

AMERICAN UNIVERSITY OF BEIRUT

ON THE CLUSTERING OF DEMAND AND WEATHER
DATA FOR ELECTRICITY GENERATION EXPANSION
PLANNING

by
RACHAD ZOHEIR BOU NASSER EDDINE

A thesis
submitted in partial fulfillment of the requirements
for the degree of Master of Engineering Management
to the Department of Industrial Engineering and Management
of the Maroun Semaan Faculty of Engineering and Architecture
at the American University of Beirut

Beirut, Lebanon
August 2024

AMERICAN UNIVERSITY OF BEIRUT

ON THE CLUSTERING OF DEMAND AND WEATHER
DATA FOR ELECTRICITY GENERATION EXPANSION
PLANNING

by
RACHAD ZOHEIR BOU NASSER EDDINE

Approved by:

Dr. Majd Olleik, Assistant Professor
Department of Industrial Engineering and Management

Advisor

Dr. Wael Khreich, Assistant Professor
Suliman S. Olayan School of Business

Member of Committee

Dr. Walid Nasr, Associate Professor
Suliman S. Olayan School of Business

Member of Committee

Date of thesis defense: August 23, 2024

ABSTRACT OF THE THESIS OF

Rachad Zoheir Bou Nasser Eddine for

Master of Engineering Management
Major: Engineering Management

Title: On the Clustering of Demand and Weather Data for Electricity Generation Expansion Planning

This research investigates the performance of clustering algorithms for Electricity Generation Expansion Planning (GEP) with a focus on electricity demand and weather data. The primary objective is to identify the most effective algorithm for selecting representative periods that accurately reflect the variability in energy demand and renewable energy supply, influenced by weather conditions. The study examines the impact of different clustering algorithms, including Agglomerative Hierarchical, K-means, and K-medoids, across various settings by adjusting the number of representative periods and shifting the slice of the data.

Through rigorous a priori and a posteriori analyses, the research evaluates the algorithms' performance in replicating the statistical characteristics and GEP decisions of the full dataset. It also assesses the impact of data slicing methods on the clustering outcomes. The results indicate that K-medoids algorithm stands out for its consistent accuracy in replicating the full dataset measures, making it the best performing algorithm in both a priori and a posteriori evaluation. This algorithm excels at capturing the essential statistical characteristics of the dataset, such as variance and correlations between demand and renewable energy outputs. However, it is notably susceptible to shifts in data slicing, which can significantly influence its performance. Shifts in the data slice often lead to variations in the algorithm's output, which highlights the delicate balance between data representation and the accuracy of clustering outcomes.

The findings also highlight the dependency between initial data spread, which is a data characteristic, and the shifting effect on clustering. It is shown that a data possessing higher spread indicator will experience high shifting effect on both a priori and a posteriori measures.

This study not only contributes to the theoretical understanding of clustering in GEP but also offers practical insights for energy policy and system design, emphasizing the critical role of accurate data representation in optimizing energy planning and operations. Furthermore, this study sheds light on a topic often neglected in common practices, showing that slice shifting might have considerable effect on outputs if the historical data possess a high spread indicator.

TABLE OF CONTENTS

ABSTRACT	1
ILLUSTRATIONS	4
ABBREVIATIONS	6
INTRODUCTION AND MOTIVATION	7
1.1. Overview	7
1.2. Motivation	10
LITERATURE REVIEW	11
2.1. Slicing	12
2.2. Clustering	14
2.2.1. Normalization	15
2.2.2. Assignment	16
2.2.3. Representation	18
2.3. Alternative methods for generating representative periods	18
2.4. Assessing the quality of representative periods	19
2.5. Gaps and contribution	21
METHODOLOGY	23
3.1. Data Manipulation	23
3.2. Clustering	27
3.2.1. Agglomerative Hierarchical	27

3.2.2. K-means	28
3.2.3. K-medoids.....	30
3.3. A Priori Measures Analysis	32
3.3.1. Visual Inspection	33
3.3.2. Statistical Measures	33
3.4. A Posteriori Measures Analysis.....	37
3.5. Case Studies	38
3.5.1. RTS Data.....	38
RESULTS	39
4.1. RTS1 Data Case Study	39
4.1.1. Visual Inspection	42
4.1.2. Statistical Measures	44
4.1.3. A Posteriori Measures.....	49
4.2. RTS3 Data Case Study	53
4.2.1. Visual Inspection	53
4.2.2. Statistical Measures	55
4.2.3. A Posteriori Measures.....	59
LIMITATIONS	63
CONCLUSION	64
REFERENCES	66

ILLUSTRATIONS

Figure

1. Illustrating hourly power demand and generation profiles by technology for 1 week ((adapted from (Pfenninger, 2017)).....	8
2. Showing the time step Δt and the length of a representative period T of a general slicing decision.	13
3. Illustrating the decisions involved in clustering. Adapted from (Teichgraeber and Brandt, 2019)	15
4. illustrating the experiment conducted in this research.....	23
5. illustrating the way slice shifting is occurring in hourly data dataframes.	25
6. illustrating a daily data dataframe.....	26
7. Illustrating the plots of representative days generated by Hierarchical clustering and $K=4$, for both 0-shift and 12-shift.	40
8. Serving as illustrative example of the duration curve of the 0-shift (left) and the one of the 12-shift (right).....	41
9. illustrating a comparison between representative days generated by shift 12 to those generated by shift 0, splitted among the three attributes.	43
10. Illustrating the duration curves plots of the original data and representative periods generated with 4 representative days and Kmedoids algorithm for 0-shift and 12-shift for RTS1.	44
11. Comparing the demand standard deviation of all cases against the full dataset of RTS1	46
12. Comparing the Wind duration curve distance of all cases for RTS1.....	47
13. Comparing the demand-wind correlation of all cases against the full dataset....	48
14. Displaying the values of spread indicator and shifting effect measures for RTS1 data.....	49
15. Comparing the NPV of costs for all cases against the full dataset	50
16. Illustrating the decision of the cumulative CCGT capacity added over the planning horizon, for all clustering algorithms and all K for RTS1.	51
17. Comparing Battery generation of all cases against the full dataset for RTS1. ...	52
18. Illustrating the representative periods plots generated by the 0-shift and 12-shift for the second case study.	54

19. Illustrating the duration curves plots of the original data and representative periods generated with 4 representative days and Kmedoids algorithm for 0-shift and 12-shift for RTS3	55
20. Comparing the demand standard deviation of all cases against the full dataset of RTS	56
21. Comparing the Wind duration curve distance of all cases for RTS3.....	57
22. Comparing the correlation between normalized demand and solar capacity factors of the clustered data against the original dataset of RTS3.....	58
23. Displaying the values of spread indicator and shifting effect measures for RTS3 data.....	59
24. Comparing the net present values of costs of all clustered data to the cost of full dataset for RTS3	60
25. Illustrating the decision of the cumulative CCGT capacity added over the planning horizon, for all clustering algorithms and all K for RTS3	61
26. Comparing Battery generation of all cases against the full dataset for RTS3. ...	62

ABBREVIATIONS

OM	Optimization Model
GEP	Generation Expansion Planning
CCGT	Combined Cycle Gas Turbine
OCGT	Open Cycle Gas Turbine
PV	Photovoltaics
ED	Euclidean distance
d	Ward's distance
K	Number of representative periods
CF	Capacity Factors

CHAPTER 1

INTRODUCTION AND MOTIVATION

1.1. Overview

The proportion of electricity in the final energy consumption is expected to rise steadily in the coming decades (IEA, 2022a). This trend is fueled by the progress in electrifying new sectors and the significant potential of renewable energy technologies for electricity generation. The adoption of these technologies, especially wind and solar photovoltaic (PV), is growing in global electricity systems due to significant cost reductions that have made them the most affordable electricity generation technologies (IRENA, 2022). The success of the Paris Agreement's objectives hinges on achieving nearly 100% clean electricity production by 2050 (IEA, 2022b). However, reaching this goal necessitates overcoming numerous technological and modeling obstacles, as well as challenges in policy formulation, financing, and behavioral changes (Child et al., 2019).

Electricity Generation Expansion Planning (GEP) involves determining the most cost-effective plan for installing and retiring electricity generation units to meet electricity demand over the next several decades. Given the growing importance of renewable energy technologies, whose power outputs are subject to uncertain and variable weather conditions, it's crucial for GEP models to incorporate a high temporal resolution (ideally sub-seconds) that takes into account weather variability and the operational specifics of power plants.

Figure 1 offers a visual representation of the intermittency in the generation outputs of PV and wind installations, highlighting the need to capture sub-daily details. The actual contribution of renewable energy technologies to meeting demand and the

amount of required dispatchable generation capacity (i.e., capacity that is not weather-dependent and is almost always available for activation) cannot be estimated without a high temporal representation of the system. Electricity storage technologies, which can serve as clean dispatchable technologies, also require a detailed time representation to continuously monitor their charge state and their ability to meet demand. Given these requirements, GEP models should consider sub-daily time resolutions at the level of hours or less over a multi-decade planning period (Pfenninger, 2017).

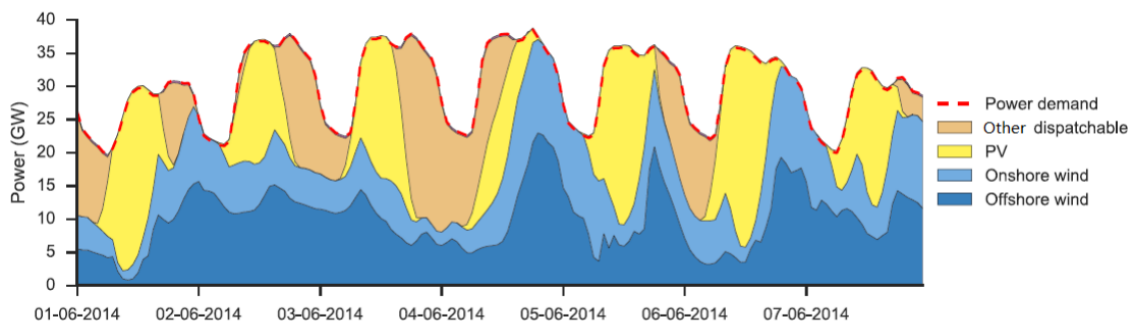


Figure 1 Illustrating hourly power demand and generation profiles by technology for 1 week ((adapted from (Pfenninger, 2017)))

GEP models are typically represented, in their basic deterministic form, as mixed integer linear optimization models. When considering the planning period and the required temporal resolution, GEP models become intractable with today's computational power (Li et al., 2022). Therefore, it's common in the literature to run these models and make operational and investment decisions based on a set of representative periods rather than every hour (or sub-hour) across the planning horizon (Hoffmann et al., 2020)

Each representative period in the Electricity Generation Expansion Planning (GEP) model includes high-resolution data related to electricity demand and renewable energy capacity factors. However, this simplification introduces several challenges. The

first challenge arises from the aggregation of periods into a representative one, which introduces statistical errors that can affect optimization outcomes. This relationship between statistical and optimization outcomes is nonlinear, indicating the complexity of the process. Therefore, the use of representative periods in optimization models, which inform policy and investment decisions, requires careful management to avoid misleading results (Scott et al., 2021).

The second challenge is the lack of unified measures of error introduced by using representative periods. Success in optimization models, which guide long-term decisions, can be measured in various ways, depending on the specific research or investment question asked (Teichgraeber and Brandt, 2022).

The third challenge involves the modeler's decisions on the amount of time series data to use, the length of the time step, and the parameters for representative periods. These decisions, which include the length and number of representative periods and the selection of important data features, impact the model's performance, fidelity, and the effectiveness of Time Series Aggregation (TSA). These choices involve trade-offs between solution time, memory requirements, and accuracy.

The fourth challenge is related to the selection of the clustering method. This selection is critical as it influences the resulting representative periods and is often a complex, nonlinear process. The clustering method must be chosen carefully to ensure that it retains important data characteristics and identifies extreme periods.

The fifth challenge concerns the assumption that operational interactions between periods can be neglected when slicing time-series data into original periods. In reality, a representative period substitutes several data points present in the cluster, and the interrelation between these data points is lost. This assumption may not always hold true,

as seen in cases like seasonal energy storage or renewable energy droughts. To address this, methodologies allowing for the linking or coupling of periods need to be developed. These challenges highlight the complexity and intricacies involved in the process of time series aggregation. (Teichgraeber and Brandt, 2022; Scott et al., 2021; Seljom et al., 2021).

1.2. Motivation

This research and implemented experiment aim to identify the most effective representative period selection algorithm while examining the impact of modifying the data slicing method. Specifically, the study explores how shifting the data slices by increments of one hour—ranging from a 1-hour shift to a full 24-hour shift—affects the clustering algorithm's results. The rationale is to investigate how these shifts might influence the selection process, as existing literature focuses on clustering techniques without considering the effects of varying data slicing methods. This research seeks to fill that gap by examining the interplay between different data slicing shifts and the outcomes of selecting representative periods, thereby offering a more comprehensive understanding of their impact on the optimization process.

CHAPTER 2

LITERATURE REVIEW

In this section, a comprehensive literature review is conducted to delve into several key areas of research.

Section 2.1 explores the prevalent methods employed in data slicing and delves into the complexities of modeling temporal resolution. This provides a foundational understanding of the current practices and methodologies in the field.

Section 2.2 focuses on the investigation of clustering, which is widely recognized as the most utilized method in generating representative periods (Teichgraeber and Brandt, 2022). Various forms and different algorithms that fall under its umbrella are examined. This allows for an understanding of its versatility and applicability in different scenarios.

In Section 2.3, the scope is broadened to investigate alternative methods for generating representative periods, beyond the realm of clustering. This enables a comparison and contrast of different approaches, and an assessment of their strengths and weaknesses.

Finally, in Section 2.4, the exploration of evaluation metrics and methods used to assess the quality and effectiveness of the generated representative periods is undertaken.

Through this structured approach, a holistic understanding of the field is gained, which informs and guides research endeavors.

2.1. Slicing

Energy system models necessitate critical decisions regarding the volume of data to be utilized and the length of the time step, Δt . Historical time series are employed, either implicitly or explicitly, under the assumption that future conditions will mirror past ones. While a single year of historical data is a common choice (Kotzur et al., 2018), some studies advocate for the use of multiple years (Nahmmacher et al., 2016) (Fazlollahi et al., 2014). This is particularly important when employing Time Series Aggregation (TSA), as it is argued that a single year of input data may not sufficiently capture the probability distribution of wind and solar availability (Pfenninger, 2017).

The time step length, Δt , is the smallest unit at which temporal data is used in the full model. In most of the studies reviewed, Δt is commonly chosen to be one hour. However, some studies also consider time step lengths of two hours (Brodrick et al., 2014), three hours (Reichenberg et al., 2018), or 24 hours (Bahl et al., 2018).

For TSA, additional decisions must be made regarding the length T of the periods, the number of representative periods K , and what features to use to select these periods.

The length of a representative period, defined by the number of time steps T , is contingent on the specific optimization problem and its input data. Commonly, day-long hourly periods are used for problems with coupling constraints (the model considers constraints that link decisions across all 24 hours of a day, $\Delta t=1, T=24$), while single time step periods ($\Delta t=1, T=1$) are used for models without such constraints (Bahl et al., 2017). It is concluded that using $T=24$ hours is superior to other alternatives (Teichgraeber and Brandt, 2022).

Some innovative approaches aimed at reducing the number of time steps within each period were provided. As an example, (Fazlollahi et al., 2014) and (Bahl et al., 2018)

proposed to aggregate within the period so that each of them consists of $S \leq T$ segments. This approach can lead to reducing computational time without significant performance deterioration.

Figure 2 below is an illustrative example used to better understand the time step Δt and the length of a representative period T of a general slicing decision:

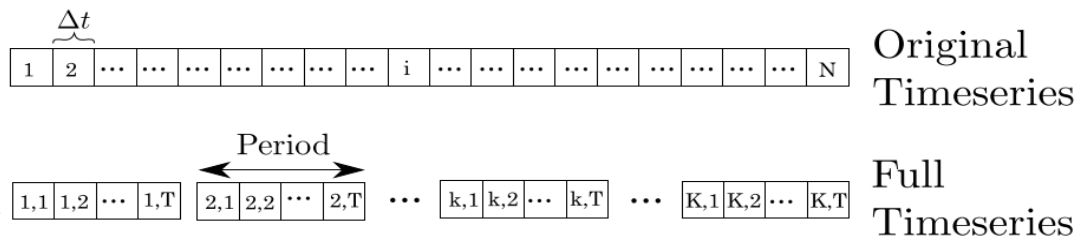


Figure 2 Showing the time step Δt and the length of a representative period T of a general slicing decision.

Other studies focused on how to identify the best number of representative periods K , which strongly influences computational complexity. A common approach to decide on the number of periods is using the ‘elbow method’, where the number of periods is plotted vs. a metric of interest (Thorndike, 1953). Typical numbers of K usually range from 5 to 30 representative periods.

Finally, determining which features within the data are most significant to represent during Time Series Aggregation (TSA) is crucial. Typically, the original time series data serves as features. However, several researchers have utilized secondary features derived from the original time series data. These include demand ramps (Green et al., 2014), price differences between nodes (Agapoff et al., 2015), and moments (Fitiwi et al., 2015). Additionally, hybrid features have been introduced where generation, demand, and price are multiplied, and principal component analysis is employed to

automatically extract prominent features of the data for use in TSA (Almaimouni et al., 2018).

2.2. Clustering

Clustering involves making a series of decisions. A framework has been devised by (Teichgraeber and Brandt, 2019) that represents these decisions, and is illustrated in figure 3 below.

There are three critical stages in the decision-making process of clustering. The first stage involves the use of normalization, which includes the operation of normalization and its scope. The second stage is the method of assignment, which considers the distance measure, the algorithm used for clustering, and the selection of the center. The final stage is the representation methods, which include the strategies for picking the representative periods and assigning weights to each period.

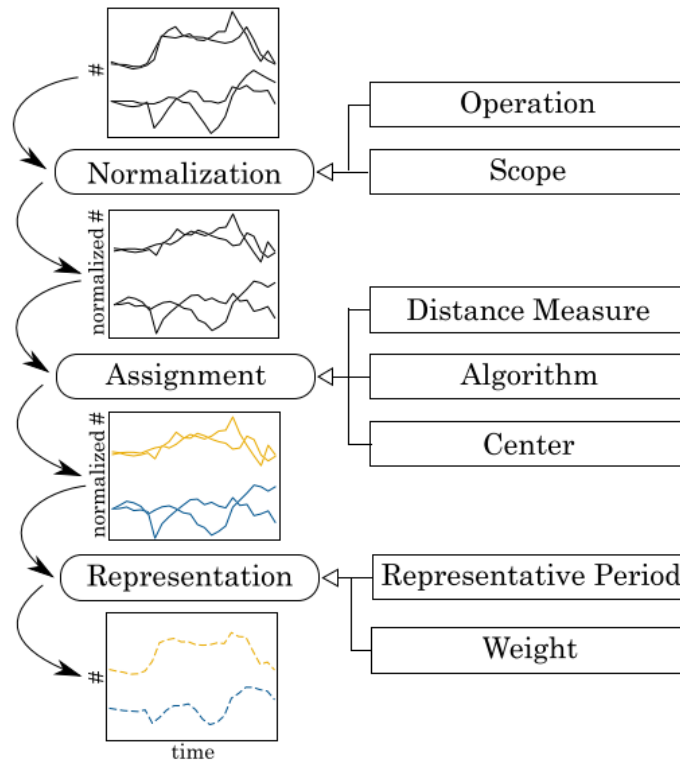


Figure 3 Illustrating the decisions involved in clustering. Adapted from (Teichgraber and Brandt, 2019)

2.2.1. Normalization

Normalization proves to be beneficial when dealing with data that encompasses diverse attributes, such as demand articulated in MW (megawatts) and capacity factors of wind and solar energy, which are typically numerical values within the range 0 to 1. The application of normalization harmonizes the units and mitigates the biases that arise due to the disparity between the large numerical values of demand attributes and the smaller numerical values of capacity factors. This ensures a more balanced and fair comparison across different attributes.

There are three prevalent procedures for normalization. The first procedure does not involve any normalization. The second procedure normalizes the data by its maximum

value. The third procedure, often known as z -normalization, adjusts the data so that it has a mean (μ) of 0 and a standard deviation (σ) of 1 (Teichgraeber and Brandt, 2019).

Normalization can be performed on three different levels. The most frequently used is the full normalization scope, where a single operation is used to normalize the entire time series. This involves using one maximum or one mean and standard deviation per attribute to normalize the entire time series for that attribute. The second level is the element-based normalization scope, where each time step is normalized independently across all periods by calculating the normalization operation for each data element (for example, hour) and attribute. The third level is the sequence-based normalization scope, also referred to as period-based normalization scope. Here, each period is normalized independently, yielding 365 values of μ and σ per attribute for a year's worth of data with daily periods (Teichgraeber and Brandt, 2019).

Once the representative periods are generated, the normalization must be reversed by denormalization so that the representative periods are expressed in the original data units and magnitude.

2.2.2. Assignment

Every clustering method needs a way to measure distance to determine how well an observation (ex: a day worth of data) fits into a cluster. Next, clustering methods need to employ an algorithm to allocate observations to clusters. Finally, the choice of cluster center in clustering methods can differ, with the primary options being the centroid or medoid of the cluster.

Distance measures compute the difference between two time series vectors. The most common distance measure is the Euclidean distance (Teichgraber and Brandt, 2022) presented below:

$$ED(x, y) = \sqrt{\sum_{t \in T} (x_t - y_t)^2}$$

Other studies used Manhattan distance, Dynamic Time Wrapping (Sarajpoor et al., 2021) and Shape-based Distance (Paparrizos et al., 2015). (Mallapragada et al., 2018) showed that the distance measure used will not affect the optimization outcome.

Once the distance measure is established, the next step is to choose a clustering algorithm. These algorithms can be either partitional or hierarchical. Widely used partitional algorithms are extensions of the k-means algorithm (Lloyd, 1982). The process begins with random initialization of cluster centers. The algorithm then iteratively performs two steps until it either converges (meaning there are no changes in cluster assignments) or reaches a predetermined maximum number of iterations. In the first step, each period vector is allocated to the nearest cluster center based on the chosen distance measure. Subsequently, the cluster centers are updated to reflect the changes in cluster assignments, aiming to minimize the distance within the cluster. The choice of cluster centers can be either the centroid (resulting in the K-means algorithm) or medoid (resulting in the K-medoids algorithm). The centroid is a hypothetical period that minimizes the within-cluster distance, while the medoid is an actual period that achieves the same goal. The outcomes from partitional methods typically converge locally and are dependent on the random initializations of clusters (Arthur et al., 2007). It's worth noting that the k-medoids clustering problem, initially solved with the partitional clustering algorithm, can also be formulated as a Binary Integer Program (BIP) as discussed in (Li

et al., 2022). Hierarchical clustering differs from partitional clustering in that it begins with a number of clusters equal to the number of data points, then combines the two nearest observations into a single cluster. Therefore, with each iteration, the number of clusters decreases by one. This process of merging two clusters continues until the desired number of clusters, K , is reached. Ward's algorithm is used to minimize the total variance within the cluster. Hierarchical clustering is deterministic and thus reproducible. It is also less computationally demanding than partitional clustering. However, this algorithm produces local solutions with sets of cluster assignments that may not necessarily be optimal (Teichgraeber and Brandt, 2022).

2.2.3. Representation

After the clusters are generated, each cluster is to be represented by either its centroid or medoid in the optimization model. The representation is accompanied by weights for each cluster, representing the proportion of the initial data assigned to the cluster by the clustering algorithm.

2.3. Alternative methods for generating representative periods

There are numerous methods besides clustering that can be employed to generate representative periods. For instance, (Almaimouni, 2018) utilized Principal Component Analysis (PCA) to identify pertinent features within the data before applying clustering. The concept behind PCA is to depict the data in a lower-dimensional space by decreasing the number of features that characterize the data. It's important to note that this doesn't reduce the number of periods. In a similar vein, (Sun et al., 2019) employed Laplacian Eigenmaps to identify relevant features before clustering. (Poncelet et al., 2016)

introduced another method for selecting representative periods: an optimization-based duration curve approximation. They minimized a self-defined statistical error measure that approximates the load duration curve of each attribute, and also incorporated ramping information by considering the ramp load duration curves of each attribute. Furthermore, (Zatti et al., 2019) proposed another optimization-based approach. They identified representative and extreme days through the formulation of a mixed-integer linear program and bounded the solution through a maximum deviation tolerance in terms of the load duration curve. Lastly, several sampling-based approaches have been suggested. (Hilbers et al., 2019) proposed using importance subsampling. In this method, random sampling weighted by an iteratively calculated importance based on the optimization problem itself is used to identify time steps that are used for system design. The authors demonstrated good performance based on a system with 36 years of input data.

2.4. Assessing the quality of representative periods

In literature, two primary types of performance measures are typically employed to assess the quality of representative periods: a priori indicators, which are mainly statistical error metrics that measure the representation error between the representative periods and the full data set, and a posteriori indicators, which measure the error in the output of the GEP model (Hoffmann et al., 2021).

The most frequently used a priori indicator is the Normalized Root Mean Square Error (NRMSE), which measures the total error between the representative period of the cluster and the cluster members for all representative periods (Nahmacher et al., 2016). This gives an indication of the quality of the clustering and the selection of cluster representatives. Other indicators such as the silhouette statistic (Adhau et al., 2015) and

gap statistic (Tibshirani et al., 2001) are also available. Additionally, visual inspection is commonly used. This involves plotting each period of the full data and the representative periods and their respective weights (Teichgraeber et al., 2017). Visual inspection provides a rough understanding of which features of the data are well captured by the TSA method used. Visual inspection of duration curves is also utilized. In a duration curve, the data are sorted and plotted from the highest to the lowest value. Load duration curves have traditionally been used for electricity generation planning where the duration curve of each parameter (demand, solar and wind) is plotted separately (Blanford et al., 2016). (Buchholz et al., 2020) demonstrated that a reasonable fit in duration curves may still introduce error in model output. Overall, statistical error metrics or visually inspecting the data alone are not adequate (Teichgraeber and Brandt, 2022).

The ultimate theoretical a posteriori error measures assess the difference in the results of the GEP model when run using the representative periods versus when run using the original full data set. In practice, it's not feasible to run the GEP model on the full data set. Researchers have adopted a two-step practical approach to measure a posteriori errors (Scott et al., 2019): Initially, the GEP model is run using the representative periods and the investment schedule is recorded. Then, the resulting investment schedule is assumed to be fixed and the GEP model is run again, this time on the entire data set, but is only allowed to make operational decisions in terms of scheduling the runtime of power plants. This new operational model (GEP model but with exogenous investment decisions) is feasible when the entire data set is used. The operational results of the GEP model run with the representative periods are then compared against the operational results after running the operational model. Previous research has shown that there is no guaranteed correlation between the quality of the a priori and a posteriori indicators

(Teichgraeber and Brandt, 2022). A significant limitation of a posteriori measures is the assumption that a complete data set is available to run the operational model and generate comparative results.

In conclusion, it is necessary to include relevant optimization outcomes when measuring the success of TSA, and that statistical metrics alone are not sufficient (Sun et al., 2019).

2.5. Gaps and contribution

While the existing literature provides a wealth of information on this topic, there is a noticeable gap in large-scale comparative studies on various clustering techniques and their performance as the number of representative periods, denoted as K , varies.

Furthermore, while there is a consensus in the literature that there is no direct correlation between statistical measures and optimization outputs, there has been limited exploration to further investigate this issue. Most importantly, no research has been conducted to examine how alterations in slicing techniques might impact the output of selecting representative periods, despite the well-acknowledged significant role that slicing plays in clustering algorithms. This is a critical area of investigation as the method of slicing can significantly influence the performance and results of clustering algorithms.

This research aims to address the identified gaps with a primary focus on determining the most effective representative periods selection algorithm under diverse conditions. This is achieved by conducting an in-depth comparative analysis of various clustering algorithms. This analysis involves:

- 1- Varying the number of representative periods ($K=4,8,16$ and 32).
- 2- Adjusting and varying the time slice by shifting it.

- 3- Employing multiple datasets to check the generalization of the findings.
- 4- Offering a more profound understanding and interpretation of representative periods selection evaluation measures.

The main contribution of this research lies in highlighting a potential issue that could impact clustering and GEP model outcomes: the slice shifting of data. To illustrate this problem, the study introduces two new a priori measures: the spread indicator and the shifting effect. The findings reveal that datasets with a higher spread indicator are more likely to experience significant shifting effects. This suggests that GEP model outcomes may vary with each shift, raising concerns about the robustness of the results. A lack of robustness in GEP model outcomes poses a serious issue, as it could lead to incorrect decisions and result in suboptimal resource usage.

CHAPTER 3

METHODOLOGY

The approach of this research involves carrying out a comprehensive experiment to evaluate various clustering algorithms and understand the impact of slice shifting on their results. These algorithms will be examined using a range of representative periods to determine how changes in this number influence the overall performance of the clustering. The experiment is depicted in Figure 4 below to provide a visual overview before we delve deeper into the specifics in the subsequent sections.

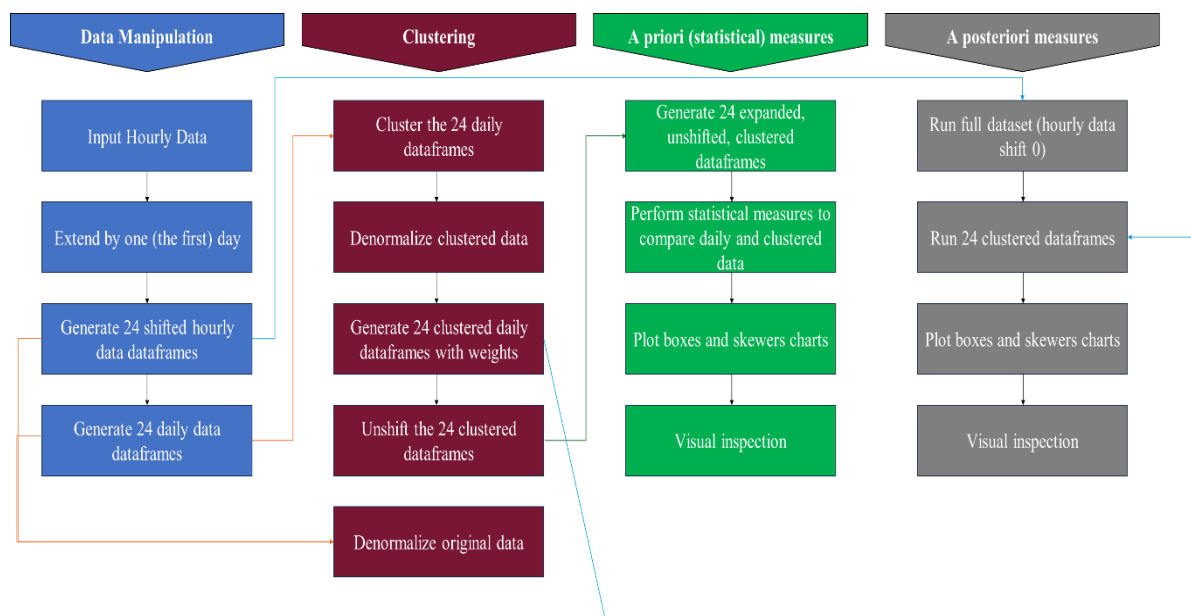


Figure 4 illustrating the experiment conducted in this research.

3.1. Data Manipulation

The process begins by inputting a year’s worth of raw data, which is then transformed into an hourly data dataframe. This dataframe consists of 8760 rows, each

representing an hour within the year, and three columns, each representing an attribute of interest: electricity demand load, solar PV capacity factors, and onshore wind turbine capacity factors.

Following the data input, normalization by maximum is performed. This involves dividing each value in a column by the maximum value of that column, effectively unifying the three attributes. The values now range from 0 to 1, which helps eliminate any potential biases in the clustering algorithm that might arise from larger numbers.

Once the initial hourly data dataframe is generated, it is expanded by 24 hours. This is achieved by copying the first 24 hours (representing the first day) of the initial hourly data dataframe and appending them to the end of the dataframe. This step facilitates the slice shifting of the data, where 24 new hourly data dataframes are generated, each shifted by just one hour from its predecessor.

This method of shifting is designed to isolate the effect of one-hour shifts on the data values selected. While the numerical values of each dataframe are preserved, only the start and end of each 24-hour interval (representing one day) are altered. The resulting dataframes are depicted in Figure 5 below for visual representation, with a focus on illustrating how the 24 hours of the final day (day 365) differ between shift 0 and shift 12.

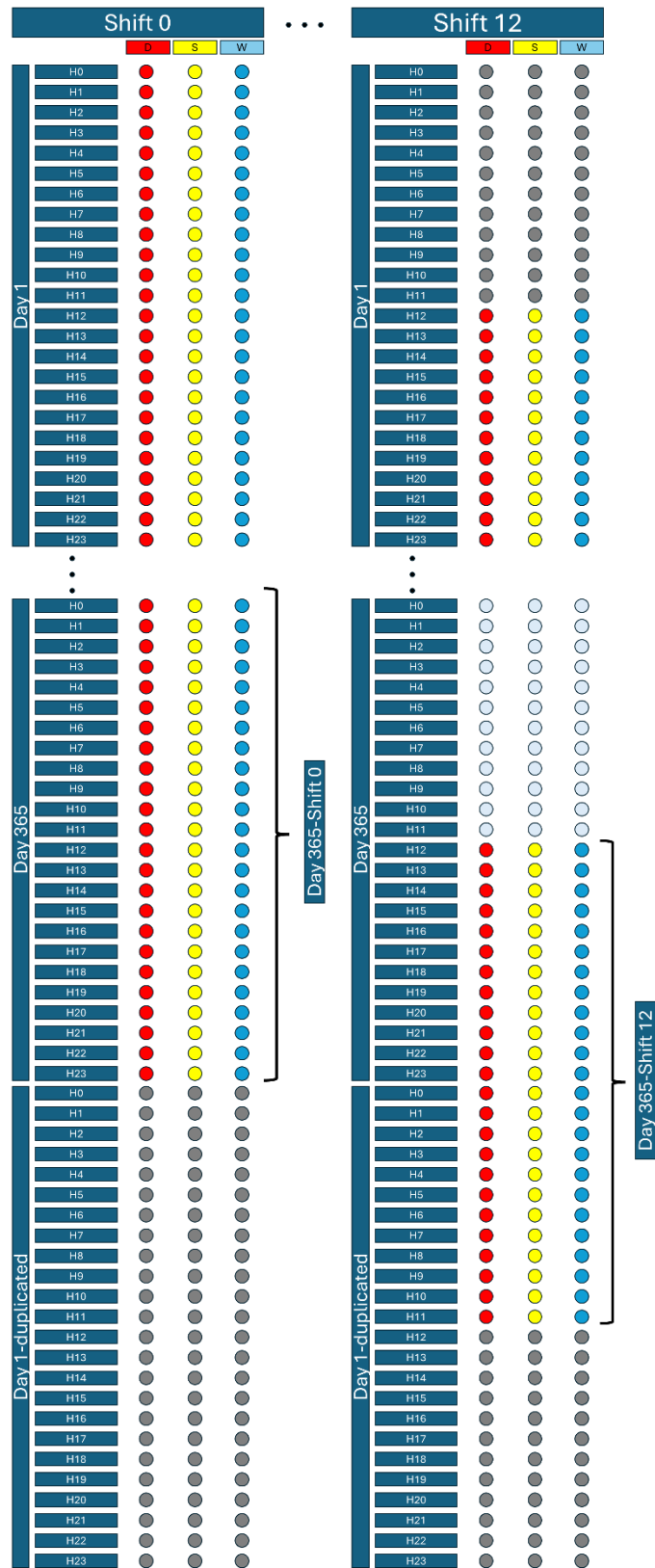


Figure 5 illustrating the way slice shifting is occurring in hourly data dataframes.

After all the hourly data dataframes are generated, each one is then transformed into a daily data dataframe. This process starts with the hourly data for each attribute, which is organized in columns. Each column represents an attribute, and each row represents an hour, with 24 rows accounting for each day. The first step is to group these hourly data points into daily chunks. For each attribute, the 24-hourly data points are combined together to form a single row. This means that each row now holds a full day's worth of data for that attribute, with each data point corresponding to a different hour of the day. This process is repeated for each attribute. As a result, the first 24 columns of the resulting dataframe represent 24 hours of normalized energy demand, the next 24 columns represent 24 hours of normalized solar capacity factors (CF), and the final 24 columns represent 24 hours of normalized wind CF. In the end, the resulting dataframe has 365 rows and 72 columns. Each row represents a day, and each set of 24 columns represents a full day's worth of data for a specific attribute. Figure 6 below provides a visual representation of a typical daily data dataframe.

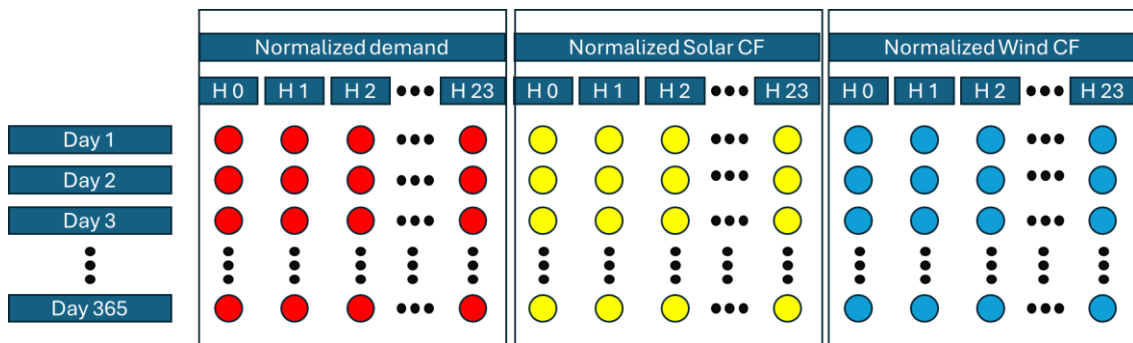


Figure 6 illustrating a daily data dataframe.

3.2. Clustering

Upon completion of the data manipulation process, 24 normalized daily data dataframes are generated. These dataframes are then subjected to clustering using three well-known algorithms: Agglomerative Hierarchical, K-means, and K-medoids. The aim is to compare the performance of these algorithms.

To assess the impact of the number of representative days (K) on each clustering algorithm, the clustering process is executed four times for each algorithm: once for K=4, once for K=8, once for K=16, and once for K=32. This results in a total of 25x3x4 clustering tasks for each dataset input.

The purpose of this extensive analysis is to compare the effects of slice shifting and the number of representative periods on the clustering algorithm, and ultimately, to determine the most effective clustering algorithm.

3.2.1. *Agglomerative Hierarchical*

Agglomerative Hierarchical Clustering is a type of clustering algorithm that builds a hierarchy of clusters by merging together smaller clusters. The process starts by treating each data point as a single cluster. Then, it repeatedly identifies the two clusters that are closest together and merges them. This iterative process continues until all the clusters are merged together into one or until it meets a termination condition (in our case, when the number of clusters reach K assigned).

In terms of assigning data to clusters, the algorithm uses the Ward's distance measure, which is widely chosen in hierarchical clustering. Ward's method is a criterion applied in statistics that aims to minimize the total within-cluster variance. In general, the Ward's minimum variance criterion minimizes the total within-cluster variance. To

implement this method, at each step, the pair of clusters with minimum between-cluster distance are merged. Mathematically, the Ward's distance between two clusters is given by the formula:

$$d(u, v) = \sqrt{\frac{|v|+|s|}{T}d(v, s)^2 + \frac{|v|+|t|}{T}d(v, t)^2 - \frac{|v|}{T}d(s, t)^2}$$

Where u is the newly joined cluster consisting of clusters s and t , v is an unused cluster in the forest, $T=|v|+|s|+|t|$, and $d(u, v)$ is the Euclidean distance between clusters u and v .

The representative period of each cluster (the centroid) is calculated as the mean value of the points within the cluster. It serves as the center of the cluster and will be used to represent the cluster in computations and visualizations.

As for the pros and cons of this algorithm, on the positive side, Agglomerative Hierarchical Clustering provides a beautiful tree-based structure for visualizing the clusters, which is useful for understanding the data. Moreover, it is reproducible, meaning that if the process is repeated many times, it will yield similar results. However, on the downside, the algorithm isn't suitable for large datasets due to its high time complexity. It also has difficulty handling clusters of different sizes and densities. Additionally, it's sensitive to the choice of the linkage method used to calculate the distance between clusters. Once a decision is made to combine two clusters, it cannot be undone, which may lead to premature decisions. Lastly, this clustering algorithm does not necessarily generate globally optimal solutions.

3.2.2. *K-means*

K-means is a type of partitioning clustering that separates the data into K non-overlapping subsets (or clusters) without any cluster-internal structure. The way it works is by assigning each data point to the nearest centroid, where centroids are the centers of

the clusters. Each centroid of a cluster is a collection of feature values which define the resulting groups.

The process starts by randomly initializing these centroids. Then, each data point is assigned to the cluster with the nearest centroid. After all data points have been assigned to clusters, the centroids are recalculated as the mean of all data points in the cluster. This process is repeated until the centroids do not change significantly, indicating that the algorithm has converged. In this research, the algorithm is run multiple times with different initial centroid seeds - specifically, 100 different seeds. The best output is selected from these runs. The “best” output is defined as the one that results in the smallest total distance between each data point and its assigned centroid. This total distance is also known as the inertia.

The distance between a data point and a centroid is typically measured using the Euclidean distance, which for two points x and y in n -dimensional space is calculated as:

$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$

The representative period of each cluster in K-means is the centroid of the cluster, which is the arithmetic mean of all the data points in that cluster.

As for the pros and cons of K-means, on the positive side, it’s simple to understand and easy to implement. It’s efficient in terms of computational cost, making it a good choice for large datasets. It’s also guaranteed to converge, and with a high number of different initial centroid seeds, it’s likely to find a good solution. On the downside, K-means assumes that clusters are convex and isotropic, which is not always the case. It’s also sensitive to the initial selection of centroids and can fall into local minima, although running the algorithm with many different initial centroid seeds can

mitigate this. Another limitation is that it requires the number of clusters to be specified beforehand. Finally, like Agglomerative Hierarchical Clustering, K-means does not necessarily yield a globally optimal solution.

3.2.3. *K-medoids*

K-medoids is a type of partitioning clustering that is similar to K-means. However, instead of using centroids as the representative days, it employs medoids. A centroid is a mean value that may not necessarily be an actual data point in the dataset, while a medoid is an actual data point within the dataset that minimizes the sum of distances to other points in its cluster.

In this research, an optimization-based approach for K-medoids as proposed by (Li et al., 2022) is used. This proposition involves using mixed-integer linear programming (MILP) to achieve global optimality—a trait not commonly found in other clustering algorithms.

The MILP formulation is described as follows:

The objective is to minimize the sum of distances from each point to its medoid, mathematically represented as:

$$\text{Minimize } \sum d_{ij} z_{ij}$$

Where d_{ij} represents the distance between data point (i) and medoid (j) and z_{ij} is a binary variable that equals 1 if data point (i) is in the cluster with medoid (j), and 0 otherwise.

Subject to the constraint that each point is assigned to exactly one medoid:

$$\sum z_{ij} = 1 \forall i = 1, 2, \dots, n$$

And the binary variable z_{ij} is forced to be 0 if point (j) is not a medoid, where y_j is another binary variable set to 1 if point (j) is a medoid, 0 otherwise:

$$z_{ij} \leq y_j \forall i, j = 1, 2, \dots, n$$

Finally, the number of medoids is set to the number of representative periods (K) by the following:

$$\sum y_j = k$$

As for the pros and cons of the optimization-based K-medoids, on the positive side, it can result in a globally optimal solution. This means that the solution found is the best possible solution and not just a locally optimal one. However, on the downside, this approach can be computationally expensive and time-consuming. This might make it less suitable for very large datasets or for applications where computational resources or time are limited.

After the clustering is performed, four clustered dataframes are generated from each daily data dataframe, where the numbers of columns are always equal to 72, but the number of rows will vary depending on the number of representative days set (4,8,16 or 32). Each row of the clustered dataframes represents 1 representative period chosen depending on the clustering algorithm used. Furthermore, weights for each representative period are also generated, representing the number of initial datapoints stored in the cluster represented by the row of the clustered dataframe.

The next step is to denormalize the data, and this includes the clustered data and the original daily and hourly data dataframes. The denormalization is done by simply reversing the normalization process, aka by multiplying each attribute by the maximum value stored when performing normalization.

Finally, the resulting clustered dataframes, which have been denormalized, are stored by exporting them to Excel. This is done because they serve as inputs for the GEP optimization model. After exporting, an additional adjustment is made to the data,

specifically, the clustered dataframes are ‘unshifted’. This step is necessary because the shifting process distorts the chronology of the data. Such distortion can impact the analysis of statistical measures, particularly correlations, given that these time series are highly dependent on time. Therefore, to preserve the integrity of the data, the shifting that was initially applied to the datasets is reversed on the clustered dataframes. This ensures that each row of the clustered dataframes starts by hour 0 and finishes at hour 23.

3.3. A Priori Measures Analysis

This step involves carrying out visual checks and statistical analysis to evaluate the accuracy of the clustering algorithms used and to examine the effect of shifting. The specific tests and analysis performed will be detailed in the following sub-sections. Most of the tests compare the measures of the unshifted, clustered data to those of the initial daily data dataframe (the one with 0 shift).

One challenge in this analysis is that the amount of data in the clustered dataframes (4,8,16 or 32 data points) is less than the amount of data in the initial daily data dataframe (365 data points). To address this, The clustered data is used to create a dataset identical in size to the original by replacing each day in the original dataset with its corresponding representative day. This results in clustered dataframes with 365 rows of duplicated data points according to the clustering assignment. To provide a boundary for the analysis, 25 dataframes containing K random days (4,8,16 and 32) are also selected from each initial daily data dataframe and expanded according to approximately equal weights (sum of 365). This approach offers an additional perspective by comparing the clustered data not only to the initial dataset (representing the absolute truth), but also to

randomly selected days. This comparison helps to gauge how much better these clustering algorithms are than random sampling (which represents, theoretically, the worst-case scenario).

3.3.1. Visual Inspection

Two types of visual inspections are conducted for each clustering algorithm and each number of representative periods. The first involves plotting the data points of the clustered dataframe with no shift against the data points of the other 24 clustered dataframes that have been unshifted. This offers a visual representation of how shifting impacts the clustering algorithm. To gain specific insights for each attribute, these plots are divided into three sub-plots.

The second visual inspection involves plotting the duration curve of the initial daily data dataframe against each of the 25 unshifted, clustered, and expanded dataframes. This comparison provides insights into how well the clustered data represents the initial data and how the shifting process influences this representation.

3.3.2. Statistical Measures

3.3.2.1. Mean

The first statistical measure analysis performed is the mean. The mean of each attribute for the initial daily data dataframe, the 10 expanded random sample dataframes and the 24 expanded, unshifted clustered data dataframes is calculated according to the following formula:

$$\mu = \frac{1}{365} \sum_{i=1}^{365} x_i$$

Where x_i represents a day with all its attributes (demand, solar, wind). The resulting data is then exported to excel and plotted in boxes and whiskers charts to compare how accurately the clustering algorithms capture the mean of the original daily dataset and how shifting varies this accuracy. This is repeated for every clustering algorithm and for each number of representative periods.

3.3.2.2. Variance

Similar to section 3.3.2.1, the variance of each dataframe is calculated using the following formula:

$$\sigma^2 = \frac{1}{365} \sum_{i=1}^{365} (x_i - \mu)^2$$

Where x_i represents a day with all its attributes (demand, solar, wind). The resulting data is also exported to excel and plotted in boxes and skewers charts to compare how accurately the clustering algorithms capture the variance of the original daily dataset and how shifting varies this accuracy. This is repeated for every clustering algorithm and for each number of representative periods. Finally, the value is square rooted in order to get the standard deviation.

3.3.2.3. Correlation

Similar to the above sections, 3 correlations are calculated. First the correlation of demand and solar data is generated, then the correlation of demand and wind data and finally the correlation of solar and wind data. The correlations are calculated following the formula below:

$$r = \frac{\sum_{i=1}^{365} (x_i - \bar{x}) \times (y_i - \bar{y})}{\sqrt{\sum_{i=1}^{365} (x_i - \bar{x})^2 \times \sum_{i=1}^{365} (y_i - \bar{y})^2}}$$

The resulting data is also exported to excel and plotted in boxes and skewers charts to compare how accurately the clustering algorithms capture the correlation of the original daily dataset and how shifting varies this accuracy. This is repeated for every clustering algorithm and for each number of representative periods.

3.3.2.4. Sum of Euclidean Distance of Duration Curves

The duration curves are plotted according to section 3.3.1 as a way for visual inspection. In effort to quantify how accurate the clustered data capture the original data, the sum of Euclidian distances between the duration curves of the initial data and all clustered data is calculated according to the formula below. The random sampling experiment is also used in this section similar to the ones mentioned previously.

$$ED = \sqrt{\sum_{i=1}^{8760} (x_i^{initial} - x_i^{clustered})^2}$$

The resulting data is also exported to excel and plotted in boxes and whisker charts to compare how accurately the clustering algorithms capture the original daily dataset and how shifting varies this accuracy. This is repeated for every clustering algorithm and for each number of representative periods.

3.3.2.5. Time Series Error

Another statistical measure assessed is the TSE. This measure evaluates the difference between the initial daily data dataframe and the expanded clustered dataframe by calculating the Euclidean distance between corresponding elements in the two dataframes. The TSE captures the error introduced by replacing the full dataset with the clustered one for temporal reduction, providing a clear indication of how accurately the clustering algorithm replicates the original dataset.

3.3.2.6. Spread Indicator

The Spread Indicator is an initial data characteristic that measures how closely the profile of each day aligns with that of its successor. As detailed in the equation below, this indicator is calculated by summing the Euclidean distances between each day and its successor, then averaging these values to obtain a single indicator for the entire dataset.

$$SI = \frac{1}{364} \sum_{i=1}^{364} dist(x_i, x_{i+1})$$

Where x_i represents a vector containing all the values of demand, solar CF and wind CF attributes corresponding to a day i . A high Spread Indicator suggests that the data is widely dispersed, while a low Spread Indicator indicates that the data follows a consistent profile pattern over extended periods of time.

3.3.2.7. Shifting Effect

This measure is introduced to quantify the effect of shifting. For each shift, the sum of Euclidean distances is calculated between its elements and those of every other shift. This approach allows us to track how shifting generates deviations from one shift to another. These values are then summed for each clustering algorithm and each number of representative periods (K) to observe how this measure varies with these parameters. The equations below represent this measure mathematically.

$$R_{ij} = \sum_{y=1}^{365} dist(x_{i,y}, x_{j,y})$$

Where $x_{(i,y)}$ corresponds to the vector containing all the attributes' elements representing the day y in shift i .

$$\text{Then, } S.E = \frac{\sum_i \sum_j R_{i,j}}{276}$$

3.4. A Posteriori Measures Analysis

Following the execution of the GEP optimization model, a posteriori analysis is conducted. This analysis specifically considers 9 distinct optimization outputs. The first output under examination is the net present value of the cost. This is followed by an examination of the total capacity added of four technologies: CCGT, Solar PV, Wind turbines and storage batteries. The GEP optimization model considers a green field, meaning that it is assumed that no capacity of the mentioned technologies existed before.

The remaining outputs under consideration are the generation of the mentioned technologies across all the planning horizon (20 years). This helps in understanding the difference between the outputs on a dispatching level, giving deeper understanding of the decisions executed.

The analysis is performed by processing all the shifted, clustered data along with the corresponding weights (for all clustering algorithms, all K) as inputs to the GEP model. The resulting outputs are then assessed against the outputs derived from running the full case, which is the original dataset prior to clustering, representing the absolute truth. The feasibility of running the full case is due to the relative simplicity of the GEP model and its capacity to handle a large volume of data.

Similar to a priori analysis, 24 random dataframes, each containing K days, are selected from the original dataset. This selection is made to obtain the theoretically worst possible solution and to bound the analysis. Subsequently, the data are exported to Excel and visualized using box and whisker plots. These plots serve to compare how decisions, based on the clustered data, deviate from the absolute truth scenario. This comparison

aids in determining the most effective clustering algorithm and in understanding the influence of shifting on these decisions.

3.5. Case Studies

3.5.1. *RTS Data*

Both datasets utilized in this study are sourced from the GridMod/RTS-GMLC repository on GitHub (RTS, 2023). The RTS, or Reliability Test System, data is a set of empirical data generated for an imaginary area in the United States. It was created specifically for energy studies and is widely recognized in the field. This dataset provides hourly data on energy demand, solar PV capacity factors, and onshore wind turbine capacity factors, which form the foundation of the experiment done in this research. The granularity of the data allows for a detailed analysis of energy patterns and trends. Furthermore, the credibility of the RTS data is well-established, as it has been used in numerous studies, thereby adding to the robustness of the research. This data is split into three zones. Zone 1 and Zone 3 are taken as distinguished datasets for this study. Zone 2 was disregarded given that it does not contain wind data.

CHAPTER 4

RESULTS

4.1. RTS1 Data Case Study

In the following sub-sections, results of the first case study will be displayed. This includes both “a priori” and “a posteriori” results of the three clustering algorithms under evaluation.

Before delving into the comparison of results, an illustrative scenario utilizing the Hierarchical clustering algorithm with $K=4$ will be addressed to facilitate a clearer comprehension of the subsequent findings.

Firstly, data slicing and manipulation are executed. At this stage, the hourly data undergoes partitioning into shifted dataframes, with each dataframe differing from its predecessor by a 1-hour shift, as explained in Section 3.1 and depicted in Figure 5. Subsequently, these dataframes are transformed into daily dataframes to serve as inputs for the clustering algorithm.

Following the execution of the hierarchical clustering algorithm in this illustrative scenario, the 365 days of initial data are simplified into 4 representative days. The clustering algorithm is applied to each shifted dataframe, resulting in 24 sets of 4 representative days each.

The comparison will be conducted across two main categories. The first category aims to evaluate the effectiveness of the clustering algorithm in replicating the statistical and "a posteriori" measures of the initial full dataset, as detailed in Sections 3.4 and 3.5. This is primarily achieved by assessing these measures for each clustering output of the shifted dataframes and exporting them to Excel for subsequent plotting of box and

whisker diagrams comparing these clustering output measures against those of the initial dataset. These diagrams serve the dual purpose as an aid in visualizing the variability induced by slice shifting, and to check how accurately can the clustering algorithm replicate the full dataset results.

The second category serves to further shed light on the effect of slice shifting and is subdivided into two types. The first type involves plotting the representative periods generated from each slice shifting against those generated from the zero shift. The horizontal axis of these plots represents time, where the 24-hour interval symbolizes a single day. The vertical axis represents the capacity factors in the case of solar and wind attributes, and the normalized demand in the case of demand attribute. A comparison between the results of a 12-hour shift and a zero-hour shift is illustrated in Figure 7 below, where the zero-shift is represented by a smooth line, and the 12-hour shift by a dotted line. Each line, irrespective of type, represents one representative day, and due to the illustrative nature of this case with $K=4$, four lines of each type are visualized.

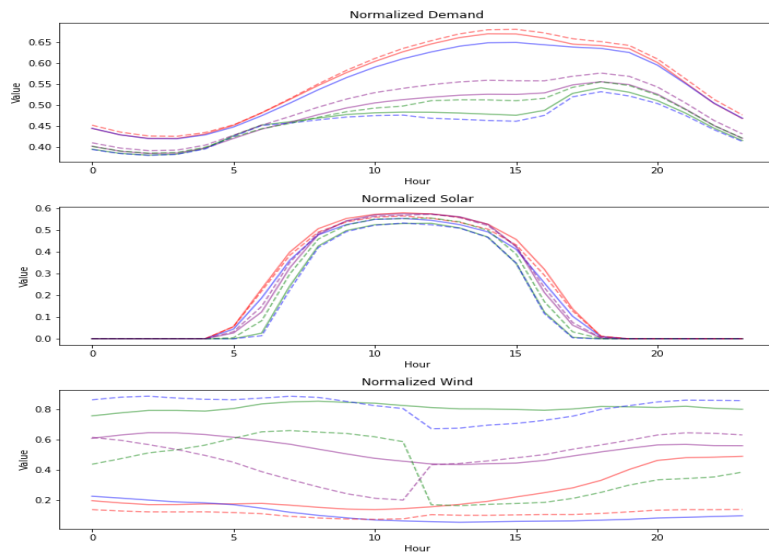


Figure 7 Illustrating the plots of representative days generated by Hierarchical clustering and $K=4$, for both 0-shift and 12-shift.

The second type of visual inspection is the duration curves plotting. This time, the clustered representative periods are expanded by multiplying each one by its corresponding weight generated from the clustering algorithm. These weights represent the number of datapoints assigned to the cluster represented by the corresponding representative period. Then, the duration curve of each shifted slice clustering output is plotted against the duration curve of the full dataset. Figure 8 below is an illustrative example of the duration curve of the 0-shift (left) and the one of the 12-shift (right).

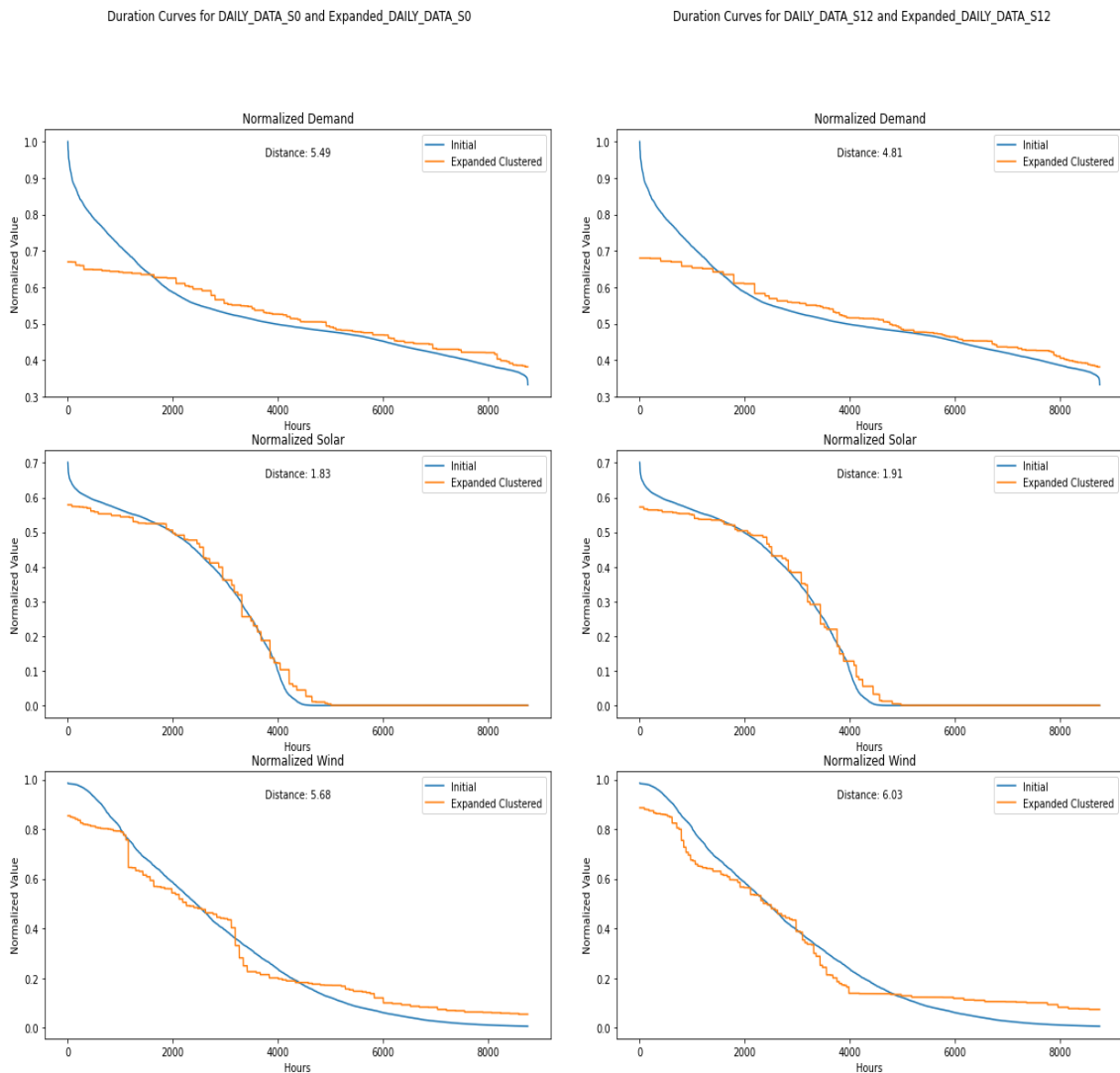


Figure 8 Serving as illustrative example of the duration curve of the 0-shift (left) and the one of the 12-shift (right).

The plots are split into the three attributes-Demand, solar and wind- to have a deeper perspective on how the shifted slicing is affecting each one. The horizontal axis also represents time, having 8760 hours representing the whole yearly data (of the 365 days). The vertical axis represents the capacity factors in the case of solar and wind attributes, and the normalized demand in the case of demand attribute.

What was explained in this subsection is repeated for every number of representative periods (4,8,16 and 32) and for all the clustering algorithms under study (Hierarchical, Kmeans and Kmedoids). The analysis and comparison of results is provided in the following subsections, each representing certain testing criteria.

4.1.1. Visual Inspection

The process of visual inspection involves the comparison of representative days generated from the shifted slices with those from the zero-shift slice. This comparison aids in determining whether slice shifting has an impact on the representative days generated. To maintain traceability, only the outcomes from the zero-shift slice (the original slice) and the 12-shift slice (the slice shifted by half a day) will be presented. Moreover, the results displayed will be limited to K values of 4 and 16, as the usage of K=32 is infrequent in existing literature and the utilization of these two K values is deemed sufficient for the validation of the findings. The graphs of the plots are captured in figure 9 below, where each colored line represents 1 representative day:

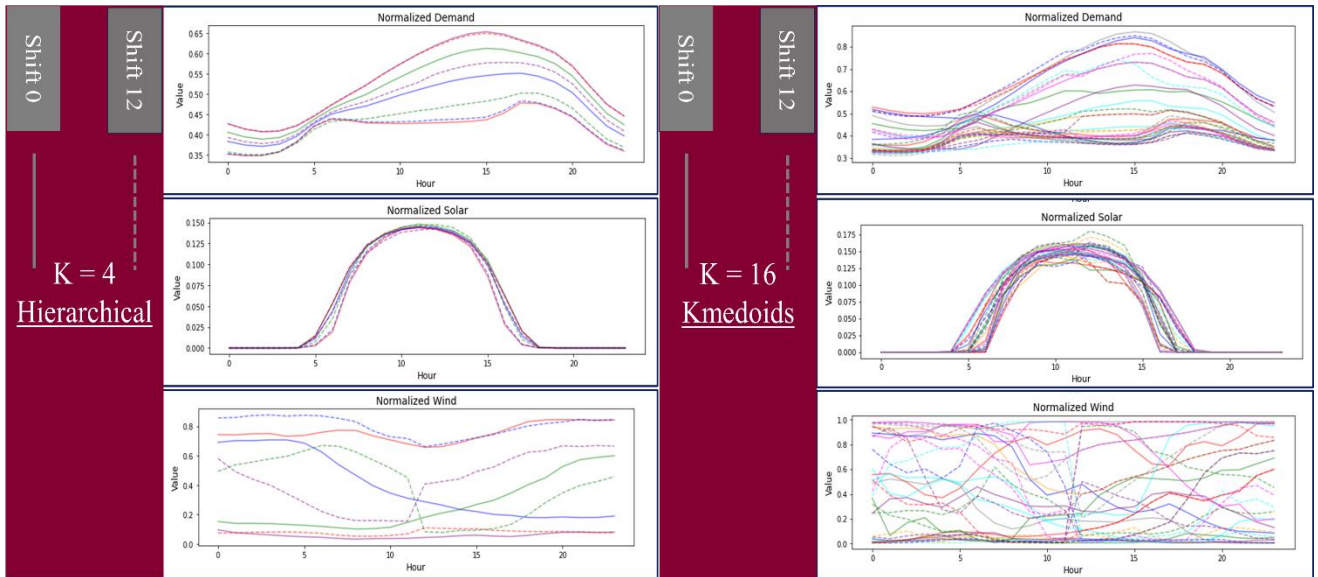


Figure 9 illustrating a comparison between representative days generated by shift 12 to those generated by shift 0, splitted among the three attributes.

While these graphs do not allow for definitive conclusions, two primary observations can be made. Firstly, the representative days produced by the shifted data (indicated by dotted lines) exhibit a slight deviation from those generated by the zero-shift (indicated by smooth lines). This deviation persists for all the clustering algorithms and even as K increases from 4 to 16, suggesting that slice shifting may indeed influence the clustering outputs. Secondly, the deviation is minimal in the case of solar data but is considerably more pronounced for wind data. This is primarily due to the high variability of wind data and its asymmetric plots, which make it more susceptible to the effects of slice shifting.

An additional visual inspection involves examining the duration curves of the generated representative periods and comparing them to the duration curves of the original data. This comparison aids in determining whether an increase in the number of representative periods enhances the capacity to capture the entirety of the data, and how slice shifting might influence this capacity. For this analysis, the duration curves for the

representative periods generated by zero-shift and 12-shift for $K=4$ in the Kmedoids clustering algorithm are depicted in Figure 10 below.

In terms of the impact of shifting, visualized in the figure below, a slight variation is observable between the duration curves of zero-shift and 12 hours-shift. This deviation is particularly noticeable in the wind duration curves, underscoring the notion that wind data is the attribute most significantly affected by slice shifting.

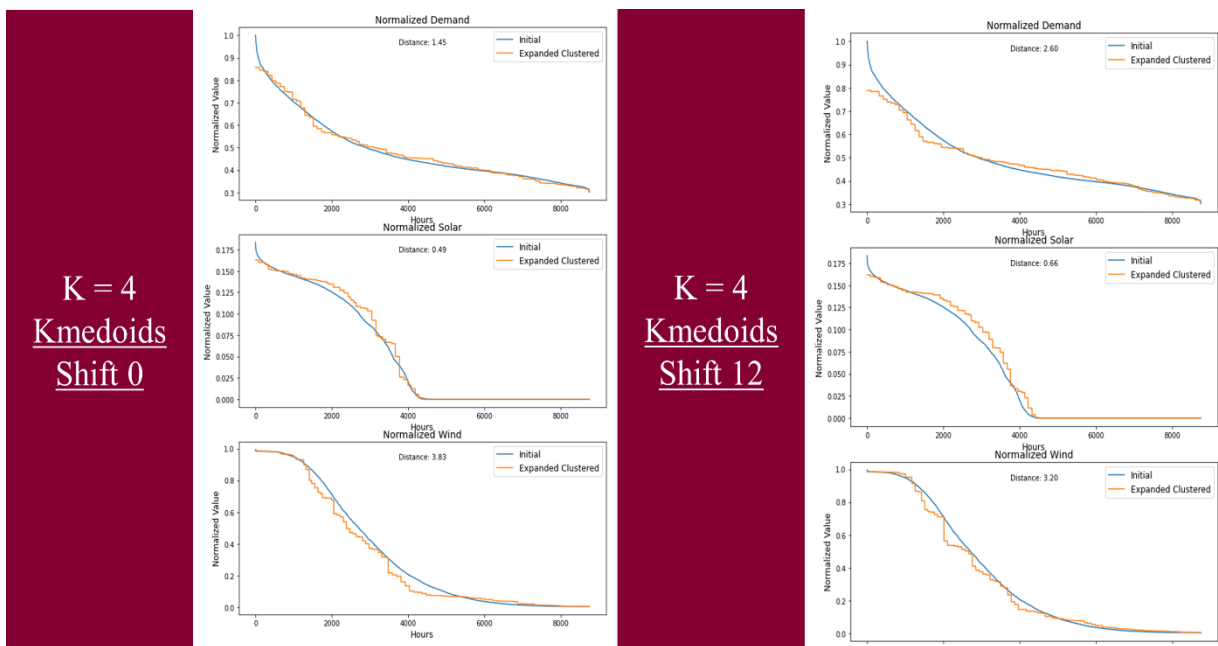


Figure 10 Illustrating the duration curves plots of the original data and representative periods generated with 4 representative days and Kmedoids algorithm for 0-shift and 12-shift for RTS1.

4.1.2. Statistical Measures

This section will present the results of various statistical measures. Specifically, a comparative analysis will be conducted between the clustered data and the full dataset, focusing on aspects such as mean, standard deviation, and correlations. A more detailed examination of the duration curves will also be undertaken, involving the calculation of the Euclidean distance between the curves of the clustered and original data. This approach facilitates the conversion of the visual inspection conducted in the previous

section into a numerical analysis. Accommodations for the effect of shifting will be made by representing the relevant measures in box and whisker plots, aiding in the visual identification of deviations caused by the 24 shifts. While these statistical measures are performed for each attribute - demand, solar, and wind - only selected results of certain measures will be displayed for traceability purposes.

4.1.2.1. Demand Standard Deviation

As illustrated in Figure 11, the K-medoids algorithm consistently outperforms other clustering algorithms in capturing the demand standard deviation across all values of K, demonstrating its superiority in this context. Notably, as K increases from 4 to 32, the standard deviation values gradually converge toward those of the full dataset. This outcome aligns with established findings in the literature, as a higher K value allows for a more accurate representation of the underlying data dynamics by accommodating a greater number of representative periods.

Additionally, the analysis reveals that the effect of shifting—represented by the width of the boxes in the figure—diminishes as K increases. This trend suggests that the impact of shifting becomes less significant at higher K values, indicating a more stable clustering performance and reduced sensitivity to temporal variations. This behavior further underscores the advantage of using a larger number of clusters to capture the intricacies of the dataset.

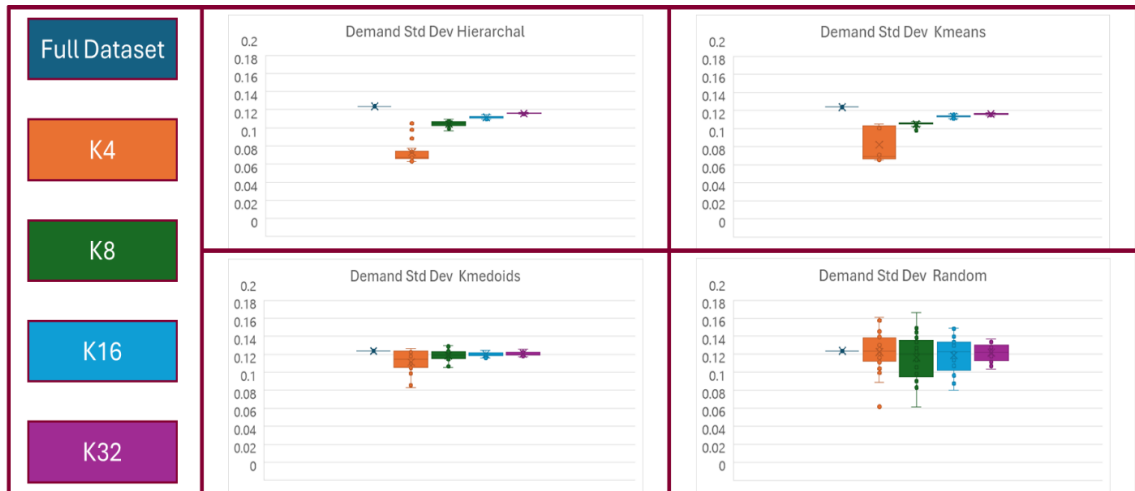


Figure 11 Comparing the demand standard deviation of all cases against the full dataset of RTS1

4.1.2.2. Wind Duration Curve Distance

This measure calculates the difference between the duration curves of the full dataset and those of the expanded clustered dataframes. Ideally, the best-case scenario is a value of 0, indicating perfect alignment between the two curves. As depicted in Figure 12 below, the K-medoids algorithm consistently produces the smallest values across all values of K, reaffirming its superior performance in minimizing the discrepancy between the original and clustered data.

Another important observation is that as K increases, the distance between the duration curves decreases. This trend further supports the concept that a higher number of clusters leads to a more accurate representation of the data, allowing the clustered dataset to better replicate the characteristics of the full dataset. This behavior highlights the importance of selecting an appropriate K value to achieve a balance between computational efficiency and data fidelity.

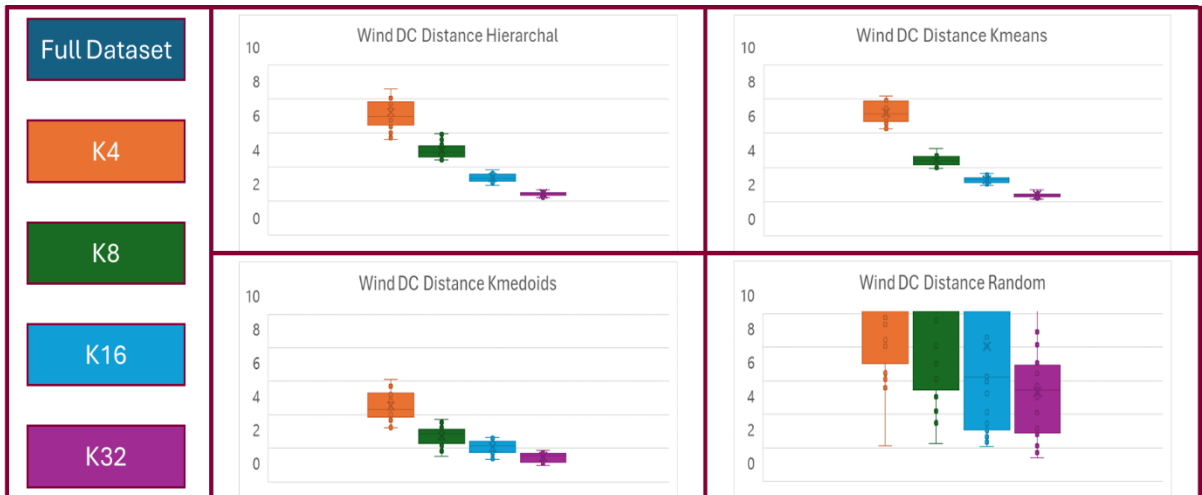


Figure 12 Comparing the Wind duration curve distance of all cases for RTS1

4.1.2.3. Demand-Wind data Correlation

As depicted in Figure 13, clustering algorithms have a tendency to underestimate the demand-wind correlation of the original dataset. This underestimation is most pronounced when K equals 4, but it gradually diminishes as K increases. Once again, K-medoids demonstrates its superior capability in accurately capturing the statistical characteristics of the original dataset, as it consistently produces values closest to the original data correlation among all clustering algorithms. However, this only applies when K equals 4 and 8. For K values of 16 and 32, all clustering algorithms tend to yield similar values. The shifting process has a significant impact, particularly when there are 4 representative days, leading to considerable variability. This variability, however, decreases as K increases. K-medoids is again the algorithm most affected by shifting. Interestingly, this variability is not trivial when compared to the variability observed in

the randomly selected days. This highlights the sensitivity of the K-medoids algorithm to the shifting process, which is an important factor to consider in this analysis.

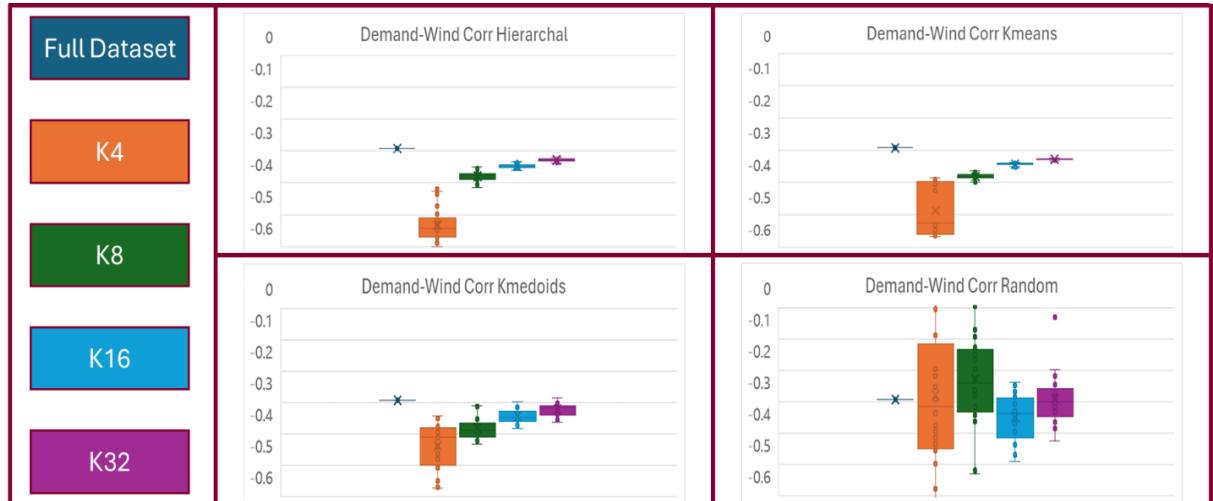


Figure 13 Comparing the demand-wind correlation of all cases against the full dataset

4.1.2.4. Spread Indicator & Shifting Effect

As illustrated in Figure 14, the Spread Indicator for this dataset is 0.2122, leading to relatively high values for the shifting effect. Notably, K-medoids exhibit the highest shifting effect values, confirming numerically that this algorithm is the most sensitive to shifts. This observation underscores the impact of data shifts on K-medoids, making it particularly important to consider this effect when using the algorithm for temporal data clustering.

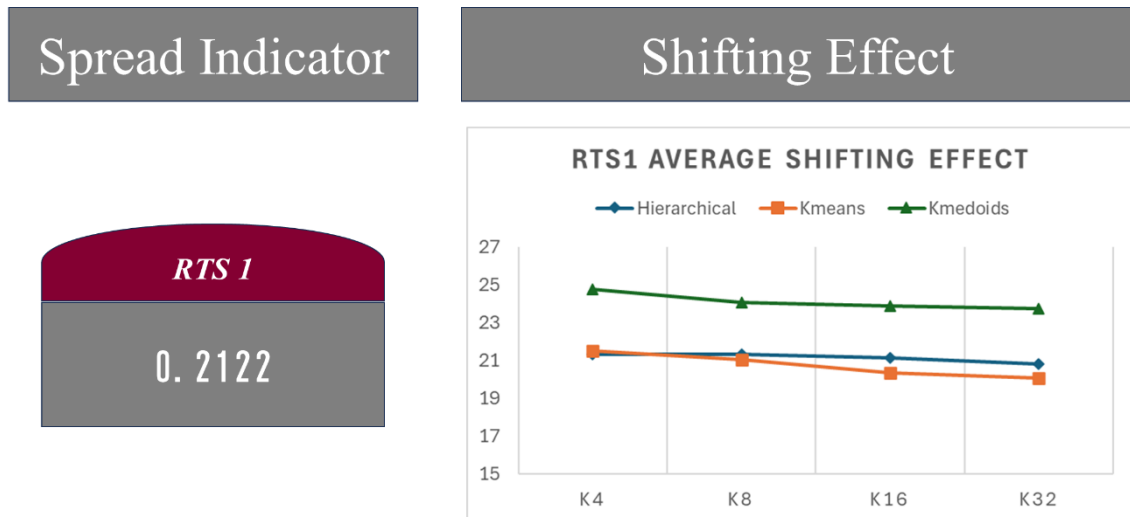


Figure 14 Displaying the values of spread indicator and shifting effect measures for RTS1 data

4.1.3. A Posteriori Measures

Following the a priori measures, which consistently suggest that the k-medoids algorithm is the most effective for clustering and the most susceptible to shifting, a posteriori measures will be employed to validate these findings. These measures will be analyzed using methodologies similar to those applied to the statistical measures. After implementing the optimization model developed by Mr. Joseph Hammana, a co-researcher and fellow student at the American University of Beirut, on all cases, key outputs from the optimization are considered. These include the net present value (NPV) of costs, the cumulative capacity built over a 20-year planning horizon for four critical technologies: Combined Cycle Gas Turbines (CCGT), Photovoltaics (PV), Wind turbines and storage batteries. Additionally, generation of these technologies are also considered to provide deeper insights into the behavior of the optimization model throughout the planning horizon. For traceability, selected results will be displayed in subsequent sections.

4.1.3.1. Net Present Value of Costs

The vertical axis of Figure 15 represents the net present value (NPV) of costs in millions of dollars. The figure clearly shows that the k-medoids algorithm consistently yields costs that are closest to those of the full dataset, especially for lower K values, further validating its superiority among the clustering algorithms. The gap between the costs resulting from the clustered data and those of the full dataset narrows as the number of representative periods (K) increases. This indicates that increasing K not only enhances the accuracy of the data representation but also positively affects the optimization outcomes.

It is also worth noting that as K increases, the results across all algorithms converge, making this measure less effective for evaluating a posteriori performance. This convergence underscores the limitations of relying solely on this measure, thereby validating the importance of the other metrics considered in this study. These additional measures provide a more comprehensive evaluation of the clustering algorithms' performance, ensuring a more robust analysis of their effectiveness.



Figure 15 Comparing the NPV of costs for all cases against the full dataset

4.1.3.2. CCGT Cumulative Capacity Added

Figure 16's vertical axis displays the cumulative added capacity of Combined Cycle Gas Turbines (CCGT) technology in Megawatts (MW) over the 20-year planning horizon. This figure reinforces previous findings, showing that the k-medoids algorithm continues to be the most effective clustering algorithm at replicating the decisions made by the optimization model using the full dataset as input. Additionally, as the number of representative periods (K) increases, the discrepancy between the capacity decisions based on the clustered data and those based on the full dataset diminishes. This trend underscores the effectiveness of increasing K in aligning clustered data outcomes with those of the full dataset. These interpretations hold true for all remaining technologies.

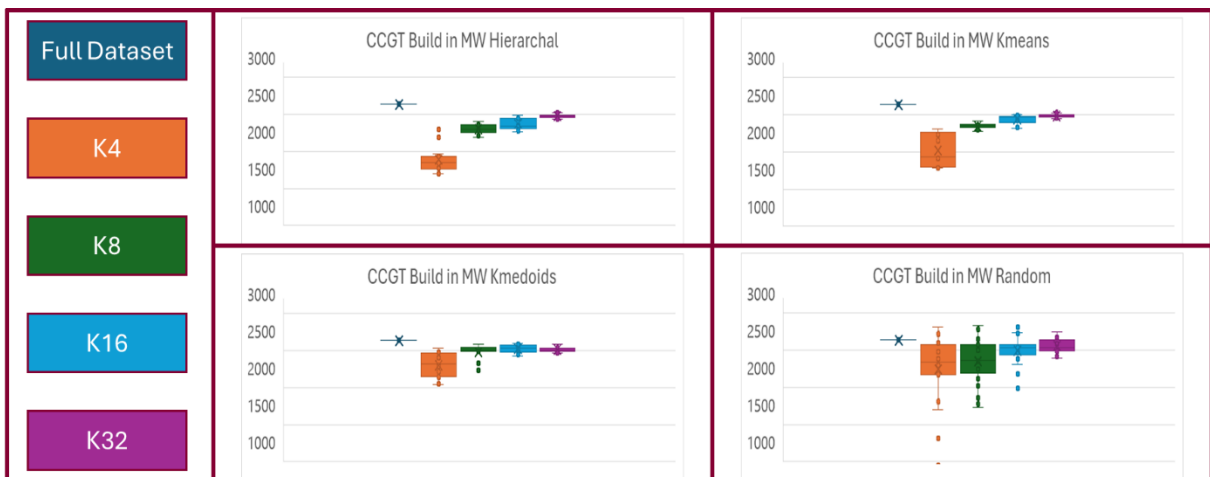


Figure 16 Illustrating the decision of the cumulative CCGT capacity added over the planning horizon, for all clustering algorithms and all K for RTS1.

4.1.3.3. Battery Generation

Figure 17 illustrates the battery generation in terawatt hours (TWh) across the planning horizon, with the vertical axis representing this key metric. Unlike previous observations, the clustering algorithms in this scenario tend to overestimate battery

generation, with this discrepancy becoming more pronounced as the number of representative periods, K , increases.

This trend can be attributed primarily to the unique nature of energy storage systems, which heavily depend on chronological dependencies between days. When using clustering algorithms that segment the dataset into multiple representative periods, these chronological dependencies are partially disrupted, leading to inaccuracies.

The full dataset, treated as a single representative period with 8,760 hours, preserves these chronological links most effectively, as seen in the benchmark case.

Among the algorithms evaluated, K-medoids demonstrates the lowest overestimation, especially for lower values of K . This suggests that K-medoids is better at maintaining the underlying data structure and temporal dependencies, particularly when fewer representative periods are used. Its ability to minimize the gap between the clustered dataset and the full dataset underscores its superiority in scenarios where the chronological integrity of the data is crucial.

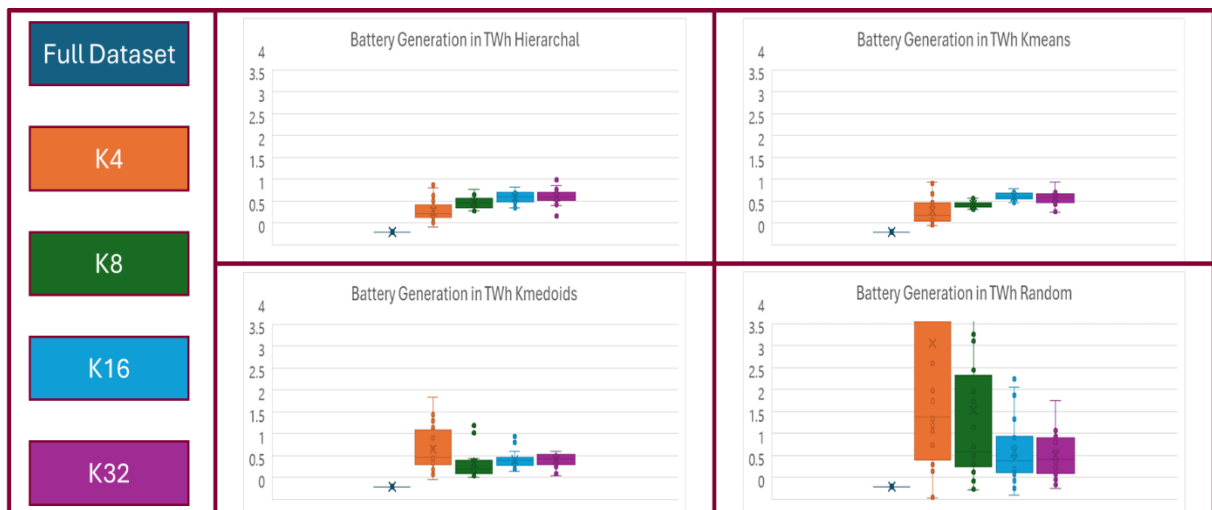


Figure 17 Comparing Battery generation of all cases against the full dataset for RTS1.

4.2. RTS3 Data Case Study

In the subsequent subsections, the results of the second case study will be presented in a manner similar to how the results of the first case study were displayed. This approach will facilitate a direct comparison between the two studies, helping to investigate the generalizability of the findings from the first case study. This comparison will be crucial in determining if the trends and patterns observed in the first study hold true across different datasets or scenarios.

4.2.1. Visual Inspection

The same method of visual inspection used in the first case study is also applied to this case study. The initial step involves plotting the representative periods generated from the 0-shift and comparing them to those generated from a 12-hour shift. As depicted in Figure 18, the deviation between the two types of representative days persists and is most pronounced in the wind attribute. Importantly, this deviation does not diminish with an increased number of clusters (K), indicating that shifting continues to impact the representation of data, particularly for the wind attribute, regardless of the number of representative periods used.

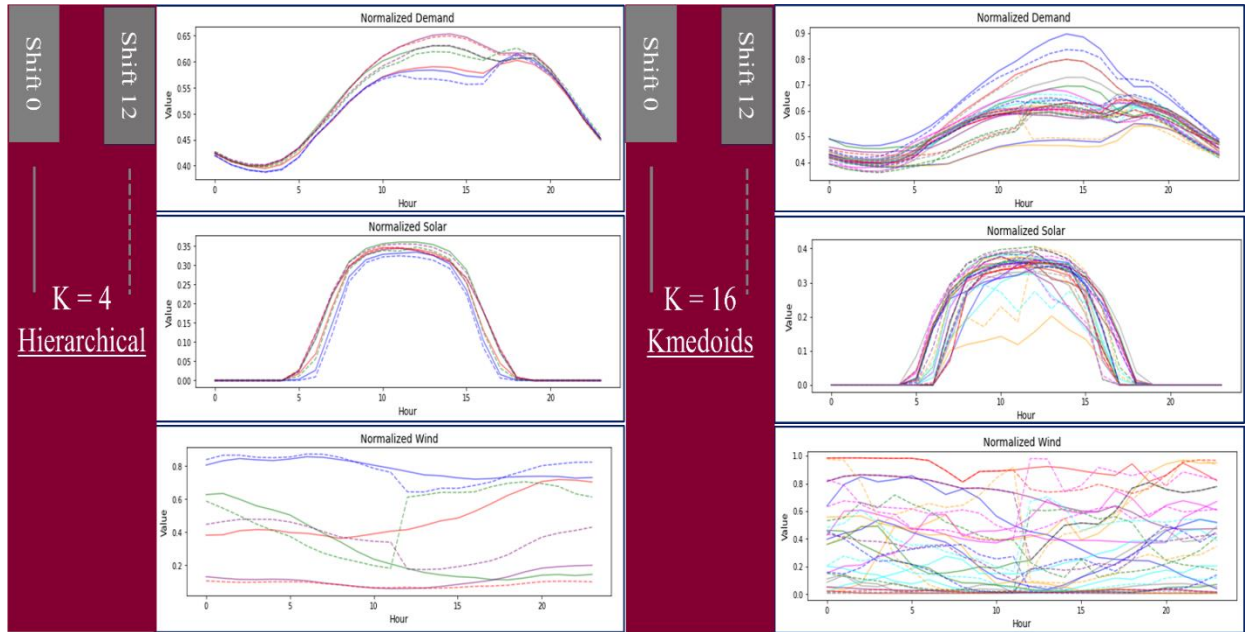


Figure 18 Illustrating the representative periods plots generated by the 0-shift and 12-shift for the second case study.

The second step in the visual inspection process involves plotting the duration curves of the generated representative periods and comparing them to those of the original data. This comparison is crucial for assessing whether an increase in the number of representative periods can enhance the representation of the entire dataset and how slice shifting might influence this capability.

Figure 19 shows the duration curves for the representative periods generated by zero-shift and 12-hour shift. The analysis concludes that as the number of representative days increases, the duration curves of the clustered data (depicted in orange) become smoother and more closely align with the duration curves of the original data (depicted in blue). This improvement indicates that augmenting the number of representative days enhances the ability of the clustered data to encapsulate the full dataset, a finding that is both logical and well-documented in the literature.

In terms of the impact of shifting, and as depicted in figure 19 below, a slight variation is observable between the duration curves of zero-shift and 12-hour shift. This deviation is particularly noticeable in the wind and demand duration curves, underscoring the sensitivity of wind and demand data to slice shifting.

These observations are consistent with those from the first case study, suggesting that the effects observed are generalizable and not specific to the initial case. This alignment further supports the reliability of the findings across different datasets and scenarios.

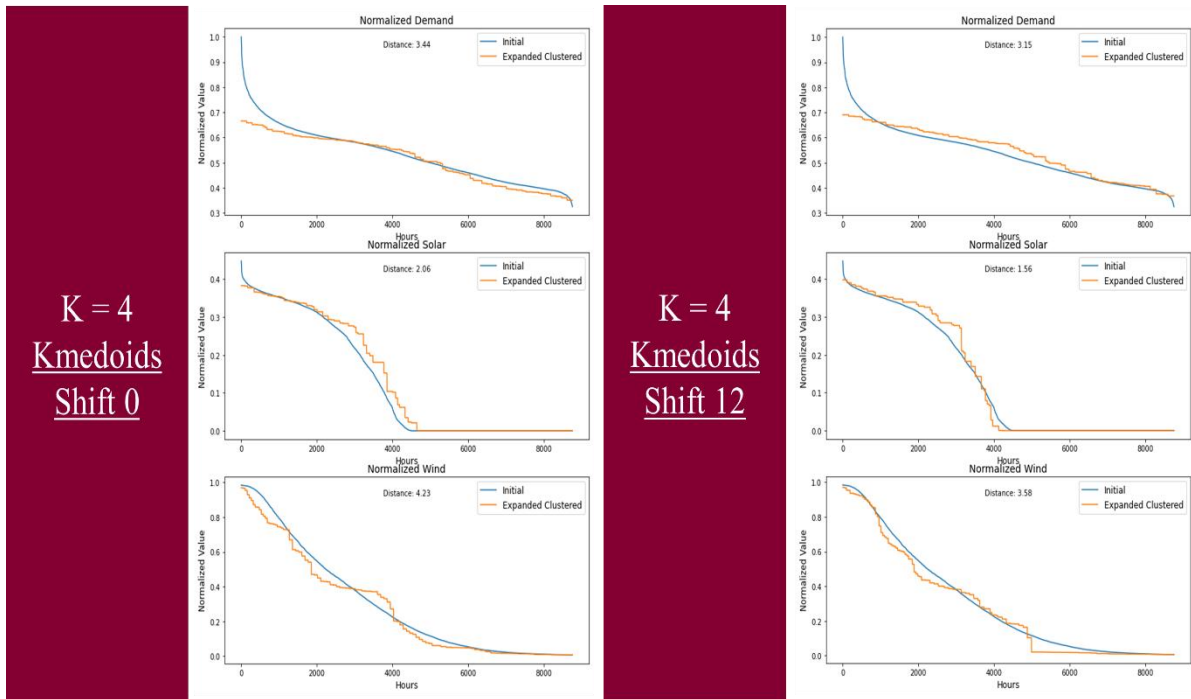


Figure 19 Illustrating the duration curves plots of the original data and representative periods generated with 4 representative days and Kmedoids algorithm for 0-shift and 12-shift for RTS3

4.2.2. Statistical Measures

For the second case study, the same statistical measures used in the first case will be applied. The results will include various statistical metrics such as mean, standard deviation, and correlation coefficients, among others.

4.2.2.1. Demand Standard Deviation

The trends observed in this dataset closely mirror those identified in the analysis of the first dataset. As seen previously, the K-medoids algorithm continues to excel in capturing the standard deviation across all values of K, with the standard deviation values converging toward those of the full dataset as K increases. This reinforces the notion that a higher number of representative periods allows for a more precise depiction of the underlying data dynamics. However, a notable distinction between the two datasets lies in the impact of shifting, which is represented by the width of the boxes in the figures. In this new dataset, and as shown in figure 20 below, the shifting effect is less pronounced, indicating a reduced sensitivity to shifting variations. This suggests that while the overall trends remain consistent, the clustering algorithms exhibit greater stability in this dataset, with less influence from the slice shifts that were more prominent in the first dataset.

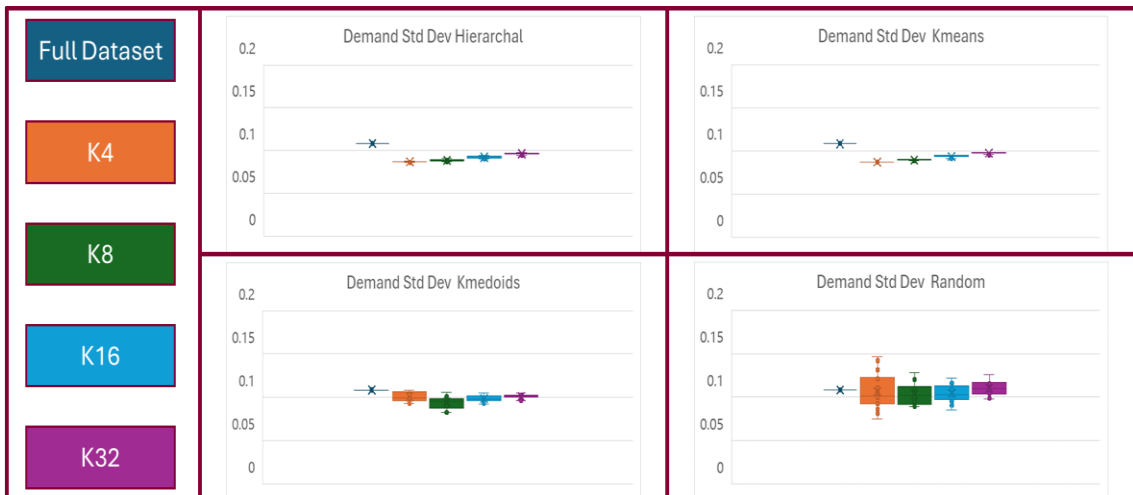


Figure 20 Comparing the demand standard deviation of all cases against the full dataset of RTS

4.2.2.2. Wind Duration Curve Distance

The trends observed in this measure for the new dataset are similar to those identified in the first dataset. Specifically, this measure calculates the difference between the duration curves of the full dataset and those of the expanded clustered dataframes, where an ideal value of 0 would indicate perfect alignment. As illustrated in Figure 21, the K-medoids algorithm consistently produces the smallest discrepancies across all values of K, reaffirming its superior performance in minimizing the deviation between the original and clustered data. This aligns with the findings from the first dataset, where a higher number of clusters (K) also led to a more accurate representation of the data, with the distance between duration curves decreasing as K increased. This consistency further supports the concept that increasing K enhances the fidelity of the clustered dataset in replicating the full dataset's characteristics.

Moreover, the observation of a reduced shifting effect, as noted in the previous section, still persists in this new dataset. This indicates the emergence of a new trend, where the impact of shifting diminishes across measures of the second dataset, indicating that RTS3 is less susceptible to slice shifting.



Figure 21 Comparing the Wind duration curve distance of all cases for RTS3

4.2.2.3. Demand-Wind data Correlation

The interpretations from the first case study concerning the demand-Wind data correlation are applicable to this case as well, as shown in figure 22. All clustering algorithms tend to underestimate this correlation. The k-medoids algorithm consistently generates values that are closest to those of the full dataset. The effect of shifting is slightly noticeable and diminishes as the number of representative days (K) increases.

The boxes width if this measure with RTS3 dataset continues to be less pronounced compared to that of RTS1, reaffirming that this dataset is less susceptible to slice shifting.

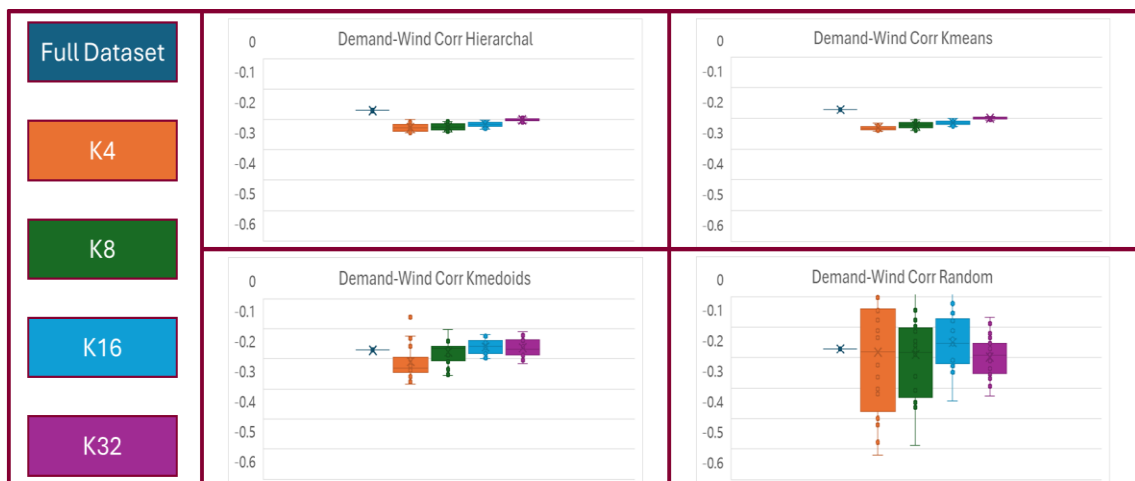


Figure 22 Comparing the correlation between normalized demand and solar capacity factors of the clustered data against the original dataset of RTS3.

4.2.2.4. Spread Indicator & Shifting Effect

As illustrated in Figure 23, the Spread Indicator for this dataset is 0.1773, which is lower compared to the spread indicator of RTS1, explaining the reason why RTS3 dataset is less susceptible to shifting effect. K-medoids still exhibits the highest shifting effect values, confirming numerically that this algorithm is the most sensitive to shifts once again. This observation underscores the impact of data shifts on K-medoids, making

it particularly important to consider this effect when using the algorithm for temporal data clustering. Moreover, these observations reassure that the shifting effect is related to the spread indicator, since a lower spread indicator showed lower shifting effect.

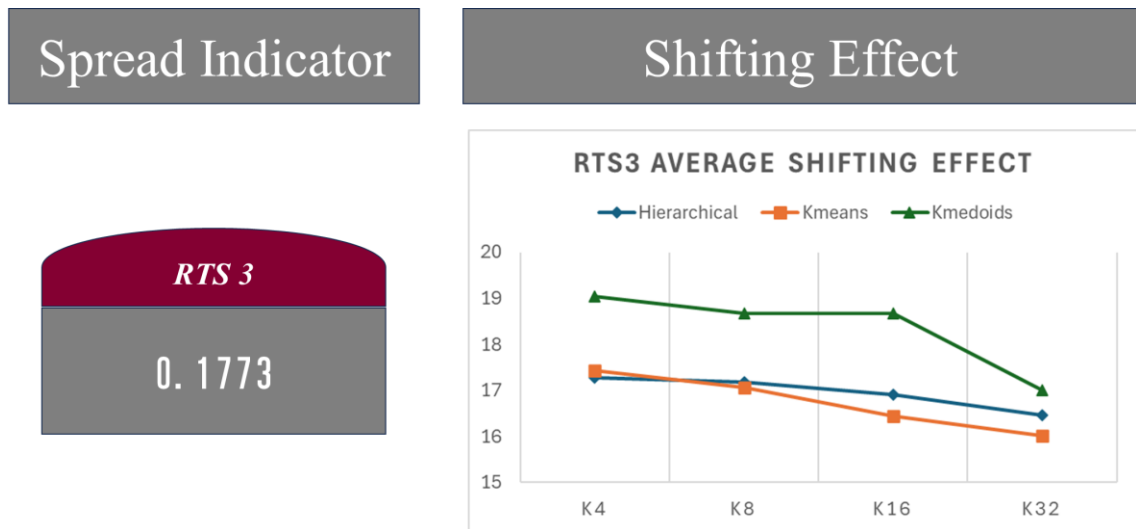


Figure 23 Displaying the values of spread indicator and shifting effect measures for RTS3 data.

4.2.3. A Posteriori Measures

The same a posteriori measures used in the first case study are also applied to this case. These measures will help assess the economic and operational performance of the clustering algorithms under study.

4.2.3.1. Net Present Value of Costs

The findings from the first case study are mirrored in this one. The vertical axis of Figure 24 displays the net present value (NPV) of costs in millions of dollars. The figure demonstrates that the k-medoids algorithm consistently produces cost estimates that closely align with those of the full dataset, affirming its superiority among the clustering algorithms. As the number of representative periods (K) increases, the disparity

between the costs derived from the clustered data and those from the full dataset diminishes. This trend suggests that increasing K improves not only the precision of data representation but also positively impacts the outcomes of optimization.

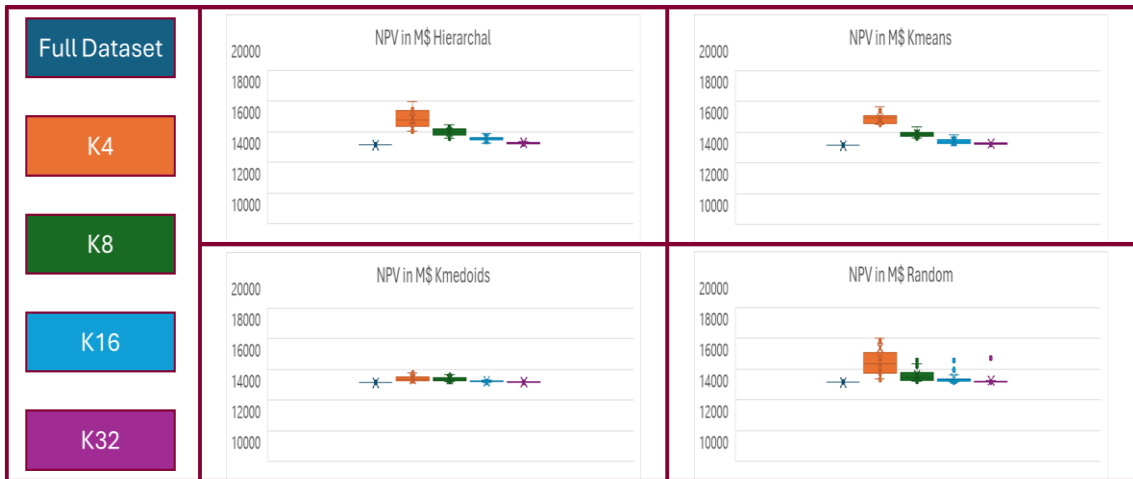


Figure 24 Comparing the net present values of costs of all clustered data to the cost of full dataset for RTS3

4.2.3.2. CCGT Cumulative Capacity Added

The trends visualized in Figure 25 for the cumulative added capacity of Combined Cycle Gas Turbines (CCGT) in Megawatts (MW) over the 20-year planning horizon closely mirror those observed in the first dataset. This figure reinforces previous findings, demonstrating that the K-medoids algorithm remains the most effective clustering algorithm in replicating the capacity decisions made by the optimization model when using the full dataset as input. Additionally, as the number of representative periods (K) increases, the discrepancy between the capacity decisions derived from the clustered data and those based on the full dataset diminishes, further underscoring the effectiveness of increasing K in aligning the clustered data outcomes with those of the full dataset.

Moreover, the observation of a reduced shifting effect, as noted in the previous sections, persists in this measure as well. This suggests that the observation made in the

previous section holds true: datasets with a smaller spread indicator consistently exhibit a less pronounced impact of shifting compared to those with a higher spread indicator.

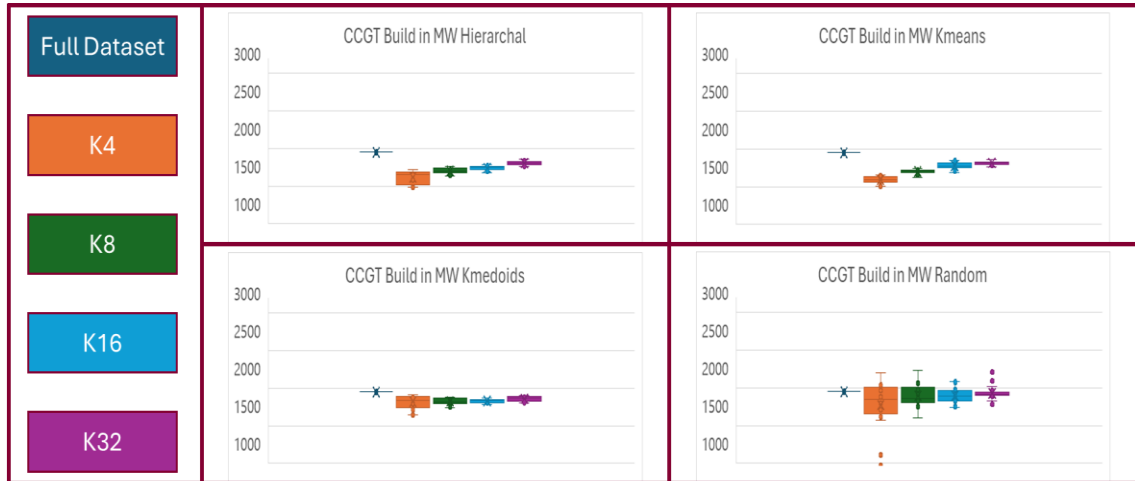


Figure 25 Illustrating the decision of the cumulative CCGT capacity added over the planning horizon, for all clustering algorithms and all K for RTS3

4.2.3.3. Battery Generation

The observations made in the previous dataset regarding battery generation in terawatt hours (TWh) hold true in this new context as well. As previously noted, the K-medoids algorithm did not outperform other clustering algorithms in preserving the underlying data structure, particularly when fewer representative periods are used. This behavior persists in the current dataset, with K-medoids continuing to be the most affected by shifting.

However, and as shown in figure 25 below, due to the unique nature of energy storage systems, the shifting effect is relatively more pronounced in this dataset compared to other measures. This indicates that storage parameters are particularly susceptible to the shifting effect, emphasizing the importance of carefully considering temporal variations when analyzing battery generation.

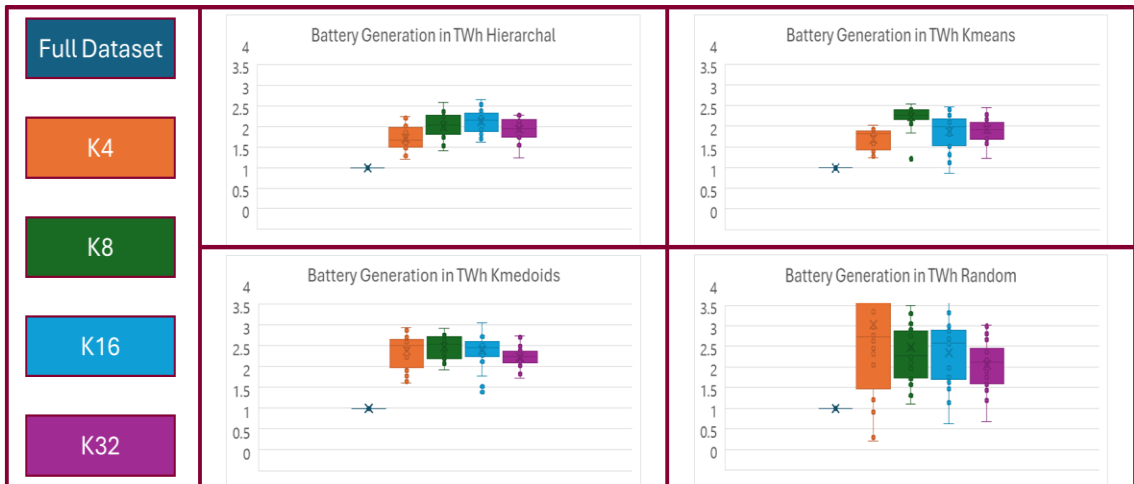


Figure 26 Comparing Battery generation of all cases against the full dataset for RTS3.

CHAPTER 5

LIMITATIONS

This study incorporates a number of limitations that must be acknowledged to appropriately contextualize the findings. Firstly, the use of a simplified Generation Expansion Planning (GEP) optimization model introduces certain constraints. While this model is effective for illustrating broad trends and theoretical implications within the defined parameters, its simplified nature means that the results cannot be considered conclusive or directly translatable to real-world scenarios without further refinement and testing. The model's assumptions and limitations could affect the reliability of the optimization outcomes, thus they should be interpreted with caution.

Secondly, the exclusion of extreme days in the selection of representative periods represents another significant limitation. By not incorporating days with extreme weather conditions or unusual demand spikes, there is a risk that the model smooths out these critical periods, potentially overlooking important aspects of system behavior under stress. This omission can lead to a misrepresentation of the true variability and risk in the energy system being studied. As a result, the clustering algorithms applied may fail to capture the full range of dynamics present in the original dataset, leading to an underestimation of potential challenges in grid management and energy supply reliability.

These limitations suggest that while the study provides valuable insights into the clustering algorithms performance and the potential effect of slice shifting, the findings should be viewed as preliminary. Further investigation with a more comprehensive model is necessary to validate and extend these results.

CHAPTER 6

CONCLUSION

This thesis has thoroughly investigated the performance of various clustering algorithms in the context of Electricity Generation Expansion Planning (GEP), with a particular emphasis on accurately selecting representative periods that reflect the variability in electricity demand and renewable energy supply. The research focused on three primary algorithms: Agglomerative Hierarchical, K-means, and K-medoids, analyzing their effectiveness in replicating the statistical characteristics and GEP decisions derived from the full dataset. Additionally, the study explored the influence of different data slicing methods on the performance of these clustering algorithms.

The findings reveal that slice shifting plays a significant role in the selection of representative periods, impacting both investment and runtime (dispatching) decisions. It was observed that a higher number of representative periods (K) can mitigate the adverse effects of slice shifts, thereby enhancing the accuracy of the clustering outcomes. Among the algorithms studied, the K-medoids algorithm consistently demonstrated superior performance in replicating the full dataset's statistical measures, making it the most effective algorithm in both a priori and a posteriori evaluation.

However, it was also the most sensitive to shifts in data slicing, which could lead to variations in clustering outputs and subsequent GEP decisions.

Moreover, the study identified that the Net Present Value (NPV) of costs is not a sufficient metric on its own to assess the quality of representative periods. This underscores the importance of considering additional measures to test the quality of representing the underlying data. The research also highlighted that the shifting effect observed in a priori measures tends to impose a corresponding shifting effect in a

posteriori measures. This phenomenon is particularly pronounced in datasets with a higher spread indicator, where the impact of slice shifting becomes more significant.

In conclusion, this research underscores the critical importance of accurate data representation in optimizing energy planning and operations. The findings provide valuable insights for both the theoretical understanding and practical application of clustering algorithms in GEP. By demonstrating the effects of slice shifting and the sensitivity of the K-medoids algorithm, this study contributes to the ongoing efforts to improve the accuracy and reliability of energy system modeling and decision-making processes.

The main contribution of this research lies in highlighting a potential issue that could impact clustering and GEP model outcomes: the slice shifting of data. To address this, the study introduces two new a priori measures—the spread indicator and the shifting effect. The results show that datasets with a higher spread indicator are more susceptible to significant shifting effects, which could lead to variability in GEP model outcomes and undermine the robustness of the results. This lack of robustness is a critical concern, as it may lead to incorrect decisions and suboptimal resource usage.

However, if the spread indicator is low, there is minimal concern regarding shifting effects. In cases where the spread indicator is high, special attention is required, and one potential solution is to use a stochastic GEP model that incorporates the outputs from 24 different shifts. This approach could help accommodate the shifting effects and ensure more reliable and consistent outcomes.

REFERENCES

- Adhau, S. P., Moharil, R. M., & Adhau, P. G. (2015). K-means clustering technique applied to availability of micro hydro power. *Sustainable Energy Technology Assessment*, 8, 191-201.
- Agapoff, S., Pache, C., Panciatici, P., Warland, L., & Lumbreras, S. (2015). Snapshot selection based on statistical clustering for transmission expansion planning. In *2015 IEEE Eindhoven PowerTech*, 1-6.
- Almaimouni, A. (2018). Taming the Curse of Dimensionality in the Generation Expansion Planning Problem (Ph.D. thesis), *University of Washington*.
- Almaimouni, A., Ademola-Idowu, A., Kutz, J. N., Negash, A., & Kirschen, D. (2018). Selecting and evaluating representative days for generation expansion planning. In *XX Power Systems Computation Conference*, Dublin, Ireland, 1-7.
- Arthur, D., & Vassilvitskii, S. (2007). K-means++: the advantages of careful seeding. In *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, Vol. 8, 1027-35.
- Bahl, B., Kümpel, A., Seele, H., Lampe, M., & Bardow, A. (2018). Typical periods for two-stage synthesis by time-series aggregation with bounded error in objective function. *Frontiers in Energy Research*, 5(January), 1-13.
- Bahl, B., Söhler, T., Hennen, M., & Bardow, A. (2017). Time-series aggregation for synthesis problems by bounding error in the objective function. *Energy*, 135, 900-12.
- Blanford, G. J., Merrick, J. H., Bistline, J. E., & Young, D. T. (2016). Simulating annual variation in load, wind, and solar by representative hour selection. *The Energy Journal*, 39(3), 189-212.
- Brodrick, P., Kang, C., Brandt, A., & Durlofsky, L. (2015). Optimization of carbon-capture-enabled coal-gas-solar power generation. *Energy*, 79, 149-62.
- Buchholz, S., Gamst, M., Pisinger, D., Management, D. T. U., & Lyngby, D.-K. (2020). Sensitivity analysis of time aggregation techniques applied to capacity expansion energy system models nuclear always online single cluster with random selection. *Applied Energy*, 269(April), 114938.
- Child, M., Kemfert, C., Bogdanov, D., & Breyer, C. (2019). Flexible electricity generation, grid exchange and storage for the transition to a 100% renewable energy system in Europe. *Renewable Energy*, 139, 80–101.
- ENTSO-E. (2023). Transparency platform. <https://transparency.entsoe.eu/>. Last Accessed: 6 January 2023

- Fazlollahi, S., Bungener, S. L., Mandel, P., Becker, G., & Marechal, F. (2014). Multi-objectives, multi-period optimization of district energy systems: I-selection of typical operating periods. *Computers & Chemical Engineering*, 65, 54-66.
- Fitiwi, D. Z., De Cuadra, F., Olmos, L., & Rivier, M. (2015). A new approach of clustering operational states for power network expansion planning problems dealing with RES (renewable energy source) generation operational variability and uncertainty. *Energy*, 90, 1360-76.
- Green, R., Staffell, I., & Vasilakos, N. (2014). Divide and conquer? k-means clustering of demand data allows rapid and accurate simulations of the British electricity system. *IEEE Transactions on Engineering Management*, 61(2), 251-60.
- Hilbers, A. P., Brayshaw, D. J., Gandy, A. (2019). Importance subsampling: improving power system planning under climate-based uncertainty. *Applied Energy*, 251(March), 113114.
- Hoffmann, M., Kotzur, L., Stolten, D., & Robinius, M. (2020). A review on time series aggregation methods for energy system models. *Energies*, 13(3), 641.
- Hoffmann, M., Priesmann, J., Nolting, L., Praktiknjo, A., Kotzur, L., & Stolten, D. (2021). Typical periods or typical time steps? a multi-model analysis to determine the optimal temporal aggregation for energy system models. *Applied Energy*, 304, 117825
- IRENA. (2022). Renewable power generation costs in 2021. Report, *International Renewable Energy Agency*.
- IEA. (2022a). Global energy and climate model. Report, *International Energy Agency*.
- IEA. (2022b). World energy outlook 2022. Report, *International Energy Agency*.
- Kotzur, L., Markewitz, P., Robinius, M., & Stolten, D. (2018). Impact of different time series aggregation methods on optimal energy system design. *Renewable Energy*, 117, 474-87.
- Li, C., Conejo, A. J., Sirola, J. D., & Grossmann, I. E. (2022). On representative day selection for capacity expansion planning of power systems under extreme operating conditions. *International Journal of Electrical Power & Energy Systems*, 137, 107697.
- Lloyd, S. P. (1982). Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2), 129-37.
- Mallapragada, D. S., Papageorgiou, D. J., Venkatesh, A., & Grossmann, I. E. (2018). Impact of model resolution on scenario outcomes for electricity sector system expansion. *Energy*, 163, 1231-44.

- Nahmmacher, P., Schmid, E., Hirth, L., & Knopf, B. (2016). Carpe diem: A novel approach to select representative days for long-term power system modeling. *Energy*, 112, 430-442.
- Pfenninger, S. (2017). Dealing with multiple decades of hourly wind and PV time series in energy models: A comparison of methods to reduce time resolution and the planning implications of inter-annual variability. *Applied Energy*, 197, 1-13.
- Poncelet, K., Hoschle, H., Delarue, E., Virag, A., & D'haeseleer, W. (2016). Selecting representative days for capturing the implications of integrating intermittent renewables in generation expansion planning problems. *IEEE Transactions on Power Systems*, 1-12.
- Reichenberg, L., Siddiqui, A. S., Wogrin, S. (2018). Policy implications of downscaling the time dimension in power system planning models to represent variability in renewable output. *Energy*, 159, 870-7.
- RTS (2023). Website, <https://github.com/GridMod/RTS-GMLC>, last accessed September 15, 2023
- Sarajpoor, N., Rakai, L., Arteaga, J., & Zareipour, H. (2021). A shape-based clustering framework for time aggregation in the presence of variable generation and energy storage. *IEEE Open Access Journal of Power and Energy*, PP, 1
- Scott, I. J., Carvalho, P. M., Botterud, A., & Silva, C. A. (2019). Clustering representative days for power systems generation expansion planning: Capturing the effects of variable renewables and energy storage. *Applied Energy*, 253, 113603.
- Scott, I. J., Carvalho, P. M., Botterud, A., & Silva, C. A. (2021). Long-term uncertainties in generation expansion planning: Implications for electricity market modelling and policy. *Energy*, 227, 120371.
- Seljom, P., Kvalbein, L., Hellemo, L., Kaut, M., & Ortiz, M. M. (2021). Stochastic modelling of variable renewables in long-term energy models: Dataset, scenario generation & quality of results. *Energy*, 236, 121415.
- Sun, M., Teng, F., Zhang, X., & Strbac, G. (2019). Data-driven representative day selection for investment decisions: A cost-oriented approach. *IEEE Transactions on Power Systems*, PP©, 1.
- Teichgraeber, H., & Brandt, A. R. (2022). Time-series aggregation for the optimization of energy systems: Goals, challenges, approaches, and opportunities. *Renewable and Sustainable Energy Reviews*, 157, 111984.
- Teichgraeber, H., Brodrick, P. G., & Brandt, A. R. (2017). Optimal design and operations of a flexible oxyfuel natural gas plant. *Energy*, 141, 506-18.

- Teichgraeber, H., & Brandt, A. R. (2019). Clustering methods to find representative periods for the optimization of energy systems: An initial framework and comparison. *Applied Energy*, 239, 1283-93.
- Thorndike, R. L. (1953). Who belongs in the family? *Psychometrika*, 18(4), 267-76.
- Tibshirani, R., Walther, G., & Hastie, T. (2001). Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(2), 411-23.
- Zatti, M., Gabba, M., Freschini, M., Rossi, M., & Gambarotta, A. (2019). K-MILP: A novel clustering approach to select typical and extreme days for multi-energy systems design optimization. *Energy*, 181, 1051-63.