19		1			1	
20	1					1
21			1			1
22			1			1
23			1			1
24			1			1
25	1					1
26	1					1
27			1			1

Day	Normal			Sequence		
	1	2	3	1	2	3
28			1			1
29			1			1
30			1			1

Table A-7: Solution of $Cl_{c,h}$ in MW for run 30-3 normal and sequence clustering

Hour	Normal			Sequence		
	1	2	3	1	2	3
1	15,796	15,223	17,675	17,011	14,972	17,413
2	15,353	14,664	17,259	16,507	14,501	17,093
3	15,040	14,371	17,034	16,043	14,256	16,967
4	15,018	14,345	17,001	15,874	14,295	17,001
5	15,206	14,276	17,123	15,853	14,276	17,123
6	16,015	14,671	17,760	16,544	14,671	17,896
7	16,994	15,246	19,123	17,749	15,880	19,144
8	18,289	15,681	20,467	19,176	17,592	20,632
9	18,604	16,287	20,586	19,438	17,639	20,586
10	18,577	16,855	20,304	19,371	17,422	20,233
11	18,762	17,374	20,426	19,209	17,442	19,972
12	18,821	17,398	20,326	19,103	17,484	19,659
13	18,729	17,483	19,992	18,953	17,487	19,582
14	18,637	17,359	19,808	18,893	17,376	19,526
15	18,472	17,270	19,604	18,982	17,407	19,597
16	18,638	17,400	19,885	19,266	17,585	19,741
17	19,708	18,112	20,771	20,263	18,389	20,514
18	20,346	19,386	21,891	21,736	19,386	21,591
19	20,598	19,406	22,113	21,735	19,607	22,056
20	20,278	18,976	21,981	21,564	19,145	21,955
21	19,947	18,685	21,582	21,202	18,802	21,582
22	19,178	17,967	20,872	20,291	18,045	20,841
23	18,226	17,090	19,844	19,189	17,110	19,373
24	17,241	16,070	18,486	17,896	16,099	18,267

Table A-8: Solution of $x_{d,x}$ for run 20-4 normal and sequence clustering

Day	Normal				Sequence			
	1	2	3	4	1	2	3	4
1			1		1			
2		1				1		
3				1		1		
4	1					1		
5			1			1		
6		1				1		
7				1		1		
8				1		1		
9				1		1		
10	1					1		
11			1				1	
12			1				1	
13			1				1	
14			1					1
15	1							1
16	1							1
17	1							1
18			1					1
19			1					1
20		1						1

Table A-9: Solution of $Cl_{c,h}$ in MW for run 20-4 normal and sequence clustering

Hour	Normal				Sequence			
	1	2	3	4	1	2	3	4
1	15,796	15,487	15,223	17,347	15,862	17,011	14,370	15,353
2	15,353	15,049	14,664	16,909	15,462	16,507	13,957	14,817
3	15,040	14,919	14,371	16,786	15,182	16,043	13,800	14,499
4	15,018	14,951	14,345	16,777	14,925	15,874	13,822	14,442
5	15,133	15,206	14,276	17,024	14,807	15,853	13,873	14,545
6	15,675	16,015	14,671	17,611	14,934	16,544	14,073	15,260
7	16,994	17,280	15,246	18,647	15,246	17,749	14,569	16,555
8	18,289	18,728	15,681	20,136	15,510	19,176	15,278	17,941
9	18,604	19,241	16,287	20,002	15,675	19,438	16,101	18,323
10	18,459	19,476	16,855	20,283	16,018	19,371	16,828	18,163
11	18,299	19,584	17,374	20,042	16,479	19,209	17,374	18,290
12	18,430	19,452	17,398	19,895	16,950	19,103	17,398	18,244
13	18,363	19,384	17,483	19,612	17,048	18,953	17,410	18,272
14	18,234	19,326	17,359	19,705	17,040	18,893	17,359	18,186
15	18,195	19,082	17,270	19,590	16,957	18,982	17,128	18,000
16	18,521	19,315	17,400	19,652	17,226	19,266	16,984	18,191
17	19,013	20,377	18,112	20,447	18,112	20,263	17,472	18,727
18	19,957	21,736	19,386	21,794	19,519	21,736	18,406	19,621
19	19,699	21,868	19,406	21,735	19,406	21,735	18,632	19,805
20	19,253	21,857	19,097	21,564	18,976	21,564	18,289	19,506
21	19,016	21,651	18,704	21,202	18,685	21,202	17,856	19,179
22	18,240	20,650	18,045	20,291	18,275	20,291	17,132	18,383
23	17,292	19,328	17,090	19,189	17,480	19,189	16,184	17,292
24	16,239	18,123	16,070	17,896	16,458	17,896	15,289	16,154

Table A-10: Solution of $x_{d,x}$ for run 20-5 normal and sequence clustering

	Normal					Sequence				
Day	**1**	**2**	**3**	**4**	**5**	**1**	**2**	**3**	**4**	**5**
1					1	1				
2		1					1			
3				1			1			
4	1							1		
5					1			1		
6		1						1		
7				1					1	
8				1					1	
9				1					1	
10	1									1
11					1					1
12			1							1
13					1					1
14					1					1
15	1									1
16	1									1
17	1									1
18					1					1
19					1					1
20		1								1

Table A-11: Solution of $Cl_{c,h}$ in MW for run 20-5 normal and sequence clustering

Hour	Normal					Sequence				
	1	2	3	4	5	1	2	3	4	5
1	15,796	15,487	14,363	17,347	15,223	15,862	17,878	15,497	17,347	15,223
2	15,353	15,049	13,865	16,909	14,664	15,462	17,383	14,922	16,909	14,664
3	15,040	14,919	13,549	16,786	14,371	15,182	17,163	14,692	16,786	14,371
4	15,018	14,951	13,414	16,777	14,345	14,925	17,102	14,566	16,777	14,345
5	15,133	15,206	13,381	17,024	14,276	14,807	17,165	14,649	17,024	14,391
6	15,675	16,015	13,572	17,611	14,671	14,934	17,611	15,481	17,710	14,827
7	16,994	17,280	13,933	18,647	15,246	15,246	18,647	16,344	19,030	16,157
8	18,289	18,728	14,581	20,136	15,681	15,510	19,627	16,996	20,397	17,689
9	18,604	19,241	15,296	20,002	16,287	15,675	20,002	17,411	20,503	17,879
10	18,459	19,476	15,875	20,283	16,855	16,018	20,422	17,583	20,283	17,588
11	18,299	19,584	16,292	20,042	17,374	16,479	20,440	17,929	20,042	17,554
12	18,430	19,452	16,351	19,895	17,398	16,950	20,696	17,924	19,895	17,528
13	18,363	19,384	16,442	19,612	17,483	17,048	20,718	17,827	19,612	17,775
14	18,234	19,326	16,357	19,705	17,359	17,040	20,651	17,652	19,705	17,891
15	18,195	19,082	16,244	19,590	17,270	16,957	20,576	17,584	19,600	17,990
16	18,521	19,315	16,373	19,652	17,400	17,226	20,543	17,756	19,935	18,083
17	19,013	20,377	17,182	20,447	18,112	18,112	21,276	18,691	20,771	18,570
18	19,957	21,736	18,406	21,794	19,386	19,519	22,476	20,056	21,794	19,602
19	19,699	21,868	18,632	21,735	19,406	19,406	22,320	19,888	21,735	19,637
20	19,253	21,857	18,289	21,564	19,097	18,976	21,981	19,257	21,564	19,145
21	19,016	21,651	17,856	21,202	18,704	18,685	21,933	18,853	21,202	18,802
22	18,240	20,650	17,132	20,291	18,045	18,275	21,156	18,151	20,291	18,063
23	17,292	19,328	16,184	19,189	17,090	17,480	19,946	17,153	19,189	17,110
24	16,239	18,123	15,289	17,896	16,070	16,458	18,756	16,239	17,896	16,099

Table A-12: Hourly solar radiation in W/m^2 for first 30 days of 2010 (case study 2)

Hour	Days										
	8	9	10	11	12	13	14	15	16	17	18
1	20	161	343	495	594	628	595	496	342	159	18
2	23	182	377	537	637	670	633	529	368	176	21
3	26	189	384	543	644	678	642	540	381	187	25
4	19	168	360	521	624	660	625	521	361	170	20
5	18	164	358	521	628	666	631	528	366	171	21
6	22	182	381	545	651	689	654	550	388	191	26
7	21	177	373	537	643	681	648	546	385	190	26
8	20	168	355	512	613	648	617	520	367	181	26
9	24	181	373	532	636	676	645	548	394	202	32
10	18	167	365	532	641	681	649	547	384	187	26
11	25	192	394	560	668	707	675	574	414	215	36
12	24	186	385	549	656	695	664	564	406	210	35
13	22	183	383	549	658	699	667	567	408	209	35
14	23	181	375	536	641	680	650	554	400	210	38
15	21	172	364	525	631	671	642	545	392	202	35
16	24	180	371	529	633	673	645	551	401	214	40
17	22	174	367	528	636	679	653	560	408	216	40
18	23	183	382	546	654	695	667	571	416	222	43
19	26	191	390	556	665	708	680	584	429	234	48
20	25	188	387	552	660	702	673	578	424	230	47

Hour	Days										
	8	9	10	11	12	13	14	15	16	17	18
21	26	189	385	546	653	694	667	575	424	233	50
22	27	190	384	547	655	699	674	582	431	240	53
23	20	167	364	531	642	686	658	562	406	213	42
24	25	185	382	547	656	699	672	578	425	234	51
25	29	196	394	560	673	722	700	610	459	261	62
26	23	181	384	557	672	718	690	593	434	235	50
27	22	179	387	563	679	725	696	596	434	233	49
28	28	199	405	575	688	734	708	612	456	258	62
29	29	200	403	572	685	729	703	608	453	257	63
30	28	195	395	561	672	715	690	597	445	252	63

Table A-13: Solution of $x_{d,x}$ for run 30-3 normal and sequence clustering

Day	Normal			Sequence		
	1	2	3	1	2	3
1			1	1		
2			1	1		
3			1	1		
4			1	1		
5			1	1		
6			1	1		
7			1	1		
8			1	1		
9			1	1		
10			1	1		
11		1			1	
12		1			1	
13		1			1	
14			1		1	
15			1		1	
16			1		1	
17			1		1	
18		1			1	
19		1			1	
20		1			1	
21		1			1	
22		1			1	
23			1		1	
24		1			1	
25	1					1
26		1				1
27	1					1

Day	Normal			Sequence		
	1	2	3	1	2	3
28	1					1
29	1					1
30	1					1

Table A-14: Solution of $Cl_{c,h}$ in W/m^2 for run 30-3 normal and sequence clustering

Hour	Normal			Sequence		
	1	2	3	1	2	3
8	28	25	21	21	24	28
9	196	188	174	168	185	196
10	395	385	367	365	383	395
11	563	549	531	532	547	563
12	679	658	636	636	655	679
13	725	699	676	670	695	725
14	700	673	645	633	667	700
15	608	578	546	529	571	608
16	453	424	385	368	414	453
17	257	233	190	181	216	257
18	62	48	26	25	42	62

Appendix B

Supplementary data for Chapter 4

Table B-1: Solution of $Cl_{c,h}$ in MW for run 365-4 normal clustering

Hour	Normal cluster			
	1	2	3	4
1	14,689	12,170	16,824	12,947
2	14,298	11,837	16,441	12,530
3	14,108	11,604	16,265	12,355
4	14,043	11,546	16,222	12,365
5	14,216	11,596	16,447	12,731
6	14,960	11,898	17,108	13,760
7	16,064	12,544	18,389	14,932
8	17,086	13,309	19,685	15,544
9	17,543	13,805	19,764	15,746
10	17,834	14,089	19,464	15,875
11	17,866	14,251	19,341	16,009
12	17,924	14,350	19,216	16,043
13	17,832	14,213	19,032	16,093
14	17,894	14,195	18,941	16,072
15	17,890	14,179	18,969	16,080
16	18,179	14,507	19,160	16,403
17	18,692	14,991	19,826	16,776
18	19,257	15,130	20,717	16,858
19	19,368	15,246	21,297	17,161
20	19,194	15,326	21,249	17,166
21	18,833	15,154	20,999	16,673
22	17,967	14,361	20,073	15,746
23	16,866	13,498	18,950	14,530
24	15,606	12,558	17,698	13,529

Table B-2: Solution of $Cl_{c,h}$ in MW for run 365-4 sequence clustering

Hour	Sequence cluster			
	1	2	3	4
1	16,496	15,167	12,789	14,602
2	16,035	14,923	12,492	14,007
3	15,853	14,692	12,318	13,662
4	15,681	14,742	12,297	13,620
5	15,845	15,174	12,631	13,763
6	16,382	16,437	13,695	14,280
7	17,548	17,682	14,903	15,651
8	18,851	17,996	15,511	17,075
9	19,127	17,919	15,682	17,429
10	18,960	17,794	15,816	17,467
11	18,846	17,594	15,923	17,473
12	18,697	17,550	15,982	17,404
13	18,518	17,445	16,036	17,264
14	18,418	17,339	16,008	17,306
15	18,269	17,135	16,049	17,325
16	18,442	17,106	16,398	17,473
17	19,029	17,395	16,701	18,240
18	20,056	17,587	16,717	19,411
19	20,598	18,659	16,975	19,270
20	20,574	19,020	17,026	19,000
21	20,256	18,324	16,586	18,668
22	19,432	17,315	15,601	17,975
23	18,355	15,984	14,379	16,866
24	17,331	15,604	13,378	15,498

Table B-3: Solution of $Cl_{c,h}$ in MW for run 365-5 normal clustering

Hour	Normal cluster				
	1	2	3	4	5
1	15,239	12,147	16,924	12,668	13,634
2	14,802	11,824	16,528	12,343	13,187
3	14,411	11,597	16,339	12,193	12,966
4	14,352	11,534	16,328	12,210	12,893
5	14,545	11,578	16,563	12,594	13,195
6	15,253	11,882	17,202	13,699	14,008
7	16,426	12,544	18,610	15,040	15,059
8	17,640	13,309	19,824	15,449	15,882
9	17,924	13,805	19,843	15,538	16,341
10	18,021	14,079	19,558	15,679	16,649
11	18,117	14,240	19,483	15,796	16,812
12	18,120	14,327	19,317	15,744	16,753
13	18,109	14,204	19,233	15,833	16,775
14	18,090	14,179	19,112	15,742	16,890
15	18,000	14,159	19,046	15,787	17,000
16	18,193	14,450	19,266	16,085	17,262
17	18,876	14,949	19,942	16,397	17,690
18	19,575	15,117	20,829	16,567	18,044
19	19,553	15,240	21,394	16,698	18,011
20	19,407	15,302	21,470	16,875	18,010
21	19,078	15,113	21,090	16,483	17,688
22	18,245	14,343	20,291	15,427	16,760
23	17,118	13,425	19,140	14,132	15,404
24	16,039	12,551	17,867	13,178	14,324

Table B-4: Solution of $Cl_{c,h}$ in MW for run 365-5 sequence clustering

Hour	Sequence cluster				
	1	2	3	4	5
1	15,965	12,585	13,607	12,798	14,602
2	15,656	12,311	13,156	12,469	14,007
3	15,574	12,193	12,908	12,292	13,662
4	15,519	12,247	12,838	12,278	13,620
5	15,764	12,689	13,029	12,615	13,763
6	16,382	13,689	14,024	13,652	14,280
7	17,636	14,868	15,357	14,833	15,651
8	18,463	15,459	16,082	15,442	17,075
9	18,478	15,441	16,545	15,631	17,429
10	18,343	15,587	16,998	15,816	17,467
11	18,353	15,635	17,393	15,975	17,473
12	18,343	15,570	17,596	16,043	17,404
13	18,203	15,612	17,706	16,231	17,264
14	18,186	15,473	17,935	16,166	17,306
15	18,079	15,390	18,076	16,285	17,325
16	18,237	15,656	18,416	16,582	17,473
17	18,698	15,818	18,487	16,911	18,240
18	19,505	15,877	18,193	17,050	19,411
19	20,016	16,136	18,239	17,306	19,270
20	20,003	16,546	18,245	17,154	19,000
21	19,482	16,449	18,108	16,644	18,668
22	18,736	15,457	17,123	15,645	17,975
23	17,764	14,081	15,494	14,431	16,866
24	16,699	13,118	14,258	13,442	15,498

Table B-5: Solution of $Cl_{c,h}$ in MW for run 365-6 normal clustering

Hour	Normal cluster					
	1	2	3	4	5	6
1	15,661	12,147	17,097	12,688	13,604	14,207
2	15,234	11,824	16,663	12,353	13,145	13,702
3	14,975	11,597	16,435	12,208	12,918	13,476
4	14,912	11,534	16,444	12,223	12,872	13,397
5	15,124	11,578	16,587	12,614	13,195	13,639
6	15,756	11,882	17,234	13,768	14,008	14,501
7	16,906	12,520	18,647	15,040	15,023	15,842
8	17,996	13,252	19,839	15,459	15,879	16,833
9	18,193	13,767	19,852	15,552	16,325	17,434
10	18,115	14,079	19,650	15,687	16,608	17,907
11	18,098	14,240	19,499	15,799	16,748	18,197
12	17,964	14,327	19,329	15,764	16,746	18,477
13	17,910	14,204	19,263	15,838	16,775	18,718
14	17,891	14,179	19,132	15,755	16,890	18,837
15	17,824	14,159	19,082	15,796	17,000	18,859
16	18,080	14,450	19,286	16,133	17,253	19,211
17	18,687	14,949	20,074	16,411	17,689	19,359
18	19,572	15,117	20,853	16,568	18,023	19,447
19	19,675	15,240	21,593	16,698	18,011	19,370
20	19,639	15,302	21,564	16,822	18,011	19,204
21	19,215	15,113	21,202	16,483	17,688	18,901
22	18,464	14,343	20,343	15,428	16,760	17,923
23	17,480	13,425	19,198	14,162	15,414	16,466
24	16,248	12,551	17,931	13,179	14,367	15,191

Table B-6: Solution of $Cl_{c,h}$ in MW for run 365-6 sequence clustering

Hour	Sequence cluster					
	1	2	3	4	5	6
1	16,496	14,576	12,531	13,338	12,429	14,602
2	16,035	14,439	12,254	12,951	12,132	14,007
3	15,853	14,339	12,116	12,716	11,931	13,662
4	15,681	14,352	12,201	12,655	11,987	13,620
5	15,845	14,796	12,628	12,901	12,443	13,763
6	16,382	16,176	13,543	13,699	13,681	14,280
7	17,548	17,249	14,810	14,715	15,100	15,637
8	18,851	17,726	15,408	15,513	15,527	17,041
9	19,127	17,608	15,408	15,991	15,490	17,296
10	18,960	17,388	15,499	16,360	15,558	17,378
11	18,846	17,393	15,569	16,726	15,544	17,428
12	18,697	17,030	15,535	17,021	15,423	17,359
13	18,518	16,660	15,579	17,246	15,475	17,219
14	18,418	16,404	15,413	17,297	15,424	17,276
15	18,269	16,362	15,313	17,368	15,426	17,283
16	18,442	16,661	15,577	17,761	15,806	17,466
17	19,029	17,099	15,811	18,025	16,387	18,213
18	20,056	17,419	15,803	17,941	16,678	19,290
19	20,598	18,043	16,034	17,777	17,094	19,259
20	20,574	18,284	16,447	17,767	17,018	18,859
21	20,256	17,947	16,417	17,628	16,249	18,504
22	19,432	16,991	15,414	16,482	15,109	17,967
23	18,355	15,887	14,051	15,137	13,850	16,819
24	17,331	15,032	13,092	14,055	12,928	15,495

Table B-7: Solution of $Cl_{c,h}$ in MW for run 365-7 normal clustering

Hour	Normal cluster						
	1	2	3	4	5	6	7
1	15,620	12,147	17,097	12,668	13,335	14,207	14,507
2	15,227	11,824	16,663	12,346	12,897	13,702	14,118
3	14,959	11,597	16,435	12,205	12,712	13,476	13,924
4	14,911	11,534	16,444	12,223	12,614	13,397	13,917
5	15,078	11,578	16,587	12,614	12,798	13,639	14,179
6	15,713	11,882	17,234	13,768	13,574	14,487	14,735
7	16,770	12,520	18,647	15,040	14,788	15,797	15,589
8	17,996	13,252	19,839	15,459	15,624	16,811	16,230
9	18,193	13,767	19,852	15,542	16,038	17,430	16,519
10	18,115	14,079	19,650	15,682	16,397	17,889	16,800
11	18,098	14,240	19,499	15,796	16,656	18,130	16,686
12	17,964	14,327	19,329	15,744	16,883	18,369	16,624
13	17,910	14,204	19,263	15,837	16,963	18,703	16,412
14	17,891	14,179	19,132	15,755	17,031	18,796	16,242
15	17,824	14,159	19,082	15,802	17,282	18,825	16,177
16	18,080	14,450	19,286	16,137	17,597	19,173	16,459
17	18,687	14,949	20,074	16,422	17,926	19,313	16,985
18	19,572	15,117	20,853	16,577	18,055	19,444	17,402
19	19,675	15,240	21,593	16,673	17,995	19,368	17,963
20	19,639	15,302	21,564	16,822	17,913	19,195	18,100
21	19,215	15,113	21,202	16,475	17,688	18,892	17,525
22	18,464	14,343	20,343	15,427	16,601	17,874	16,738
23	17,480	13,425	19,198	14,132	15,295	16,464	15,860
24	16,248	12,551	17,931	13,174	14,150	15,191	15,023

Table B-8: Solution of $Cl_{c,h}$ in MW for run 365-7 sequence clustering

Hour	Sequence cluster						
	1	2	3	4	5	6	7
1	16,496	14,576	12,531	13,338	12,280	12,947	14,602
2	16,035	14,439	12,254	12,951	12,024	12,634	14,007
3	15,853	14,339	12,116	12,716	11,840	12,437	13,662
4	15,681	14,352	12,201	12,655	11,885	12,427	13,620
5	15,845	14,796	12,628	12,901	12,402	12,608	13,763
6	16,382	16,176	13,543	13,699	13,865	13,304	14,280
7	17,548	17,249	14,810	14,715	15,175	14,654	15,637
8	18,851	17,726	15,408	15,513	15,420	15,701	17,041
9	19,127	17,608	15,408	15,991	15,424	15,746	17,296
10	18,960	17,388	15,499	16,360	15,494	15,679	17,378
11	18,846	17,393	15,569	16,726	15,445	15,859	17,428
12	18,697	17,030	15,535	17,021	15,274	15,802	17,359
13	18,518	16,660	15,579	17,246	15,339	15,898	17,219
14	18,418	16,404	15,413	17,297	15,318	15,977	17,276
15	18,269	16,362	15,313	17,368	15,361	15,955	17,283
16	18,442	16,661	15,577	17,761	15,746	16,280	17,466
17	19,029	17,099	15,811	18,025	16,080	16,893	18,213
18	20,056	17,419	15,803	17,941	16,342	17,585	19,290
19	20,598	18,043	16,034	17,777	16,818	17,545	19,259
20	20,574	18,284	16,447	17,767	16,955	17,330	18,859
21	20,256	17,947	16,417	17,628	16,195	16,878	18,504
22	19,432	16,991	15,414	16,482	15,014	15,823	17,967
23	18,355	15,887	14,051	15,137	13,722	14,675	16,819
24	17,331	15,032	13,092	14,055	12,856	13,793	15,495

Appendix C
Supplementary data for chapter 5

Table C-1: Parameters for unit commitment model [116]

Unit	P^L	P^U	a	b	TU	TD	Hsc	Csc	T^{cold}	T^{ini}	UR	DR
1	150	455	960.61	16.479	8	8	4500	9000	5	8	91	91
2	150	455	944.56	17.447	8	8	5000	10000	5	8	91	91
3	20	130	691.13	16.9	5	5	550	1100	4	-5	26	26
4	20	130	670.65	16.817	5	5	560	1120	4	-5	26	26
5	25	162	423.06	20.447	6	6	900	1800	4	-6	32.4	32.4
6	20	80	355.05	22.972	3	3	170	340	2	-3	16	16
7	25	85	477.93	27.827	3	3	260	520	2	-3	17	17
8	10	55	656.49	26.188	1	1	30	60	0	-1	11	11
9	10	55	663.11	27.414	1	1	30	60	0	-1	11	11
10	10	55	668.53	27.902	1	1	30	60	0	-1	11	11

Where

P_i^L Minimum power generation of unit *i* (MW)

P_i^U Maximum power generation of unit *i* (MW)

a_i, b_i Coefficients of the fuel cost function of unit *i* ($)

TU_i Minimum uptime of unit *i* (hours)

TD_i Minimum downtime of unit *i* (hours)

Hsc_i Hot start cost of unit *i* ($)

Csc_i Cold start costs of unit *i* ($)

T_i^{cold} Cold start hours of unit *i* (hours)

T_i^{ini} Initial status of unit *i* (hours)

DR_i Ramp-down limit of unit *i* (MW)

UR_i Ramp-up limit of unit *i* (MW)

The following figures are the regressions of operation costs for all units.

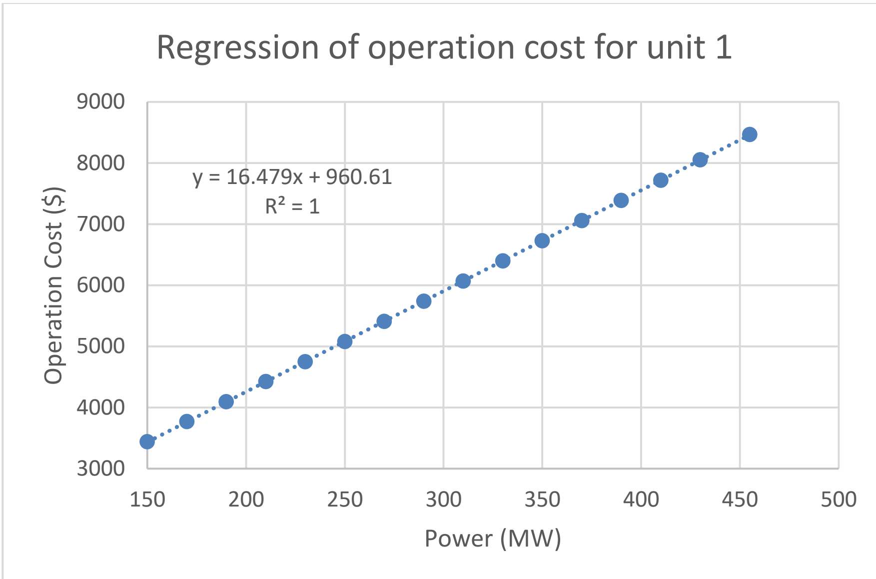

Figure C-1: Regression of operation cost for unit 1

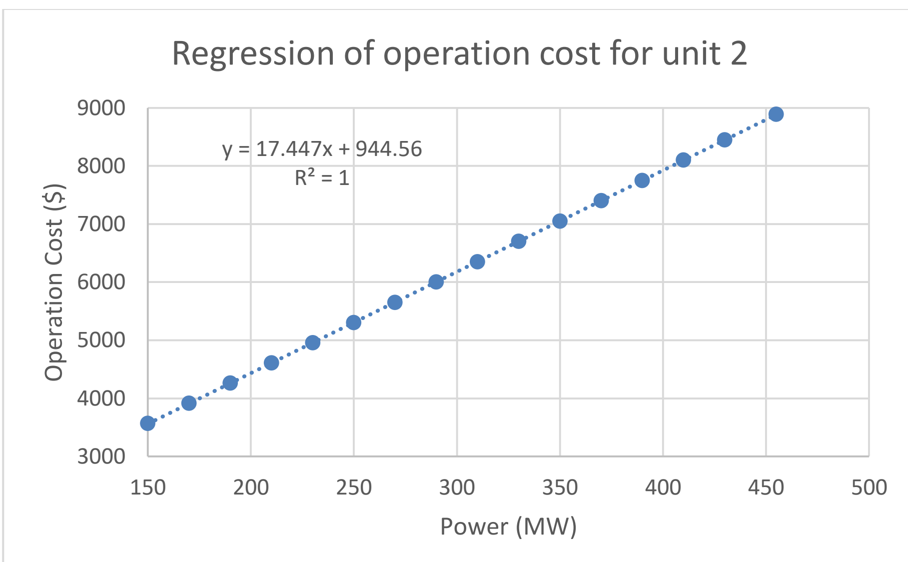

Figure C-2: Regressions of operation cost for unit 2

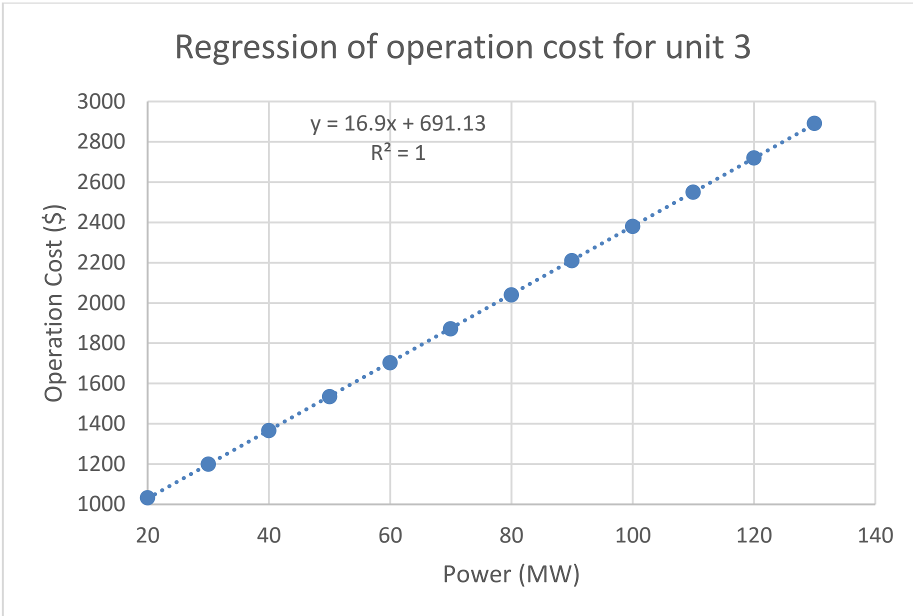

Figure C-3: Regressions of operation cost for unit 3

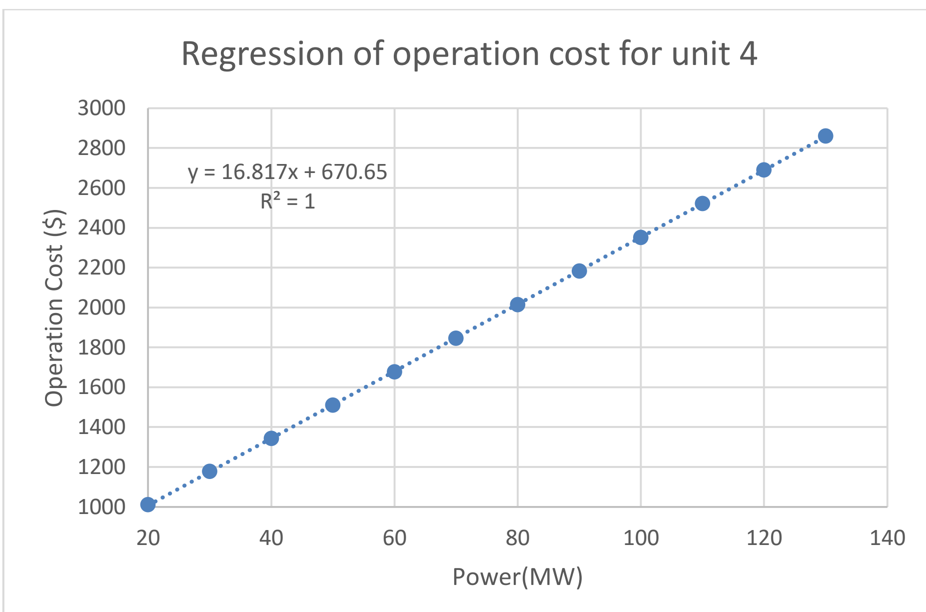

Figure C-4: Regressions of operation cost for unit 4

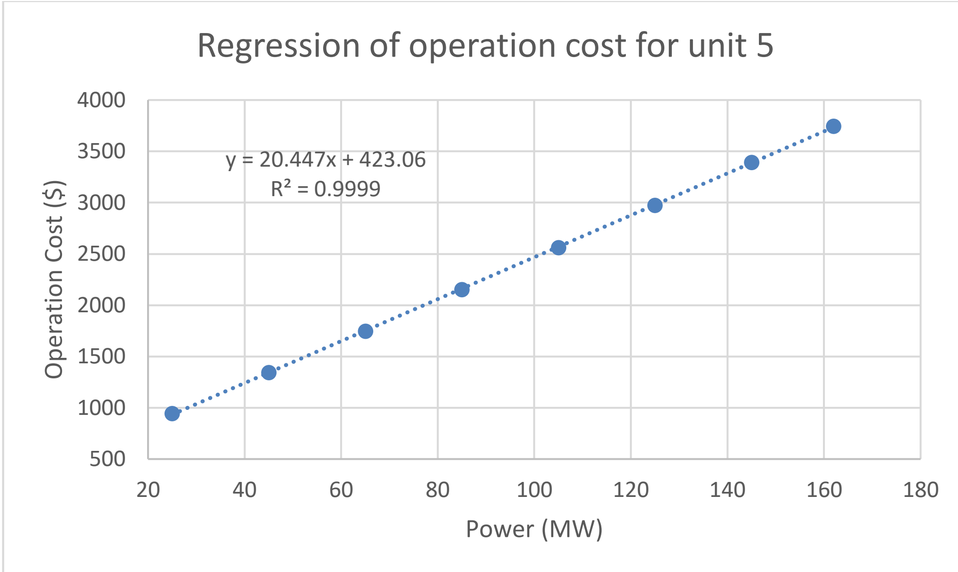

Figure C-5: Regressions of operation cost for unit 5

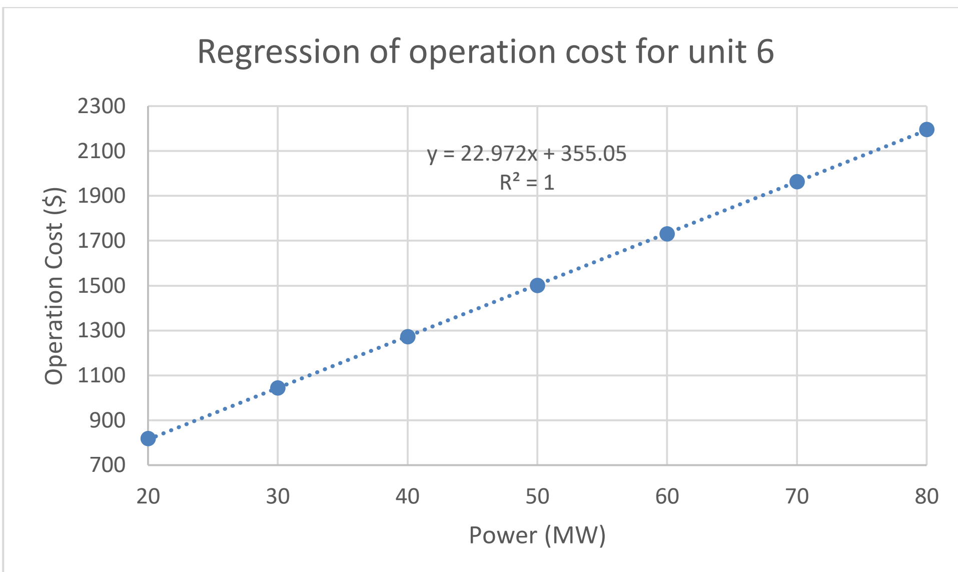

Figure C-6: Regressions of operation cost for unit 6

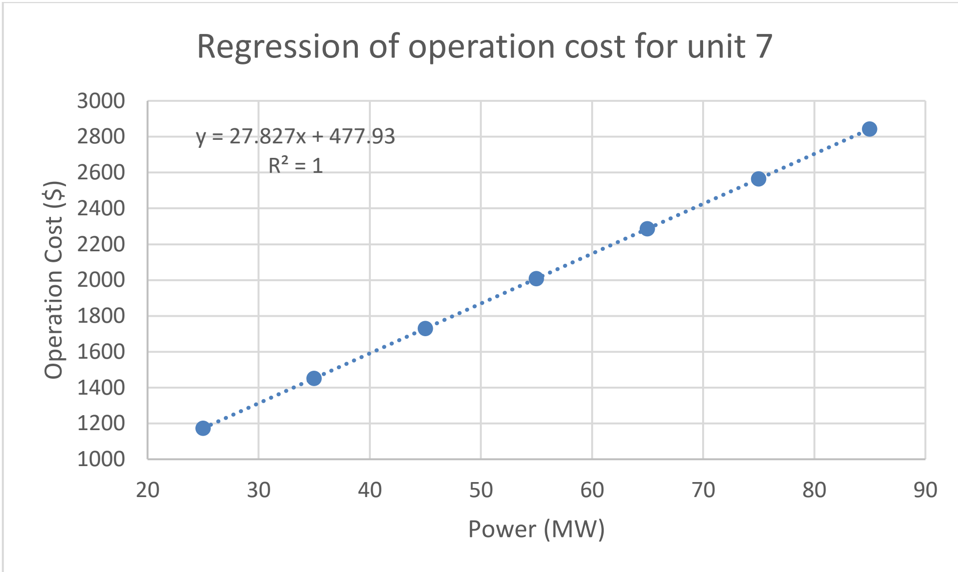

Figure C-7: Regressions of operation cost for unit 7

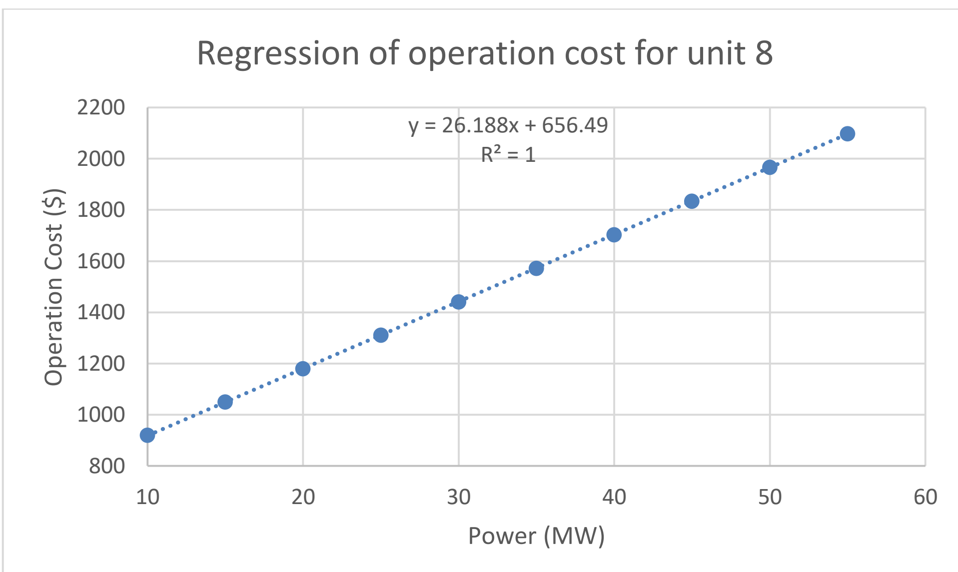

Figure C-8: Regressions of operation cost for unit 8

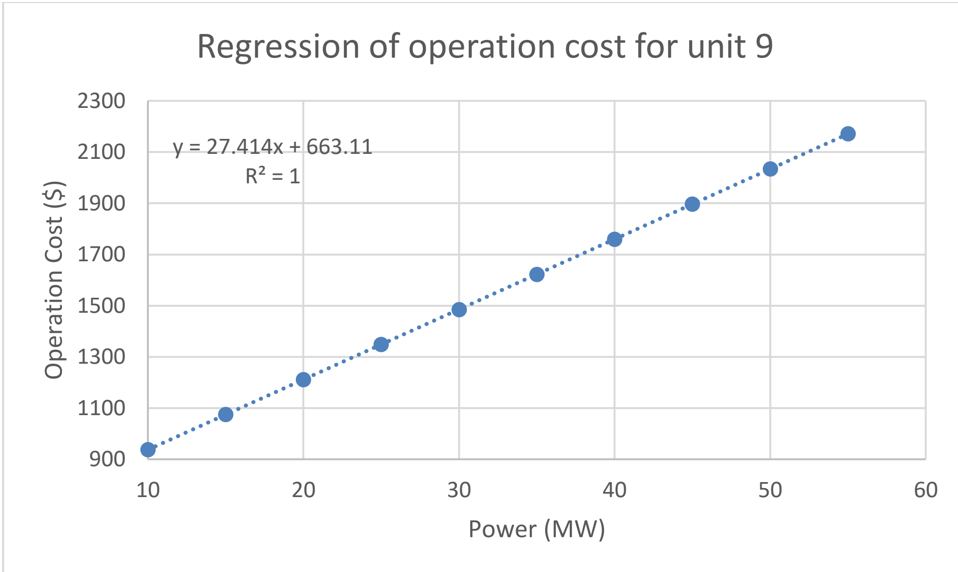

Figure C-9: Regressions of operation cost for unit 9

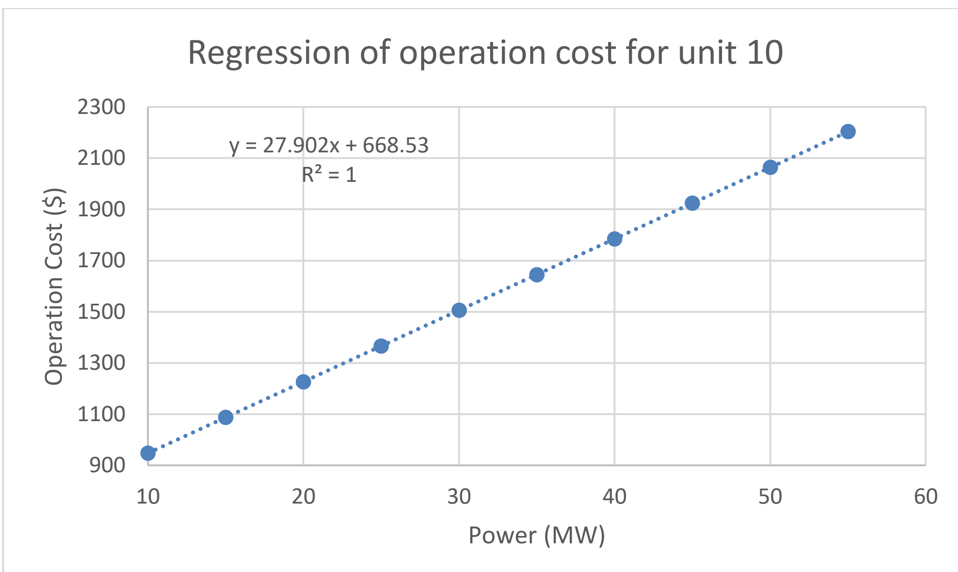

Figure C-10: Regressions of operation cost for unit 10

Table C-2: Produced power (MWh) of optimal and normal clustering for 10 units

		Normal clusters			
Units	**Optimal**	**4**	**5**	**6**	**7**
1	3.98×10^6	3.97×10^6 (-0.13%)	3.97×10^6 (-0.09%)	3.97×10^6 (-0.08%)	3.97×10^6 (-0.12%)
2	2.86×10^6	2.85×10^6 (-0.31%)	2.85×10^6 (-0.40%)	2.85×10^6 (-0.54%)	2.86×10^6 (-0.22%)
4	3.75×10^4	5.76×10^4 (54%)	5.36×10^4 (43%)	4.35×10^4 (16%)	4.35×10^4 (16%)
5	8.55×10^4	5.27×10^4 (-38%)	7.88×10^4 (-7.9%)	8.88×10^4 (3.9%)	7.63×10^4 (-11%)
6	2.65×10^4	3.81×10^4 (44%)	2.62×10^4 (-0.9%)	2.62×10^4 (-1.2%)	3.26×10^4 (23%)
8	1.09×10^3	4.10×10^2 (-62%)	3.50×10^2 (-68%)	0	8.60×10^2 (-21%)

Table C-3: Produced power (MWh) of optimal and sequence clustering for 10 units

		Sequence clusters			
Units	**Optimal**	**4**	**5**	**6**	**7**
1	3.98×10^6	3.97×10^6 (-0.08%)	3.98×10^6 (0.02%)	3.98×10^6 (0.11%)	3.98×10^6 (0.10%)
2	2.86×10^6	2.91×10^6 (1.56%)	2.92×10^6 (1.94%)	2.92×10^6 (2.12%)	2.92×10^6 (2.16%)
4	3.75×10^4	0	0	0	0
5	8.55×10^4	6.92×10^4 (-19%)	7.38×10^4 (-14%)	6.98×10^4 (-18%)	6.98×10^4 (-18%)
6	2.65×10^4	4.66×10^4 (76%)	3.69×10^4 (39%)	4.66×10^4 (76%)	4.71×10^4 (78%)
8	1.09×10^3	0	7.20×10^2 (-34%)	9.40×10^2 (-14%)	9.40×10^2 (-14%)

Table C-4: Online number of hours for optimal and normal clustering for 10 units

		Normal clusters			
Units	**Optimal**	**4**	**5**	**6**	**7**
1	8760	8760 (0%)	8760 (0%)	8760 (0%)	8760 (0%)
2	8760	8760 (0%)	8760 (0%)	8760 (0%)	8760 (0%)
4	289	451 (56.1%)	420 (45.3%)	341 (18.0%)	341 (18.0%)
5	2459	1470 (-40.2%)	2549 (3.7%)	2673 (8.7%)	2176 (-11.5%)
6	1303	1906 (46.3%)	1312 (0.7%)	1301 (-0.2%)	1621 (24.4%)
8	109	41 (-62.4%)	35 (-67.9%)	0	86 (-21.1%)

Table C-5: Online number of hours for optimal and sequence clustering for 10 units

		Sequence clusters			
Units	**Optimal**	**4**	**5**	**6**	**7**
1	8760	8760 (0%)	8760 (0%)	8760 (0%)	8760 (0%)
2	8760	8760 (0%)	8760 (0%)	8760 (0%)	8760 (0%)
4	289	0	0	0	0
5	2459	1652 (-32.8%)	2241 (-8.9%)	1723 (-29.9%)	1723 (-29.9%)
6	1303	2330 (78.8%)	1845 (41.6%)	2328 (78.7%)	2356 (80.8%)
8	109	0	72 (-33.9%)	94 (-13.8%)	94 (-13.8%)

Table C-6: Produced power (MWh) of optimal and normal clustering for 20 units

		Normal clusters			
Units	**Optimal**	**4**	**5**	**6**	**7**
1	3.99×10^{6}	3.98×10^{6} (-0.04%)	3.9×10^{6} (-0.12%)	$3.98\ 10^{6}$ (-0.21%)	3.97×10^{6} (-0.33%)
2	3.98×10^{6}	3.99×10^{6} (0.03%)	3.99×10^{6} (0.03%)	3.99×10^{6} (0.03%)	3.98×10^{6} (0.01%)
3	3.34×10^{6}	3.51×10^{6} (4.88%)	3.31×10^{6} (-1.15%)	3.29×10^{6} (-1.56%)	3.57×10^{6} (6.78%)
4	2.32×10^{6}	2.24×10^{6} (-3.33%)	2.41×10^{6} (3.95%)	2.45×10^{6} (5.74%)	2.14×10^{6} (-7.52%)
5	1.17×10^{4}	0	0	0	0
6	5.13×10^{3}	0	0	0	0
7	8.11×10^{4}	0	0	8.27×10^{4} (1.94%)	0
8	8.09×10^{4}	6.40×10^{4} (-21%)	1.30×10^{5} (60.8%)	0	4.03×10^{4} (-50.2%)
9	7.91×10^{4}	9.87×10^{4} (24.8%)	4.49×10^{4} (-43.2%)	1.03×10^{5} (30%)	1.18×10^{5} (49.3%)
10	6.28×10^{4}	4.74×10^{4} (-24.6%)	6.67×10^{4} (6.19%)	2.78×10^{4} (-55.8%)	1.02×10^{5} (61.56%)
11	1.53×10^{4}	2.30×10^{4} (49.7%)	1.38×10^{4} (-9.82%)	2.59×10^{4} (68.8%)	1.93×10^{4} (25.46%)
12	1.28×10^{4}	0	2.97×10^{4} (133%)	1.59×10^{4} (24.60%)	1.39×10^{4} (8.53%)
15	8.70×10^{2}	0	3.50×10^{2} (-59.8%)	0	3.60×10^{2} (-58.6%)
16	1.17×10^{3}	4.10×10^{2} (-65%)	1.26×10^{3} (7.69%)	8.50×10^{2} (-27.4%)	5.00×10^{2} (-57.3%)
17	4.00x10	0	0	0	0
18	1.00x10	0	0	0	0

Table C-7: Produced power (MWh) of optimal and sequence clustering for 20 units

		Sequence clusters			
Units	**Optimal**	**4**	**5**	**6**	**7**
1	$3.99x10^6$	$3.96x10^6$ (-0.52%)	$3.98x10^6$ (-0.21%)	$3.98x10^6$ (-0.16%)	$3.98x10^6$ (-0.16%)
2	$3.98x10^6$	$3.98x10^6$ (-0.01%)	$3.98x10^6$ (-0.13%)	$3.9x10^6$ (-0.01%)	$3.98x10^6$ (-0.03%)
3	$3.34x10^6$	$3.54x10^6$ (5.71%)	$3.54x10^6$ (5.73%)	$3.52x10^6$ (5.30%)	$3.52x10^6$ (5.36%)
4	$2.32x10^6$	$2.29x10^6$ (-1.18%)	$2.34x10^6$ (1.08%)	$2.33x10^6$ (0.84%)	$2.33x10^6$ (0.58%)
5	$1.17x10^4$	0	0	0	0
6	$5.13x10^3$	0	0	0	0
7	$8.11x10^4$	$8.45x10^4$ (4.19%)	0	$8.45x10^4$ (4.19%)	$9.36x10^4$ (15.41%)
8	$8.09x10^4$	0	0	0	0
9	$7.91x10^4$	$6.11x10^4$ (-22.72%)	$7.81x10^4$ (-1.29%)	$6.54x10^4$ (-17.3%)	$9.28x10^4$ (17.40%)
10	$6.28x10^4$	$4.82x10^4$ (-23.33%)	$7.23x10^4$ (15.13%)	$6.81x10^4$ (8.39%)	$4.30x10^4$ (-31.6%)
11	$1.53x10^4$	0	$3.02x10^4$ (96.52%)	0	0
12	$1.28x10^4$	$2.80x10^4$ (119.51%)	0	$6.04x10^3$ (-52.7%)	
15	$8.70x10^2$	$2.20x10^2$ (-74.71%)	$1.31x10^3$ (50.57%)	0	$2.19x10^3$ (151.72%)
16	$1.17x10^3$	0	$7.20x10^2$ (-38.5%)	$9.40x10^2$ (-19.7%)	$2.90x10^2$ (-75.21%)
17	4.00x10	0	0	0	0
18	1.00x10	0	0	0	0

Table C-8: Online number of hours for optimal and normal clustering for 20 units

		Normal clusters			
Units	**Optimal**	**4**	**5**	**6**	**7**
1	8760	8760 (0%)	8760 (0%)	8760 (0%)	8760 (0%)
2	8760	8760 (0%)	8760 (0%)	8760 (0%)	8760 (0%)
3	8214	8760 (6.6%)	8172 (-0.5%)	8160 (-0.7%)	8760 (6.6%)
4	8230	8301 (0.9%)	8760 (6.4%)	8760 (6.4%)	7560 (-8.1%)
5	91	0	0	0	0
6	40	0	0	0	0
7	625	0	0	636 (1.8%)	0
8	625	492 (-21.3%)	1001 (60.2%)	0	310 (-50.4%)
9	2162	2489 (15.1%)	889 (-58.9%)	2844 (31.5%)	1895 (-12.3%)
10	2318	1593 (-31.3%)	2514 (8.5%)	928 (-60.0%)	3458 (49.2%)
11	758	1133 (49.5%)	692 (-8.7%)	1272 (67.8%)	952 (25.6%)
12	630	0	1487 (136%)	781 (24.0%)	691 (9.7%)
15	87	0	35 (-59.8%)	0	36 (-58.6%)
16	117	41 (-65.0%)	126 (7.7%)	85 (-27.4%)	50 (-57.3%)
17	4	0	0	0	0
18	1	0	0	0	0

Table C-9: Online number of hours for optimal and sequence clustering for 20 units

		Sequence clusters			
Units	**Optimal**	**4**	**5**	**6**	**7**
1	8760	8760 (0%)	8760 (0%)	8760 (0%)	8760 (0%)
2	8760	8760 (0%)	8760 (0%)	8760 (0%)	8760 (0%)
3	8214	8760 (6.65%)	8760 (6.65%)	8760 (6.65%)	8760 (6.65%)
4	8230	8760 (6.44%)	8760 (6.44%)	8760 (6.44%)	8760 (6.44%)
5	91	0	0	0	0
6	40	0	0	0	0
7	625	650 (4.00%)	0	650 (4.00%)	720 (15.20%)
8	625	0	0	0	0
9	2162	1805 (-16.5%)	2042 (-5.6%)	2008 (-7.1%)	3107 (43.7%)
10	2318	1742 (-24.8%)	2266 (-2.2%)	2598 (12.1%)	1593 (-31.3%)
11	758	0	1508 (98.9%)	0	0
12	630	1401 (122.4%)	0	302 (-52.1%)	0
15	87	22 (-74.7%)	131 (50.6%)	0	219 (152%)
16	117	0	72 (-38.5%)	94 (-19.7%)	29 (-75.2%)
17	4	0	0	0	0
18	1	0	0	0	0

Table C-10: Produced power (MWh) of optimal and normal clustering for 30 units

		Normal clusters			
Units	**Optimal**	**4**	**5**	**6**	**7**
1	$3.99x10^6$	$3.98x10^6$ (-0.06%)	$3.99x10^6$ (0.01%)	$3.98x10^6$ (-0.21%)	$3.97x10^6$ (-0.37%)
2	$3.99x10^6$	$3.99x10^6$ (0.01%)	$3.99x10^6$ (0.01%)	$3.98x10^6$ (-0.06%)	$3.98x10^6$ (-0.12%)
3	$3.99x10^6$	$3.99x10^6$ (0.02%)	$3.99x10^6$ (0.02%)	$3.99x10^6$ (0.02%)	$3.99x10^6$ (0.02%)
4	$3.46x10^6$	$2.93x10^6$ (-15.38%)	$3.80x10^6$ (9.94%)	$3.55x10^6$ (2.66%)	$3.13x10^6$ (-9.51%)
5	$2.61x10^6$	$3.02x10^6$ (16.05%)	$2.58x10^6$ (-0.92%)	$2.52x10^6$ (-3.16%)	$2.96x10^6$ (13.56%)
6	$2.48x10^6$	$2.64x10^6$ (6.43%)	$2.21x10^6$ (-10.8%)	$2.50x10^6$ (0.67%)	$2.52x10^6$ (1.61%)
7	$4.53x10^3$	0	0	0	0
8	$1.22x10^4$	0	0	0	0
9	0	$5.33x10^4$	0	0	0
10	$7.44x10^4$	0	$1.25x10^5$ (68.7%)	$4.03x10^4$ (-45.8%)	$4.03x10^4$ (-45.8%)
11	$7.76x10^4$	$5.86x10^4$ (-24.5%)	$5.46x10^4$ (-29.6%)	$1.07x10^5$ (37.4%)	$9.41x10^4$ (21.3%)
12	$7.70x10^4$	0	0	$7.87x10^4$ (2.20%)	$7.80x10^4$ (1.35%)
13	$6.84x10^4$	$9.21x10^4$ (34.6%)	$5.97x10^4$ (-12.8%)	$6.54x10^4$ (-4.34%)	$7.54x10^4$ (10.28%)
14	$6.13x10^4$	$7.56x10^4$ (23.34%)	$8.63x10^4$ (40.73%)	0	$6.69x10^4$ (9.05%)
15	$5.86x10^4$	$6.55x10^4$ (11.69%)	0	$8.26x10^4$ (40.95%)	0

		Normal clusters			
Units	**Optimal**	**4**	**5**	**6**	**7**
16	1.05×10^4	2.72×10^4 (158%)	3.95×10^4 (276%)	1.42×10^4 (35.25%)	1.23×10^4 (17.37%)
17	9.24×10^3	0	1.51×10^4 (63.72%)	2.80×10^4 (203%)	2.49×10^4 (170%)
18	9.43×10^3	0	9.52×10^3 (0.99%)	0	0
22	4.60×10^2	0	0	1.17×10^3 (154 %)	0
23	1.80×10^2	9.20×10^2 (411%)	9.10×10^2 (406%)	0	2.31×10^3 (1183%)
24	1.05×10^3	0	0	0	0
25	1.00x10	0	0	0	0
27	1.00x10	0	0	0	0
28	2.00x10	0	0	0	0
30	1.00x10	0	0	0	0

Table C-11: Produced power (MWh) of optimal and sequence clustering for 30 units

		Sequence clusters			
Units	**Optimal**	**4**	**5**	**6**	**7**
1	$3.99x10^6$	$3.99x10^6$ (-0.01%)	$3.99x10^6$ (0.01%)	$3.98x10^6$ (-0.06%)	$3.98x10^6$ (-0.14%)
2	$3.99x10^6$	$3.99x10^6$ (0.01%)	$3.99x10^6$ (0.01%)	$3.99x10^6$ (-0.01%)	$3.98x10^6$ (-0.14%)
3	$3.99x10^6$	$3.99x10^6$ (0.02%)	$3.99x10^6$ (0.02%)	$3.98x10^6$ (-0.11%)	$3.98x10^6$ (-0.10%)
4	$3.46x10^6$	$3.53x10^6$ (2.10%)	$3.77x10^6$ (9.07%)	$3.73x10^6$ (7.82%)	$3.75x10^6$ (8.50%)
5	$2.61x10^6$	$2.81x10^6$ (7.97%)	$2.14x10^6$ (-17.8%)	$2.74x10^6$ (5.00%)	$2.65x10^6$ (1.77%)
6	$2.48x10^6$	$2.38x10^6$ (-4.05%)	$2.81x10^6$ (13.3%)	$2.32x10^6$ (-6.81%)	$2.38x10^6$ (-4.21%)
7	$4.53x10^3$	0	0	0	0
8	$1.22x10^4$	0	0	0	0
10	$7.44x10^4$	0	0	0	0
11	$7.76x10^4$	$8.45x10^4$ (8.89%)	0	0	0
12	$7.70x10^4$	0	0	$1.40x10^5$ (81.4%)	$1.45x10^5$ (88.3%)
13	$6.84x10^4$	$6.35x10^4$ (-7.10%)	$1.14x10^5$ (67.3%)	$6.63x10^4$ (-3.10%)	$7.26x10^4$ (6.10%)
14	$6.13x10^4$	$6.50x10^4$ (6.04%)	$1.21x10^5$ (97.6%)	$4.91x10^4$ (-20.0%)	$4.46x10^4$ (-27.4%)
15	$5.86x10^4$	$5.65x10^4$ (-3.55%)	$9.20x10^4$ (56.9%)	$7.28x10^4$ (24.2%)	$5.22x10^4$ (-10.9%)
16	$1.05x10^4$	0	0	$3.72x10^3$ (-64.6%)	$2.76x10^4$ (162%)

		Sequence clusters			
Units	**Optimal**	**4**	**5**	**6**	**7**
17	$9.24x10^3$	$3.45x10^4$ (274%)	0	0	0
18	$9.43x10^3$	0	$5.53x10^3$ (-41.3%)	0	0
22	$4.60x10^2$	0	0	$2.21x10^3$ (380%)	$6.50x10^2$ (41.30%)
23	$1.80x10^2$	0	$2.50x10^2$ (38.9%)	$1.95x10^3$ (983%)	0
24	$1.05x10^3$	$2.20x10^2$ (-79.0%)	$1.74x10^3$ (65.7%)	$5.80x10^2$ (-44.8%)	$9.40x10^2$ (-10.5%)
25	1.00x10	0	0	0	0
27	1.00x10	0	0	0	0
28	2.00x10	0	0	0	0
30	1.00x10	0	0	0	0

Table C-12: Online number of hours for optimal and normal clustering for 30 units

		Normal clusters			
Units	**Optimal**	**4**	**5**	**6**	**7**
1	8760	8760 (0%)	8760 (0%)	8760 (0%)	8760 (0%)
2	8760	8760 (0%)	8760 (0%)	8760 (0%)	8760 (0%)
3	8760	8760 (0%)	8760 (0%)	8760 (0%)	8760 (0%)
4	8005	6570 (-17.93%)	8760 (9.43%)	8336 (4.13%)	7458 (-6.83%)
5	7882	8438 (7.05%)	8010 (1.62%)	6936 (-12.0%)	8760 (11.14%)
6	8047	8760 (8.86%)	6972 (-13.4%)	8760 (8.86%)	8760 (8.86%)
7	35	0	0	0	0
8	94	0	0	0	0
9	0	410	0	0	0
10	572	0	965 (68.7%)	310 (-45.8%)	310 (-45.8%)
11	597	451 (-24.5%)	420 (-29.6%)	820 (37.4%)	724 (21.3%)
12	592	0	0	605 (2.2%)	600 (1.4%)
13	2173	1960 (-9.8%)	2003 (-7.8%)	2449 (12.7%)	2777 (27.8%)
14	2245	2679 (19.3%)	3273 (45.8%)	0	2675 (19.2%)
15	2195	2224 (1.3%)	0	3305 (50.6%)	0
16	524	1326	1975	711	617

		Normal clusters			
Units	**Optimal**	**4**	**5**	**6**	**7**
		(153%)	(277%)	(35.7%)	(17.7%)
17	459	0	756 (64.71%)	1401 (205%)	1246 (171%)
18	470	0	476 (1.28%)	0	0
22	46	0	0	117 (154%)	0
23	18	92 (411%)	91 (406%)	0	231 (1183%)
24	105	0	0	0	0
25	1	0	0	0	0
27	1	0	0	0	0
28	2	0	0	0	0
30	1	0	0	0	0

Table C-13: Online number of hours for optimal and sequence clustering for 30 units

		Sequence clusters			
Units	**Optimal**	**4**	**5**	**6**	**7**
1	8760	8760 (0%)	8760 (0%)	8760 (0%)	8760 (0%)
2	8760	8760 (0%)	8760 (0%)	8760 (0%)	8760 (0%)
3	8760	8760 (0%)	8760 (0%)	8760 (0%)	8760 (0%)
4	8005	8182 (2.21%)	8760 (9.43%)	8760 (9.43%)	8760 (9.43%)
5	7882	8760 (11.14%)	6046 (-23.3%)	8760 (11.14%)	8580 (8.86%)
6	8047	7348 (-8.69%)	8760 (8.86%)	7618 (-5.33%)	7618 (-5.33%)
7	35	0	0	0	0
8	94	0	0	0	0
10	572	0	0	0	0
11	597	650 (8.88%)	0	0	0
12	592	0	0	1074 (81.42%)	1115 (88.34%)
13	2173	1648 (-24.16%)	2721 (25.22%)	1536 (-29.3%)	1879 (-13.53%)
14	2245	1814 (-19.20%)	3522 (56.88%)	1547 (-31.1%)	1658 (-26.15%)
15	2195	1822 (-16.99%)	3082 (40.41%)	2759 (25.69%)	2001 (-8.84%)
16	524	0	0	186 (-64.5%)	1379 (163%)
17	459	1727	0	0	0

		Sequence clusters			
Units	**Optimal**	**4**	**5**	**6**	**7**
		(276%)			
18	470	0	261 (-44.5%)	0	0
22	46	0	0	221 (380%)	65 (41.30%)
23	18	0	25 (38.89%)	195 (983%)	0
24	105	22 (-79.05%)	174 (65.71%)	58 (-44.8%)	94 (-10.48%)
25	1	0	0	0	0
27	1	0	0	0	0
28	2	0	0	0	0
30	1	0	0	0	0

Appendix D
Supplementary data for chapter 6

Table D-1: Computational statistics Normal 365-4

Weight	Electricity			Heat		
	Average (%)	Std (%)	IAE (MWh)	Average (%)	Std (%)	IAE (MWh)
1	0.95	18.09	62.0	574	18,862	109
2	1.12	17.72	59.6	531	17,974	110
3	0.82	15.23	51.2	813	28,176	114
4	0.71	14.96	50.3	779	26,726	115
5	0.78	14.78	49.7	709	24,864	116
6	0.64	14.26	48.8	775	25,011	118
7	0.38	11.98	37.7	652	19,788	149
8	-0.17	10.57	35.1	1,940	66,593	165

Table D-2: Computational statistics Normal 365-5

Weight	Electricity			Heat		
	Average (%)	Std (%)	IAE (MWh)	Average (%)	Std (%)	IAE (MWh)
1	1.07	18.30	62.4	549	17,766	95
2	1.12	15.67	49.4	426	12,935	100
3	1.11	15.57	49.0	426	13,099	100
4	1.07	15.44	48.4	429	13,100	101
5	0.62	13.36	41.2	533	16,688	109
6	0.45	12.73	39.6	537	15,893	112
7	0.26	11.51	36.8	726	20,728	120
8	-0.08	9.40	29.2	1,013	25,552	160

Table D-3: Computational statistics Normal 365-6

Weight	Electricity			Heat		
	Average (%)	Std (%)	IAE (MWh)	Average (%)	Std (%)	IAE (MWh)
1	1.21	15.97	49.8	258	8,108	89
2	0.80	14.67	46.8	358	10,537	90
3	0.85	14.48	46.1	356	10,443	90
4	0.81	14.20	44.5	370	9,534	91
5	0.55	13.37	41.4	474	14,399	95
6	0.30	11.73	33.4	692	19,766	111
7	0.19	10.06	29.9	916	25,388	121
8	-0.09	9.39	29.2	1,077	27,458	125

Table D-4: Computational statistics Sequence 365-4

Weight	Electricity			Heat		
	Average (%)	Std (%)	IAE (MWh)	Average (%)	Std (%)	IAE (MWh)
1	-0.89	20.55	84.3	939	28,209	159
2	-0.89	20.55	84.3	939	28,209	159
3	0.07	19.57	74.8	690	18,806	164
4	0.48	18.95	69.6	761	22,879	165
5	0.59	18.88	68.7	772	22,766	166
6	0.75	18.94	68.1	691	17,166	167
7	0.48	18.56	66.8	791	22,426	171
8	0.48	18.56	66.8	791	22,426	171

Table D-5: Computational statistics Sequence 365-5

Weight	Electricity			Heat		
	Average (%)	Std (%)	IAE (MWh)	Average (%)	Std (%)	IAE (MWh)
1	-0.55	20.65	83.2	759	24,238	145
2	-0.55	20.65	83.2	759	24,238	145
3	0.63	17.73	60.1	982	28,268	155
4	0.64	17.52	59.5	963	27,832	156
5	0.32	16.45	56.0	1055	34,650	160
6	0.36	15.87	54.7	929	28,813	162
7	0.27	15.49	53.5	928	28,768	166
8	0.28	15.45	53.4	1,040	32,355	166

Table D-6: Computational statistics Sequence 365-6

Weight	Electricity			Heat		
	Average (%)	Std (%)	IAE (MWh)	Average (%)	Std (%)	IAE (MWh)
1	0.25	19.77	74.6	571	16,340	137
2	0.62	19.18	69.6	560	15,525	137
3	0.69	18.70	67.1	543	15,499	140
4	0.54	17.03	58.3	829	25,572	144
5	0.55	16.39	55.8	682	18,461	148
6	0.76	15.43	51.9	790	26,247	160
7	0.35	15.29	51.4	790	23,021	159
8	-0.31	14.53	50.9	1,174	37,545	164

Appendix E

Supplementary data for chapter 7

Table E-1: Hourly grid price[120]

Hour	$Price_h^{Grid}$	Hour	$Price_h^{Grid}$
1	0.077	**13**	0.14
2	0.077	**14**	0.14
3	0.077	**15**	0.14
4	0.077	**16**	0.14
5	0.077	**17**	0.114
6	0.077	**18**	0.114
7	0.114	**19**	0.077
8	0.114	**20**	0.077
9	0.114	**21**	0.077
10	0.114	**22**	0.077
11	0.14	**23**	0.077
12	0.14	**24**	0.077

Table E-2: Energy production (MWh) for 4 normal clusters

Weight	Electricity		Heat	
	GRID	CHP	Boiler	CHP
1	409	218	171	278
2	410	219	171	278
3	410	220	168	280
4	410	220	167	280
5	410	220	166	279
6	411	218	166	278
7	405	225	149	286
8	399	229	149	291
Optimal	415	220	180	279

Table E-3: Energy production (MWh) for 5 normal clusters

Weight	Electricity		Heat	
	GRID	CHP	Boiler	CHP
1	412	216	173	274
2	407	225	166	286
3	407	224	165	285
4	407	225	166	286
5	408	222	171	283
6	409	221	167	281
7	409	220	166	280
8	399	230	149	293
Optimal	415	220	180	279

Table E-4: Energy production (MWh) for 6 normal clusters

	Electricity		Heat	
Weight	**GRID**	**CHP**	**Boiler**	**CHP**
1	413	219	174	279
2	411	220	174	279
3	411	220	174	280
4	412	219	174	278
5	410	220	174	279
6	409	222	168	282
7	410	221	169	281
8	407	223	168	284
Optimal	415	220	180	279

Table E-5: Energy production (MWh) for 4 sequence clusters

	Electricity		Heat	
Weight	**GRID**	**CHP**	**Boiler**	**CHP**
1	394	216	151	275
2	394	216	151	275
3	396	223	146	283
4	401	222	146	283
5	399	225	147	286
6	399	226	149	288
7	400	223	146	284
8	400	223	146	284
Optimal	415	220	180	279

Table E-6: Energy production (MWh) for 5 sequence clusters

Weight	Electricity		Heat	
	GRID	CHP	Boiler	CHP
1	401	211	155	269
2	401	211	155	269
3	408	218	149	277
4	407	219	151	278
5	405	220	146	280
6	403	223	146	284
7	405	221	145	281
8	404	222	149	282
Optimal	415	220	180	279

Table E-7: Energy production (MWh) for 6 sequence clusters

Weight	Electricity		Heat	
	GRID	CHP	Boiler	CHP
1	407	213	160	271
2	409	215	160	274
3	406	219	159	278
4	407	219	157	279
5	404	223	154	284
6	405	224	146	285
7	403	223	149	284
8	399	223	148	284
Optimal	415	220	180	279