energies

MDPI

Article

Clustering Method for Load Demand to Shorten the Time of Annual Simulation

Yuya Tanigawa [1,*], Narayanan Krishnan [2], Eitaro Oomine [3], Atushi Yona [1], Hiroshi Takahashi [4] and Tomonobu Senjyu [1]

[1] Fuculty of Engineering, University of the Ryukyus, 1 Senbaru, Nishihara-cho 903-0213, Nakagami, Okinawa, Japan
[2] Department of Electrical and Electronics Engineering, SASTRA Deemed University, Thanjavur 613401, Tamil Nadu, India
[3] Central Research Institute of Electric Power Industry, 2-6-1 Nagasaka, Yokosuka-City 240-0196, Kanagawa, Japan
[4] Fuji Elctric Co., Ltd., Tokyo 141-0032, Japan
* Correspondence: k228478@eve.u-ryukyu.ac.jp

Abstract: UC (unit commitment) for grid operation has been attracting increasing attention due to the growing interest in global warming. Compared to other methods, MILP, which is one of the calculation methods for UC, has the disadvantage of a long calculation time, although it is more accurate in considering constraints and in finding solutions. However, RLCs (representative load curves) require a more accurate clustering method to select representative dates because the calculation results vary greatly depending on the clustering method. DBSCAN, one of the clustering methods, has the feature that the clustering accuracy varies depending on two parameters. Therefore, this paper proposes two algorithms to automatically determine the two parameters of DBSCAN to perform RLCs using DBSCAN. In addition, since DBSCAN has the feature of being able to represent different data as two-dimensional elements, a survey of the data to be used as clustering was conducted. As a result, the proposed algorithms enabled a more accurate clustering than the conventional method. It was also proved that clustering including temperature and load demand as clustering classification factors enables clustering with higher accuracy. The simulation with shorter time was also possible for the system including storage batteries as a demand response.

Keywords: unit commitment; clustering; DBSCAN; RLC

check for updates

Citation: Tanigawa, Y.; Krishnan, N.; Oomine, E.; Yona, A.; Takahashi, H.; Senjyu, T. Clustering Method for Load Demand to Shorten the Time of Annual Simulation. *Energies* **2023**, *16*, 2264. https://doi.org/10.3390/en16052264

Academic Editor: Alan Brent

Received: 21 December 2022
Revised: 4 February 2023
Accepted: 24 February 2023
Published: 27 February 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

In recent years, renewable energy power generation has been attracting attention due to growing concern over global warming. Fossil fuel power generation is a cause of increased greenhouse gas emissions [1]. Renewable energies are non-fossil fuels and do not emit greenhouse gases when power is generated, so countries worldwide are aiming to convert their existing power generation facilities to renewable energies [2]. However, the amount and duration of power generation from renewable energies are unstable, which causes uncertainty in the stable supply of electricity. In particular, the duck curve phenomenon, which occurs with the introduction of large amounts of PV, causes load fluctuations to increase due to PV generation, leading to problems with the number of generator startups and shutdowns at certain times [3]. The introduction of renewable energy sources causes unstable grid operation, resulting in a situation where a stable power supply cannot be achieved.

Against this background, there has been growing interest in unit commitment (UC) for power system operation and scheduling [4]. UC is a study of stable power system operation and load demand distribution by proposing optimal operation schedules for generators [5]. In Ref. [6], it was proved that the proposal of optimal generator operation

scheduling by UC can reduce the grid operation cost. In Ref. [7], UC was proposed to improve the duck curve phenomenon by peak shifting through the introduction of demand response and ESS. The UC simulation can solve the grid operation unstable caused by the increase of renewable energy generation that may occur in the future.

Although various methods have been used to derive UC, MILP has been proven to obtain optimal solutions with high quality when compared to each method [8]. In addition, MILP is characterized by the fact that it is relatively easy to introduce and take into account constraint conditions, and simulations can be performed with a model that is close to reality [9]. However, one of the disadvantages of MILP is that the computation time increases due to the increase in the number of constraints and variables to be considered and the larger scale of the model [8,10]. In particular, the binary variables make more realistic simulations possible, but at the same time, the computation time increases exponentially [11]. As shown in Ref. [12], relaxing binary variables does not necessarily lead to a reduction in computational load, so methods to shorten simulation time are required.

Moreover, although the introduction and study of various demand response systems is an important issue in UC [7], a one-day simulation is not sufficient. Since load demand varies greatly depending on climatic changes such as seasons and people's activities, simulations for at least one year are necessary to evaluate the proposed method [13]. However, as previously mentioned, MILP simulation requires a lot of time to solve the problem.

One of the fastest methods to obtain solutions in MILP is RLCs (representative load curves), which is a method to reduce the number of load curves using clustering of load demand over one year [14]. In RLC, load demand is classified into several patterns based on some criteria, and a representative day is created. By considering the weights of the representative days, it is possible to obtain results that are almost equivalent to running a one-year simulation with a small number of simulations and a short time period. However, it is important to use clustering with high clustering accuracy because the results can vary greatly depending on the clustering method.

Clustering Method

The main clustering methods are supervised and unsupervised learning. As a representative of supervised learning, population neural networks (ANNs) learned from past cases were used as the standard AI technique [15]. This technique is effective for future forecasting and performed better than other forecasting methods for PV and wind speed forecasting in Ref. [16]. However, a major characteristic of supervised learning is that it mostly requires large amounts of high-quality training data. Therefore, unsupervised learning is used for classification of historical data.

The k-means method is a typical clustering method for unsupervised learning. The k-means method is easy to implement and provides accurate clustering [17]. The k-means method begins by specifying the number of groups (clusters) to be classified, then randomly assigns all elements of the data to the clusters, and classifies the elements of the data by using the center of gravity within the clusters and the average value of each element. Therefore, the most important aspect of this method is how k is determined, since the results can vary greatly depending on the number of clusters k. Several studies have been conducted on how to determine the cluster k for using the k-means method. In Ref. [18,19], the number of clusters has been determined using a method called the elbow method. In this method, the number of clusters k is varied one by one, from one to the total number of elements, and the residual sum of squares (SSE value) of the clusters is calculated and plotted on a diagram. In Refs. [20], based on the idea that the electricity usage of consumers differs depending on the season and on weekdays, weekends, and holidays, the number of patterns to be classified is created in advance, and this is used as the number of clusters. However, the common problems with these methods are that the results and accuracy of the classification depend on the clustering user and that k must be determined in advance. If an incorrect number of clusters is used, the accuracy of clustering with load demand data is greatly reduced and complexity is increased [21].

In addition, several references [22–24] use k-means for clustering load demand, but since k-means can only handle one parameter, clustering is performed using load demand and renewable energy generation (PV and wind power). However, load demand and renewable energy generation vary with external factors (temperature, solar radiation, wind speed, etc.), but these external factors are not taken into account when clustering.

Therefore, in this paper, we examine the adaptation of DBSCAN (density-based spatial clustering of applications with noise), a clustering method that can perform clustering without predetermining the number of clusters, to pattern classification of load demand. DBSCAN is a clustering method characterized by the fact that it performs classification based on the distance between elements and can remove elements with weak relationships to each other from the classification as noise [25]. However, since DBSCAN uses two parameters, ϵ and MinPts, the clustering accuracy can vary greatly depending on the settings of these two parameters [26]. In this paper, we present a new approach to automatically define the eps and MinPts parameters for the DBSCAN algorithm.

In this paper, we improve DBSCAN by using two cases to enable more detailed cluster classification and noise elimination compared to the conventionally used DBSCAN algorithm (Ref. [27]) to increase clustering accuracy. Since DBSCAN has the property of being able to plot data on a two-dimensional plane and cluster them, this paper investigates effective combinations for load demand classification from 12 data sets (temperature, humidity, wind speed, etc.).

In Case 1, the k-dist plot in the Ref. [27] was improved to automatically determine more knees for clustering. In Case 2, clustering was performed using DBSCAN noise reduction and k-means center-of-gravity updating. The results are compared with those of the DBSCAN algorithm (Ref. [27]), the k-means method, and annual operations without clustering to demonstrate the effectiveness of the algorithm.

The remainder of this paper is organized as follows. Section 2 introduces DBSCAN and the proposed method. Section 3 presents the objective function for minimizing operating costs and the constraints considered in performing the optimization. Section 4 presents the power system model assumed in this paper. Section 5 presents and discusses the simulation results. Section 6 concludes the paper.

2. DBSCAN

2.1. DBSCAN Method

The advantages of DBSCAN are that, unlike the k-means method, it is not necessary to provide a pioneering k number of clusters and that it can determine the noise. DBSCAN does not require the number of clusters k, but it does require the constant parameters ϵ and MinPts. The detailed method of DBSCAN is shown below, and an image of DBSCAN is shown in Figure 1.

(1) Label the elements of the data set based on the following three conditions
   - A core point is a point such that there are at least the specified number (MinPts) of adjacent points within the specified radius ϵ.
   - A border point is a point such that the number of adjacent points of radius ϵ is less than MinPts, but is located within radius ϵ of the core point. In Figure 1, the center point of the red dotted line is the boundary point.
   - All other points that are neither core nor border points are considered noise points.

(2) Clusters are formed for each core point. If the core points are within radius ϵ of each other, they are assumed to belong to the same cluster.

(3) Assign each border point within radius ϵ of the core point to a cluster of that core point.

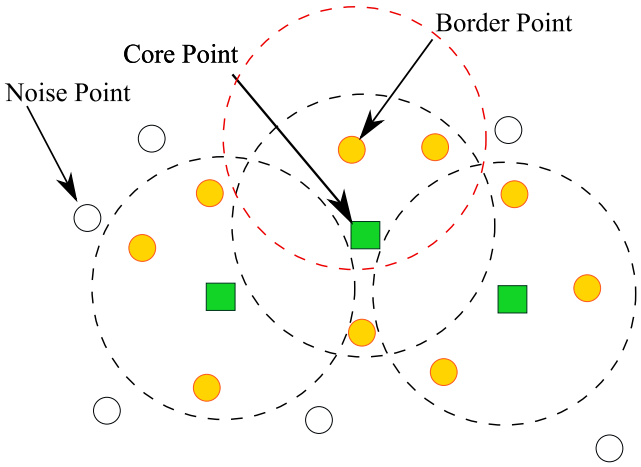


Figure 1. DBSCAN Image.

2.2. DBSCAN Adaptation Methods

As explained in Section 2.1, unlike the k-means method, DBSCAN requires elements to be represented as points. Therefore, in order to classify load demand patterns using DBSCAN, this paper performs DBSCAN by selecting two different data sets from 12 different data sets for one year and plotting their combinations on a two-dimensional plane. Representative days are then created by assigning days with similar characteristics to the same cluster. Finally, the representative days are used for operational planning to find the most appropriate combination of data for classifying load demand in terms of simulation time and error with the results for one year. The types of data used in this paper are listed in Table 1. Since DBSCAN obtains the radius ϵ by the Euclidean distance, it is not possible to properly classify the data if they are combined with each other, each of which has a different scale. Therefore, when combining two sets of data, each data value was converted to the centile notation to accommodate clustering. The percentage conversion formula is shown in Equation (1). The data in Table 1 were taken from the 2016 dataset published by the JMA and Okinawa Electric Power Company [28,29].

- Percentage notation conversion formula for each data value

$$X_{100,i} = \frac{X_i}{X_{max}} \times 100 \tag{1}$$

where X: the set of data for 365 days, X_{100}: the set of data for 365 days (after percentage transformation), $X_{100,i}$: the i-th data value after percentage notation transformation, X_i: the i-th data value in X, X_{100} is plotted on the two combined coordinate planes.

Table 1. Clustering Elements of DBSCAN.

Data Element
1. Average humidity [%]
2. Load demand [MW]
3. Total precipitation [mm]
4. Total solar radiation [Mj/m^2]
5. Maximum temperature [°C]
6. Average temperature [°C]
7. Minimum temperature [°C]
8. Load maximum time [hour]
9. Minimum load time [hour]
10. Maximum wind speed [m/s]
11. Average wind speed [m/s]
12. Weekdays, weekends, and holidays

2.3. DBSCAN Implementation Steps

DBSCAN needs to determine the radius ϵ of an element and the number of elements MinPts within the radius. The setting of these two parameters determines the clustering accuracy. Therefore, we used an algorithm to automatically determine the parameters of ϵ and MinPts using the k-dist plot method introduced in Ref. [27]. The procedure is shown below.

- The distance from each element to the *k*th element away is k-dist.
- The k-dist-mean is the average of the distances of each element to all elements contained within the radius k-dist.
- Arrange the obtained k-dist-means in descending order.
- Find one point P that is the knee from the graph.
- The resulting k-dist value of point P is the radius ϵ, and the number of elements that the k-dist-mean of point P contains is MinPts.

Figure 2 shows the k-dist graph. Figure 3 shows the k-dist mean and the determined knee. Although it is difficult to determine the knee from Figure 2, Figure 3 graph makes it easier to find the knee.

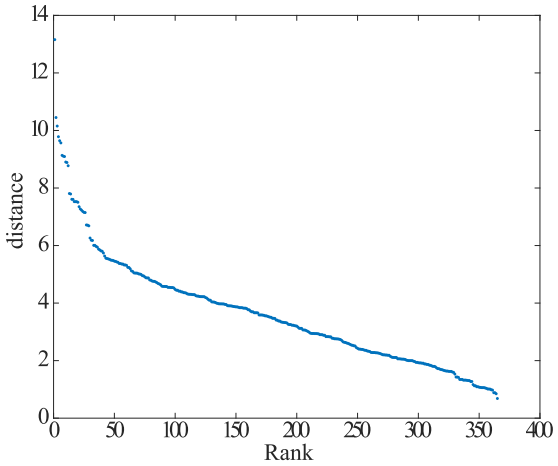


Figure 2. K-dist graph.

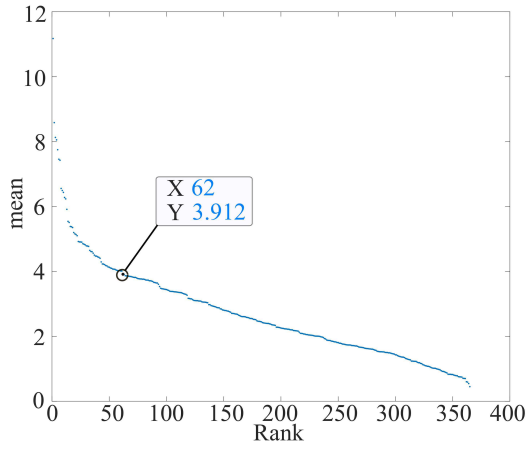


Figure 3. K-dist-mean graph.

2.4. Proposal Method

Figure 4 shows the results of DBSCAN using the method described in Section 2.3. The k-dist plot method can determine only one knee point, so one cluster may become huge as shown in Figure 4. Therefore, it is not possible to properly classify each element from the results, so improvements were made in two cases.

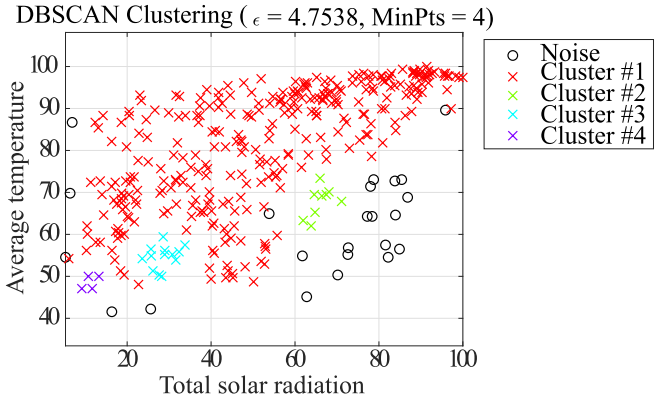


Figure 4. DBSCAN results (before improvement).

- Case 1: Clustering by improving k-dist plot

Since only one knee is determined in the k-dist plot, a problem arose in which one cluster became too large. Therefore, in Case 1, multiple knees are automatically determined in the k-dist plot, DBSCAN is performed sequentially starting from the knee with the smallest k-dist value, and the obtained results are updated to remove noise and perform detailed cluster classification. The procedure is described below.

1. Calculate the gradient between adjacent points of elements of the reordered k-dist-mean graph.
2. From the graph, divide the elements into fixed intervals (10 equal parts in this paper) and find the average value of the gradient of the elements within the interval. The average slope for ranks 37–72 in Figure 5 is 0.0213.
3. The element with a gradient value that is more prominent than the mean value (five times in this paper) is designated as the knee-point and stored in the knee-list. The gradient of the white circle for ranks 37–72 in Figure 5 was 0.1543, so it was stored in the knee-list as a Knee-Point.
4. Here, for noise elimination, all the knee-points in the knee-list that have a prominent gradient ground (5 times in this paper) above the average of the gradient of the entire graph are eliminated. The average value of the overall slope in this study is 0.0303.
5. Select the knee-point with the smallest k-dist value from the knee-list, calculate the radius ϵ and MinPts, and perform DBSCAN.
6. All the elements assigned to clusters in the result obtained at this time are saved, and the saved elements are removed from the two-dimensional plane. Then, DBSCAN is performed on the next knee-point with a large k-dist as in step 5.
7. Perform step 6 for all knee points in the knee list and finish clustering by adapting DBSCAN on all elements. Finally, the stored grouped elements and noise are shown in the same graph.

The graph of k-dist mean adapted from case 1 is shown in Figure 5, and the process from step 5 to step 7 is shown in Figures 6–11. The final DBSCAN results for case 1 are shown in Figure 12. By finding multiple knees as shown in Figure 5 and repeating DBSCAN as shown in Figures 6–11, it was possible to perform detailed classification of the entire data and eliminate noise as shown in Figure 12. The flow diagram for Case 1 is shown in Figure 13.

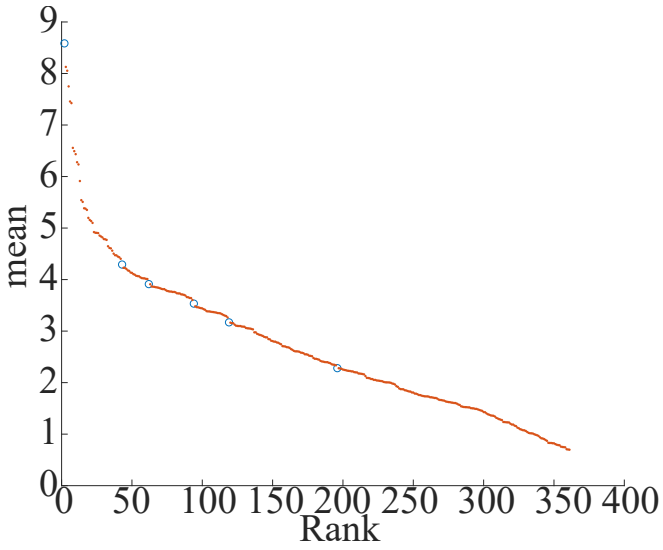


Figure 5. K-dist-mean graph after knee search.

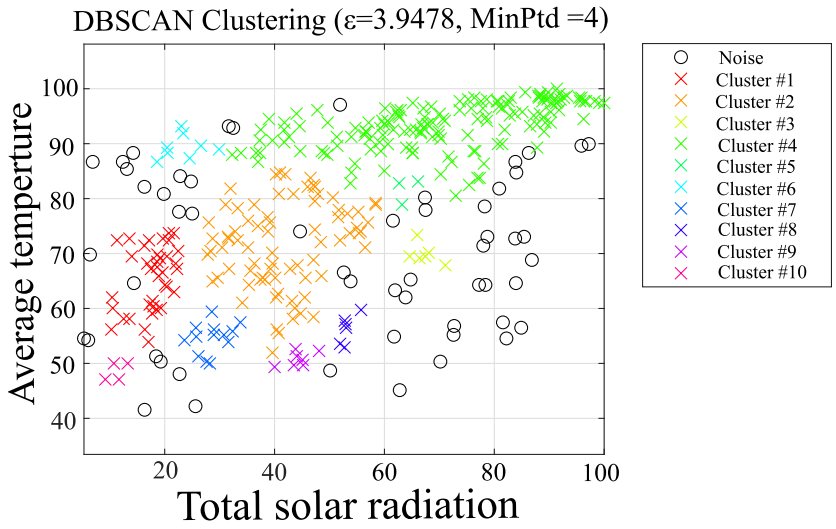


Figure 6. Case1-1.

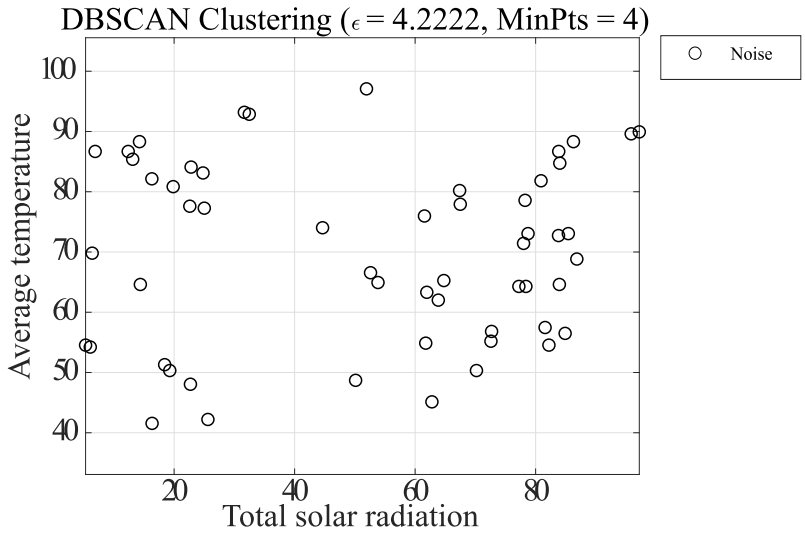


Figure 7. Case 1-2.

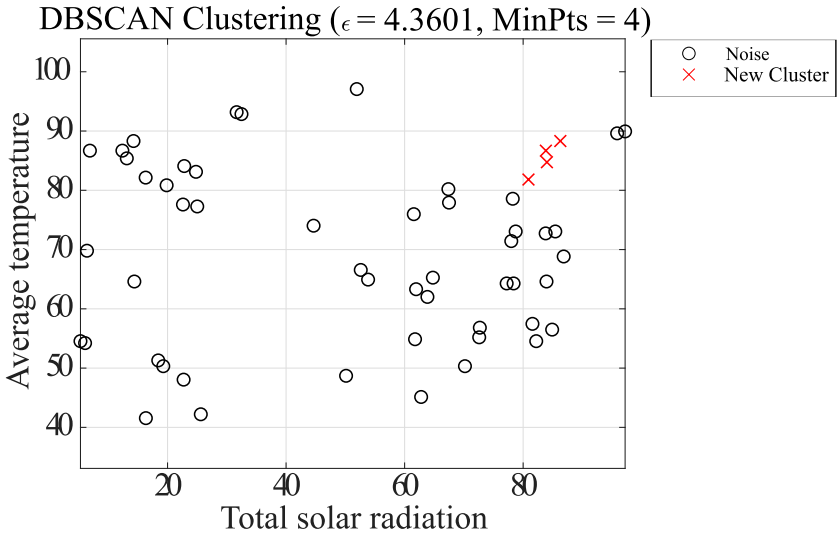


Figure 8. Case 1-3.

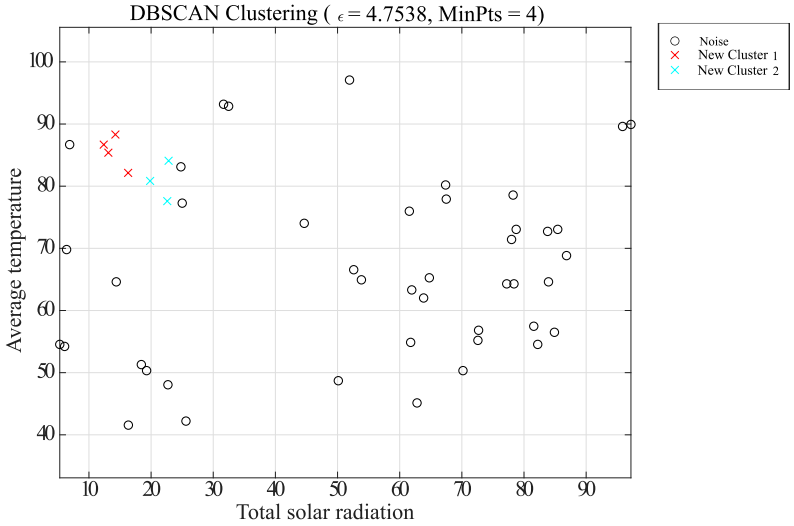


Figure 9. Case 1-4.

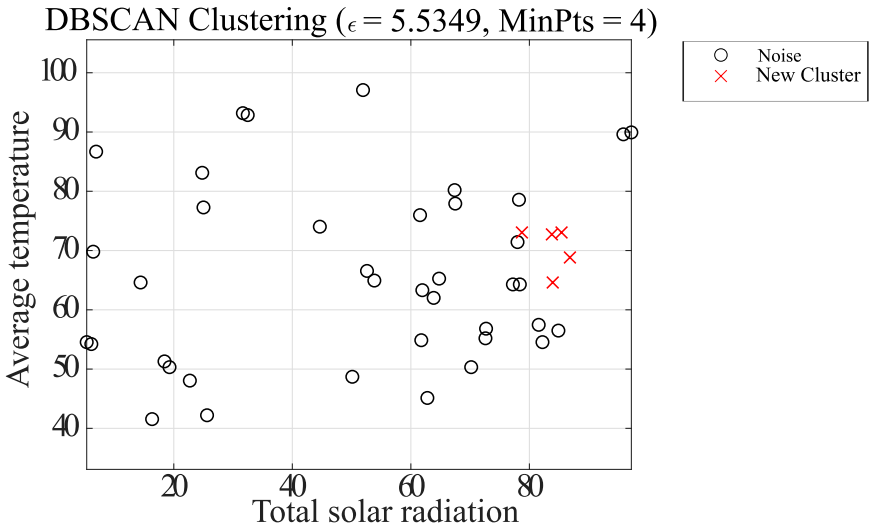


Figure 10. Case 1-5.

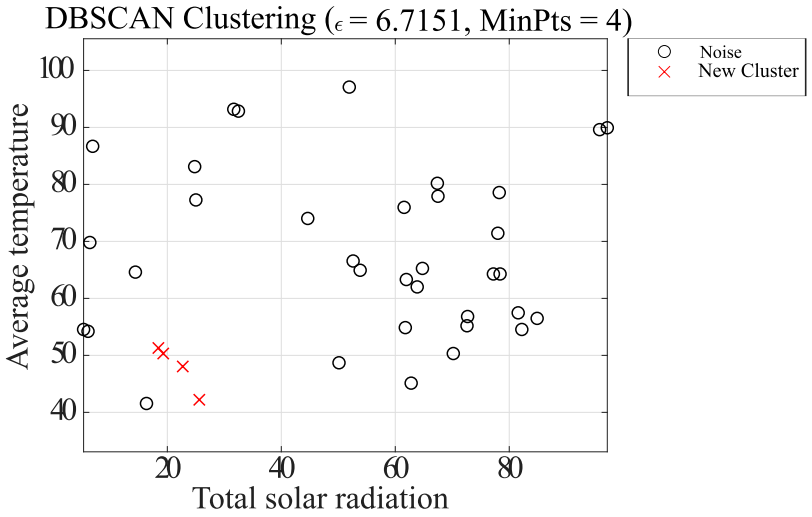


Figure 11. Case 1-6.

- Case 2: Clustering by combination of DBSCAN and k-means method

Case 1 has an improved k-dist plot that automatically determines the radius ϵ and element pts in more detail. However, it is clear that cluster 4 in Figure 12 has a rather broad distribution. This may not be sufficient clustering.

In Case 2, the DBSCAN approach is used to remove noisy elements in the dataset, followed by classification of the elements using the k-means method center-of-gravity update, and finally final clustering using the k-means method. The proposed clustering algorithm is shown below. It is also adapted to clustering using the data combinations shown in Table 1. Figures 13–18, which shows Case 2, uses the same combination of average temperature and total solar radiation as Case 1.

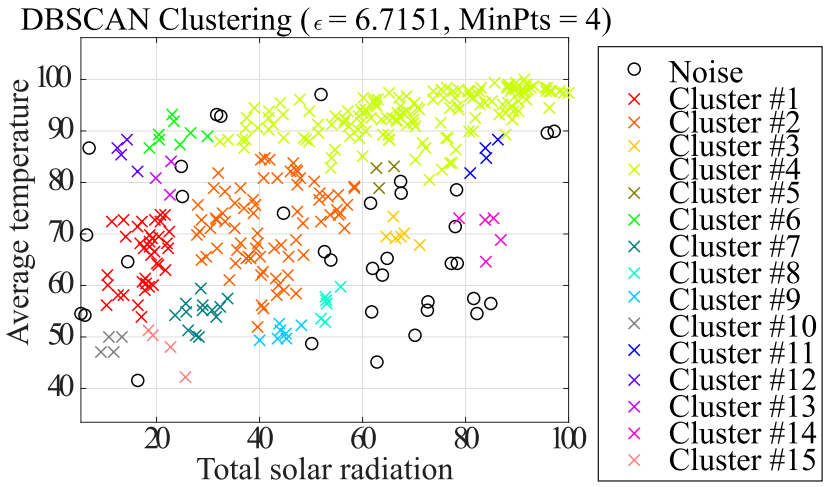


Figure 12. DBSCAN results (after improvement).

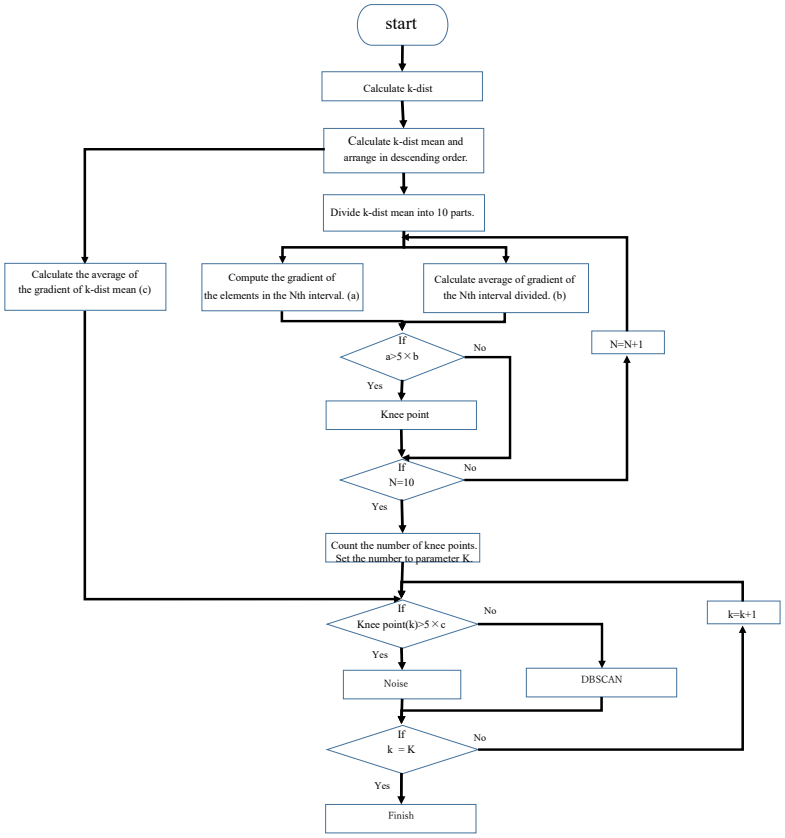


Figure 13. Case 1 flow diagram.

- Removal of noisy elements

1. Plot the points that are the centers of gravity of all the elements on the diagram, with P as the center of gravity.
2. Create a circle centered at point P that covers the entire element, and let r be its radius.
3. The radius r is contracted, and the point S is the point of leakage from the circle.
4. Consider a circle of radius ϵ with respect to S. Find the center of gravity of the elements contained in the circle and plot it on the diagram.
5. Steps 3 to 4 are repeated until there are no more elements. The elements used in step 4 to calculate the center of gravity are excluded from the next step if they have been considered once.
6. For every center of gravity obtained, a judgment is made as to whether it has Pts elements within a radius ϵ, and all elements within the circle of the center of gravity that do not satisfy the condition are considered noise.

- Classification of Elements

7. To classify the remaining centers of gravity and elements, the following method is used for grouping.
   (a) Focusing on the center of gravity farthest from the center of gravity P obtained in step 1, group together the elements contained within the radius ϵ of the center of gravity and make a new grouping.
   (b) Create a new center of gravity from the elements in the summarized group, and create a new group of elements contained within a radius ϵ centered on the center of gravity.
   (c) Repeat updating the center of gravity and groups as above, and save the groups when there are no more updates.

(d) Next, the points away from the center of gravity P are grouped in the same way. At this time, elements that have been grouped once are not considered.

8. If groups have the same elements within the radius of the center of gravity of each other, they are considered as one group. In such a case, the group is considered to be a single group. A group is considered as a group with multiple centers of gravity.
9. Finally, clustering is performed on all groups using the k-means method. The number of clusters k is the number of centers of gravity that the group has. After updating the center of gravity with k-means, the clustering is finished.

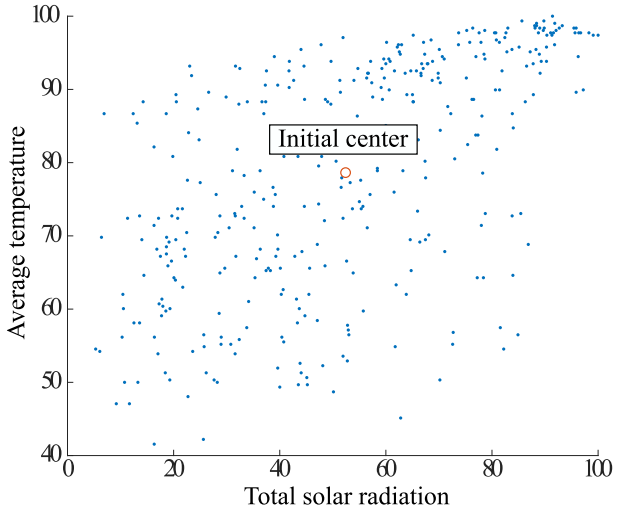


Figure 14. Step 1 result (Initial center:P).

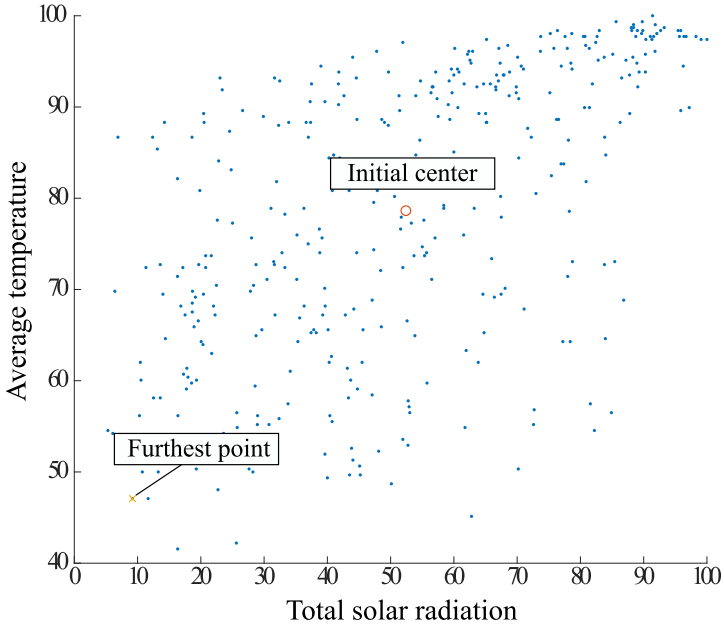


Figure 15. Step 2 and 3 result (Futher point:S).

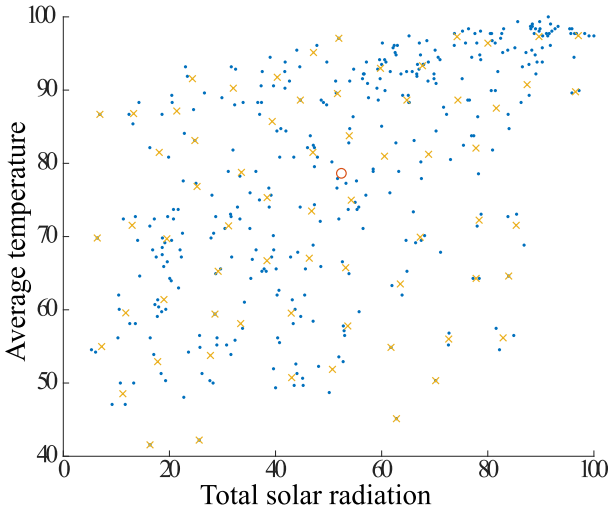


Figure 16. Step 4, 5, and 6 result.

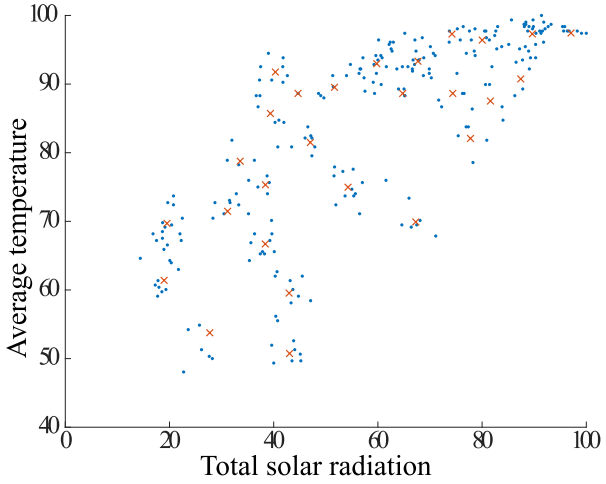


Figure 17. Step 7 result.

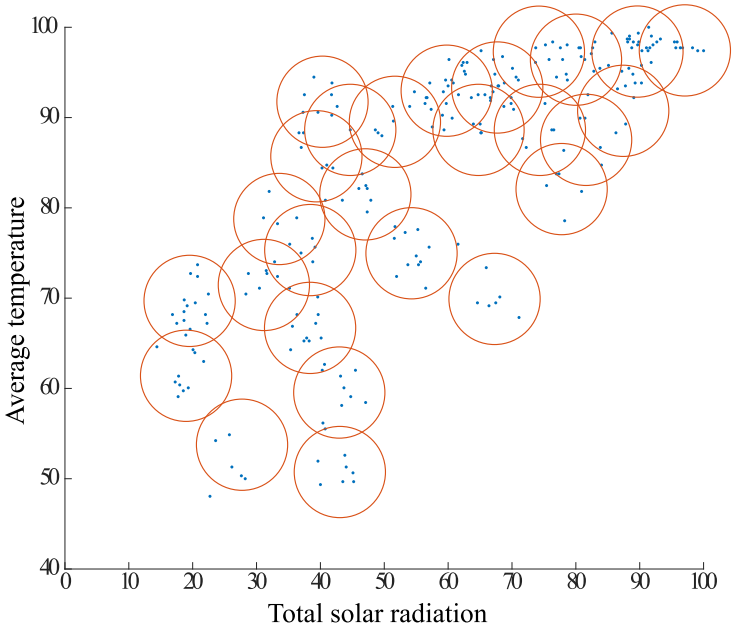


Figure 18. Step 8 result.

Comparing Figures 4 and 19, one can see that a more detailed classification has been made. The flow diagram for Case 2 is shown in Figure 20.

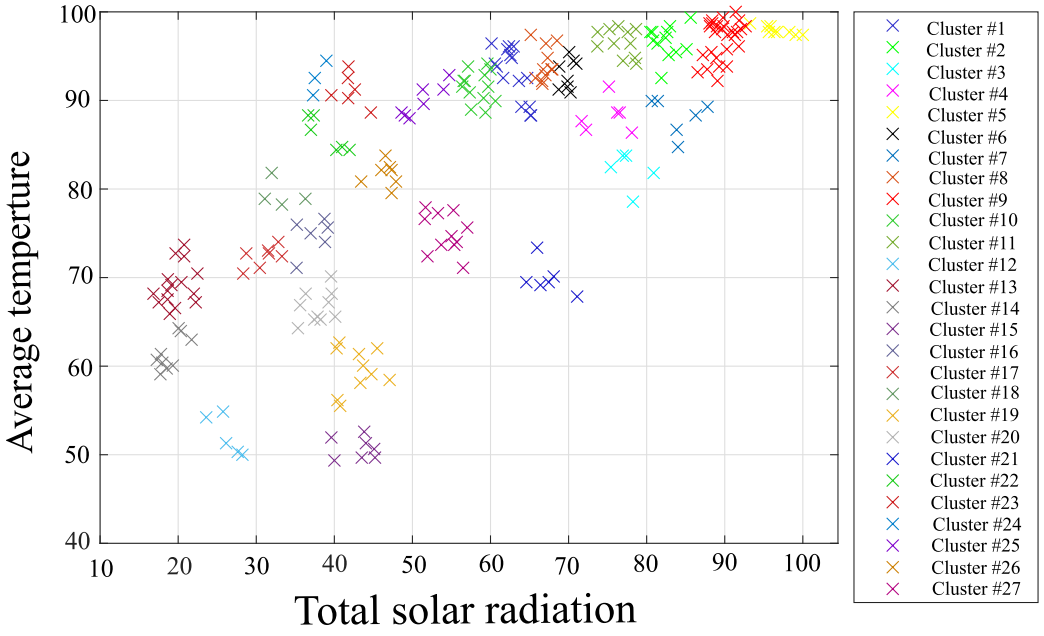


Figure 19. Step 9 result.

Time Complexity

It is known that the time complexity of DBSCAN is determined by the number of calls to the parameters (ϵ,Pts) of each element and is O(n^2). The conventional method, k-dist plot, determines only one parameter and simulates it, so the time complexity is O(n^2). Case 1 of the proposed method increases the number of knee points in the k-dist plot and automatically updates the parameters to perform the simulation. Therefore, the time complexity is O($P \times n^2$) (P: number of knees).

Case 2 of the proposed method once adapts the DBSCAN algorithm on all points, then updates the center of gravity and creates clusters by elements with noise points removed. Therefore, the time computational complexity is O($n^2 + k(n - noise) \times t$) ($k$: number of centers of gravity, n: number of elements, t: number of iterations).

The time computation resulted in a larger time computation than the original DBSCAN, but if we were to cluster the load demand for one year, the elements would be 365 elements.

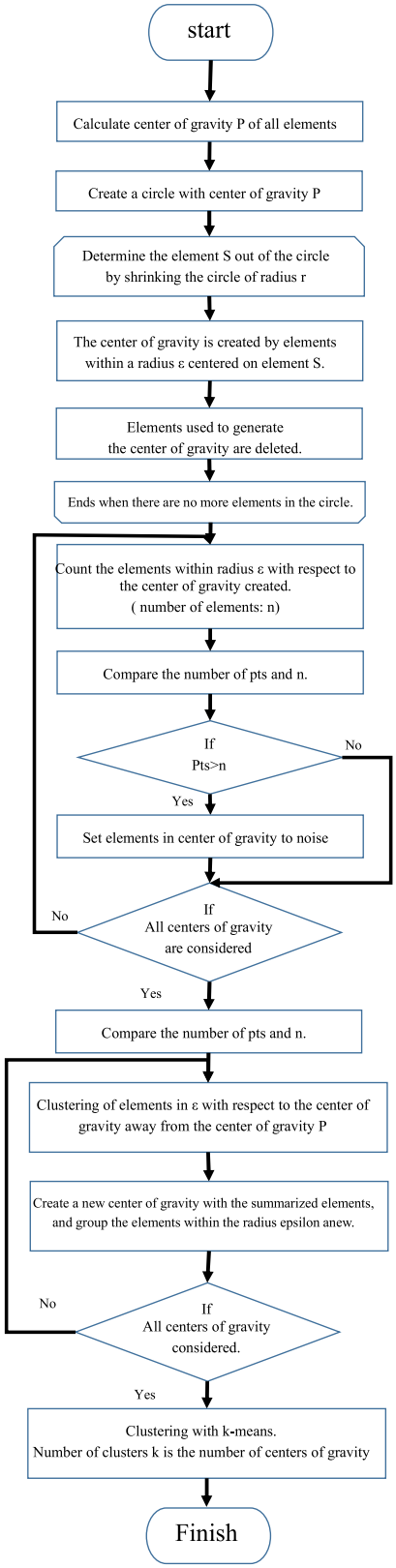


Figure 20. Case 2 flow diagram.

3. Formulation of the Optimization Problem

Consider a one-year generator start-up and shutdown plan to verify the effectiveness of the clustering results. The formulation of the generator start-up and shutdown plan to

be implemented is shown below. The objective function is to minimize the total cost of fuel and start-up costs of thermal generators. The maximum and minimum parameters for each constraint are shown in Tables 2 and 3.These parameters are based on information on generators and transmission lines used in the Okinawa Prefecture.

- objective function

$$Minimize = \sum_{k} \sum_{j} (c^{F}_{k,j} + c^{S}_{k,j}) \tag{2}$$

where $c^{F}_{k,j}$: the fuel cost of generator j at time k and $c^{S}_{k,j}$: the starting cost of generator j at time k. The $c^{F}_{k,j}$ is given by the following equation.

$$c^{F}_{k,j} = A_j v_{k,j} + B_j p_{k,j} + C_j p^{2}_{k,j} \tag{3}$$

where $p_{k,j}$: the output of generator j at time k, A_j, B_j, C_j: the fuel cost coefficient of generator j.

- Constraints
  - Supply and demand balance constraints:
    Constraint that load demand and power supply at each bus bar must be equal.

$$LD_{k,b} = \sum_{j} p_{k,j,b} + PF_{k,b} \tag{4}$$

    where $LD_{k,b}$: load demand on bus b at time k, $p_{k,j,b}$: output of generator j on bus b at time k, $PF_{k,b}$: transmission line tidal flow on bus b at time k.
  - Generator output upper and lower limit constraints:
    The output of the generator must be within a certain range.

$$P^{min}_{j} \leq p_{k,j} \leq P^{max}_{j} \tag{5}$$

    where P^{min}_{j}: minimum output of generator j, P^{max}_{j}: maximum output of generator j.
  - Generator output change rate constraint:
    The rate of change of the generator output in one hour must be within a certain value.

$$|p_{k,j} - p_{k,j-1}| \leq \Delta P^{max}_{j} \tag{6}$$

    where ΔP^{max}_{j}: the maximum output change of generator j.
  - Power line tidal constraints
    The amount of transmission line tidal flow must be within a certain range.

$$-PF^{max}_{s,r} \leq \frac{\theta_{k,s} - \theta_{k,r}}{X_{s,r}} \leq PFs, r^{max} \tag{7}$$

    where $PF^{max}_{s,r}$: maximum capacity of the transmission line between busbars s, r, $X_{s,r}$: reactance of the transmission line between busbars s, r, $\theta_{k,s}, \theta_{k,r}$: transfer angle of busbars s, r at time k.
  - Generator minimum run/stop time constraints:
    If the generator is started, it must not be stopped for the minimum operating time. Conversely, if the generator is stopped, it must not be started for the minimum time.

$$T^{on}_{j} \leq X^{on}_{k,j}(t) \tag{8}$$

$$T^{off}_{j} \leq X^{off}_{k,j}(t) \tag{9}$$

where T_j^{on}, T_j^{off}: minimum operation and shutdown time of generator j, $X_j^{on}(t)$, $X_j^{off}(t)$: time to keep generator j stopped at time k.

– Operating reserve capacity constraint:

$$LD_{k,b} + SR_{k,b} \leq \sum_j (v_{k,j,b} \times p_j^{max}) \tag{10}$$

where $SR_{k,b}$: the operating reserve at time k busbar b.

– Prediction error correspondence constraints:
The following constraint equation is considered in the UC problem, which is a constraint to prevent changes in the start-up and shutdown states of generators determined in the pre-decision phase when errors in PV output occur within $\pm 2\sigma$ in the day's operation plan.

$$D - \mu^{PV} + 2\sigma^{PV} \leq \sum_j (P_j^{max} \times X_j) \tag{11}$$

$$D - \mu^{PV} - 2\sigma^{PV} \geq \sum_j (P_j^{max} \times X_j) \tag{12}$$

where D: load demand forecast, μ^{PV}: PV output forecast, σ^{PV}: standard deviation of PV output forecast error.

Table 2. Generator's parameters.

	P^{max} [MW]	P^{min} [MW]	Δ P [MW]	A [JNY]	B [JNY/MWh]	C [JNY/MWh^2]	SUC [¥]	$T^{on,off}$ [h]
g1-1(Coal)	220	84	44	80,000	4000	0.4	1,100,000	8
g1-2(Coal)	220	84	44	80,000	4000	0.4	1,100,000	8
g2-1(oil)	125	103	62.5	632,000	9200	2.0	375,000	6
g2-2(oil)	103	50	51.5	632,000	9200	2.0	309,000	6
g3-1(Coal)	156	60	31.2	80,000	4000	0.4	780,000	6
g3-2(Coal)	156	60	31.2	80,000	4000	0.4	780,000	6
g4-1(LNG)	251	122	84	132,000	4400	5.0	753,000	8
g4-2(LNG)	251	122	84	132,000	4400	5.0	753,000	8
g4-3(LNG)	35	17	35	132,000	4400	5.0	105,000	4
g5-1(oil)	125	60	62.5	632,000	9200	2.1	375,000	6
g5-2(oil)	60	30	30	632,000	9200	2.1	180.000	4
g5-3(oil)	103	55	51.5	632,000	9200	2.1	309,000	6

Table 3. Power line parameter.

Bus Line	Equipment Capacity [MW]	Operating Capacity [MW]	Available Capacity [MW]
1-2	1208	1032	177
2-3	998	650	73
2-4	1208	650	286
3-4	758	432	157
4-5	1208	650	192
3-6	810	574	102
5-7	1150	650	133
6-7	596	340	110
7-8	520	260	54

4. Power System Model

The small power system model with eight bus lines used in this study is shown in Figure 21. In Figure 21, $L1$-$L4$ represent the load demand, and the respective percentages are 9%, 24%, 23%, and 43%. g_1–g_5 indicate the locations of thermal power generators, and a total of 12 thermal power generators are assumed to be installed, with a total maximum output of 1805 MW. For load demand, we used data for fiscal year 2016 (April 2016–March 2017) published by Okinawa Electric Power Company, Inc. Photovoltaic systems were considered to be installed in the power system, and a total of 270 MW of

photovoltaic systems were assumed to be installed on each load bus line at the same ratio as the load. The solar radiation was predicted by the neural network function of Matlab using data from 2015, and the prediction error from the same day was derived.

This simulation was performed using MILP in the Matlab Tool box. The maximum time for one duration was set as 500 s.

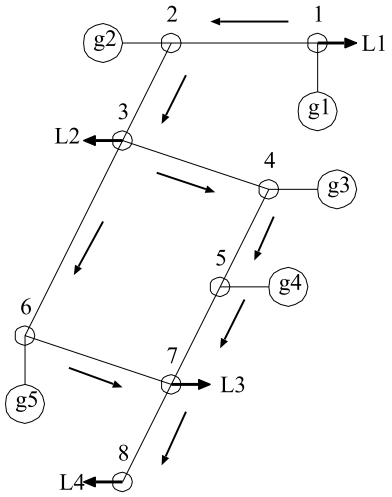


Figure 21. Power system model.

5. Simulation

5.1. Simulation Conditions

To confirm the effectiveness of the proposed method, simulations are performed on representative days created using the Case 1 and Case 2 methods, and the results are compared with the operational cost results of a one-year operational plan that does not use clustering to determine the error and comparison of simulation time. Similarly, a comparison will be made with the k-means method and k-dist plot [27], which are existing methods.

In the k-means method, the number of clusters was determined using the elbow method. The results of the elbow method are shown in Figure 22. The number of clusters was determined to be 18 based on the results in Figure 22.

$$SSE = \sum_{i=1}^{k} (\sum_{j=1}^{A_i} (y_i - y_{i,j}^*))^2 \tag{13}$$

Here, k is the number of clusters, A_i is the total number of data classified as cluster i, y_i is the cluster i, $y^*_{i,j}$ In addition, the j th data classified as cluster i.

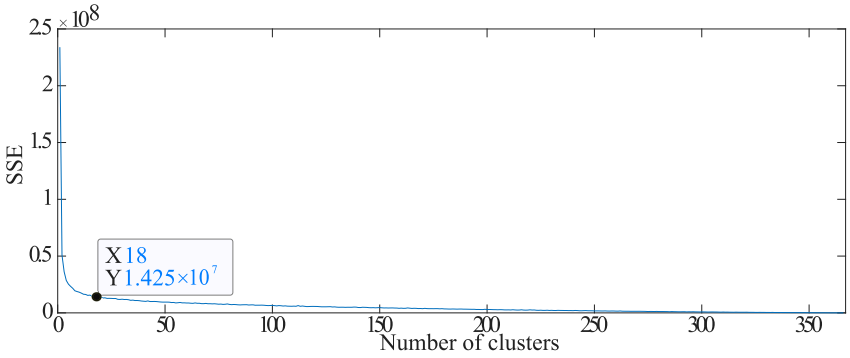


Figure 22. Elbow method.

5.2. Simulation Results

Figure 23 shows the results using case 1, case 2, and clustering by the k-means method. Figure 23 shows and compares the best points in terms of simulation time and cost error among the three cases. The results of the DBSCAN clustering in the elements of Table 1 are the combination of data 4 and data 6 (total solar radiation and average temperature), case 1 is the combination of data 2 and data 12 (total load demand and weekday, weekend and holidays), case 2 is the combination of data 6 and data 9 (average temperature and minimum load hours).

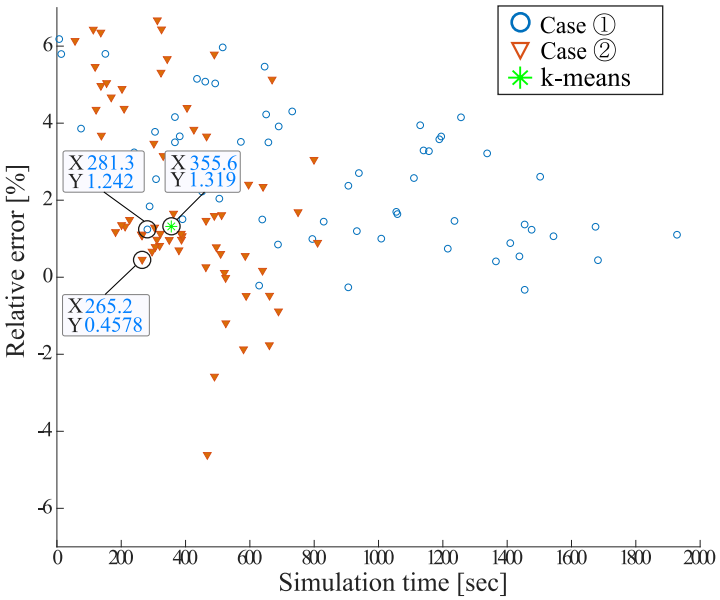


Figure 23. Compare of each result.

Table 4 shows the simulation time and cost error at the best point for each case. The time rate in Table 4 is the percentage of simulation time when the one-year operation is set to 100%. The proposed methods, case 1 and case 2, have more clusters than the conventional method, DBSCAN, resulting in a longer simulation time. However, the cost error can be reduced in both Case 1 and Case 2. The results show that DBSCAN is more cost-effective than DBSCAN. The results clearly show that the clustering is more accurate than the conventional DBSCAN method, and although the simulation time increases slightly, it can be said that the objective of reducing the computation time has been achieved compared to the annual operation. A comparison of the cost error between case 1 and case 2 shows that the cost error is 0.7%. Therefore, case 2 is the superior clustering method between case 1 and case 2, which is the proposed method.

Table 4. Simulation results on the optimal data set.

	Error [%]	Simulation Time [s]	Number of Clusters	Time Error [%]
One year operation	-	9689	-	100
DBSCAN (data4 and 6)	2.7	62	4	0.63
k-means	1.3	356	18	3.6
Case 1 (data2 and 12)	1.2	182	12	1.9
Case 2 (data6 and 9)	0.5	265	18	2.7

5.3. Analysis of Each Data

The results from case 2 are used to analyze which of the data shown in Table 1 are suitable for load-demand pattern classification. Figure 24 shows the averages by error of the operating costs for the 12 types of data. Figure 25 shows their standard deviations. From Figures 24 and 25, the clustering including four of the data cases (2, 5, 6, and 7) results in the

lowest error. These results are related to total load demand and temperature, respectively, and it is appropriate to use these data in classifying load demand patterns. In fact, since load demand is expected to change significantly from season to season, the load demand pattern classification based on temperature and load demand is considered appropriate.

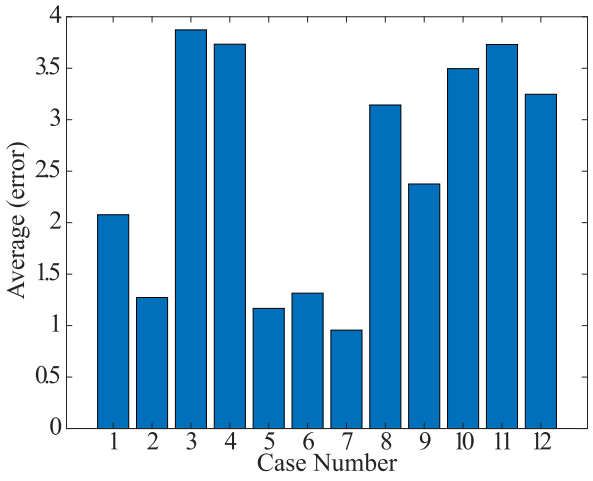


Figure 24. Comparison of each data in Table 1 (average).

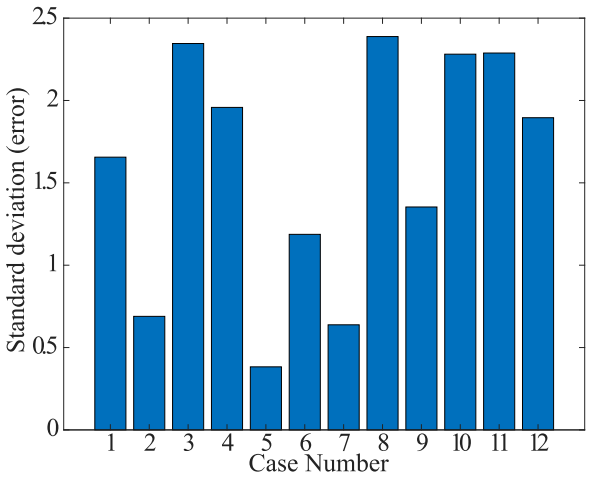


Figure 25. Comparison of each data in Table 2 (standerd deviation).

5.4. Simulation with Storage Batteries Installed

Section 5.2 proves that the proposed method is more accurate than the clustering of conventional methods. In this section, we investigate whether the proposed method is also effective for UC extension problems such as the optimal placement of storage batteries [30–32] using Case 2, which had higher accuracy. The constraints considered are shown below.

$$s_b = s_b(k-1) + ch_b(k)\eta - dis_b(k)/\eta \tag{14}$$

$$0 \leq s_b \leq s_b^{max} \tag{15}$$

where $s_b(k)$: residual energy of the storage battery at time k, s_b^{max}: upper limit of residual power of the storage battery at time k, $ch_b(k)$: charge power of the storage battery at time k, $disc_b(k)$: discharge power of the storage battery at time k, η: charge and discharge efficiency of the storage battery.

$$0 \leq ch_b(k)\eta \leq c_b^{max} u_j \tag{16}$$

$$0 \leq dis_b(k)/\eta \leq c_b^{max} u_j(1-u_j) \tag{17}$$

where c_b^{max}: maximum output rating of the storage battery, u_j: charge/discharge state of the storage battery (1: charge, 0: discharge).

Table 5 shows the status of storage battery placement and capacity according to the one-year simulation results without clustering and the results of the proposed method 2 (average temperature, minimum load time). Comparison of the results from Table 5 shows that the results are close mainly in terms of their placement on bus lines 7 and 8 and in terms of the total number of storage batteries placed. Table 6 shows the cost results of the simulation with the storage battery arrangements and capacities in Table 5. Table 6 shows that the difference in total cost is about JNY 150 million, which is an error of about 0.4% of the total cost, and that the UC simulation including storage batteries in MILP can be performed in a short time with a small error compared to annual operation. These results indicate that clustering using DBSCAN is adaptable to the UC extension problem of optimal placement of storage batteries, and is expected to reduce computation time even for simulations with increased variables.

Table 5. Optimal placement and optimal capacity of storage batteries.

Bus Number	1	2	3	4	5	6	7	8	Total
Base case	0	0	2	0	0	0	44	39	85
Proposed 2	0	0	0	2	0	2	53	23	80

Table 6. Battery placement result.

[Million JNY]	ESS Cost	Operation Cost	Total Cost
Base case	10.88	386.7	397.6
Proposed2	10.24	388.9	399.0

6. Conclusions

In this study, we proposed an improved method of DBSCAN, a method of clustering without prior determination of the number of clusters, for load demand pattern classification. Case 1 automatically determines the number of knees in a k-dist plot, which enabled more detailed clustering. In case 2, a clustering method combining DBSCAN and the k-means method was used to classify patterns of electricity load demand with updated noise and center of gravity. Comparison of each method in the annual operation of UC showed that case 1 and case 2 of the proposed method can reduce the cost error compared to the conventional method. In addition, a comparison of case 1 and case 2 showed that case 2 was superior in terms of both simulation time and cost error. When case 2 was introduced into the extended problem of optimal placement of storage batteries, the cost error was reduced to about 1.4%. By determining the representative date through clustering, it was possible to perform accurate simulations in a shorter time, even when the number of variables and constraints to be considered increased due to the introduction of demand response. Finally, the errors of the input data used from these obtained solutions were checked, respectively, and it was found that clustering by total load demand and temperature factors gave superior clustering results.

As a prospective view, this simulation was clustered using the 2016 data set. However, future grid operations are necessary to achieve the 2030 CO_2 reduction targets established in each country. Reference [33] adapted artificial neural networks (ANN) and an adaptive neurofuzzy inference system (ANFIS) to meta-heuristic algorithms such as genetic algorithms (GA), differential evolution (DE), and particle swarm optimization (PSO) to enable the formulation of future forecast data. Based on the calculated future load demand and generation data, the clustering method proposed in this paper can be used to reduce the computation time.

Author Contributions: Conceptualization, Y.T. and T.S.; data curation, Y.T.; formal analysis, Y.T., E.O. and A.Y.; investigation, Y.T.; methodology, Y.T.; project administration, T.S.; resources, H.T. and T.S.; software, Y.T.; supervision, Y.T. and T.S.; validation, Y.T. and T.S.; visualization, A.Y. and H.T.; writing—original draft, Y.T.; writing—review & editing, N.K. and E.O. All authors have read and agreed to the final version of this manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data sharing not applicable. No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Edmunds, R.; Davies, L.; Deane, P.; Pourkashanian, M. Thermal power plant operating regimes in future British power systems with increasing variable renewable penetration. *Energy Conv. Manag.* **2015**, *105*, 977–985. [CrossRef]
2. Amrutha, A.A.; Balachandra, P.; Mathirajan, M. Role of targeted policies in mainstreaming renewable energy in a resource constrained electricity system: A case study of Karnataka electricity system in India. *Energy Policy* **2017**, *106*, 48–58. [CrossRef]
3. Leonard, M.D.; Michaelides, E.E.; Michaelides, D.N. Substitution of coal power plants with renewable energy sources—Shift of the power demand and energy storage. *Energy Convers. Manag.* **2018**, *164*, 27–35. [CrossRef]
4. Muralikrishnan, N.; Jebaraj, L.; Rajan, C.C.A. A Comprehensive Review on Etionary Optimization Techniques Applied for Unit Commitment Problem. *IEEE Access* **2020**, *8*, 132980–133014. [CrossRef]
5. Li, X.; Zhai, Q.; Zhou, J.; Guan, X. A variable reduction method for large-scale unit commitment. *IEEE Trans. Power Syst.* **2020**, *35*, 261–272. [CrossRef]
6. Guan, X.; Luh, P.B.; Yan, H.; Amalfi, J.A. An Optimization—Based Method for Unit Commitment. *Electr. Power Energy Syst.* **1992**, *14*, 9–17. [CrossRef]
7. Howlader, H.O.R.; Sediqi, M.M.; Ibrahimi, A.M.; Senjyu, T. Optimal Thermal Unit Commitment for Solving Duck Curve Problem by Introducing CSP, PSH and Demand Response. *IEEE Access* **2018**, *6*, 4834–4844. [CrossRef]
8. Zhu, Y.; Gao, H. Improved Binary Artificial Fish Swarm Algorithm and Fast Constraint Processing for Large Scale Unit Commitment. *IEEE Access* **2020**, *8*, 152081–152092. [CrossRef]
9. Carrion, M.; Arroyo, J.M. A computationally efficient mixed–integer linear formulation for the thermal unit commitment problem. *IEEE Trans. Power Syst.* **2006**, *21*, 1371–1378. [CrossRef]
10. Bragin, M.A.; Luh, P.B.; Yan, B.; Sun, X. A scalable solution methodology for mixed–integer linear programming problems arising in automation. *IEEE Trans. Autom. Sci. Eng.* **2019**, *16*, 531–541. [CrossRef]
11. Viana, A.; PedroPedroso, J. A new MILP-based approach for unit commitment in power production planning. *Electr. Power Energy Syst.* **2013**, *44*, 997–1005. [CrossRef]
12. Alemany, J.; Kasprzyk, L.; Magnago, F. Effects of binary variables in mixed integer linear programming based unit commitment in large–scale electricity markets. *Electr. Power Syst. Res.* **2018**, *160*, 429–438. [CrossRef]
13. Balasubramanian, S.; Balachandra, P. Characterising electricity demand through load curve clustering: A case of Karnataka electricity system in India. *Comput. Chem. Eng.* **2021**, *150*, 107316 . [CrossRef]
14. Marton, C.H.; Elkamel, A.; Duever, T.A. An order-specific clustering algorithm for the determination of representative demand curves. *Comput. Chem. Eng.* **2008**, *32*, 1365–1372. [CrossRef]
15. Ibs-von Seht, M. Detection and identification of seismic signals recorded at Krakatau ano (Indonesia) using artificial neural networks. *J. Anology Geotherm. Res.* **2008**, *178*, 448–456. [CrossRef]
16. Lei, M.; Shiyan, L.; Chuanwen, J.; Hongling, L.; Yan, Z. A review on the forecasting of wind speed and generated power. *Renew. Sustain. Energy Rev.* **2009**, *13*, 915–920. [CrossRef]
17. MacQueen, J.B. Some Methods for Classification and Analysis of Multivariate Observation. In Proceedings of the Berkeley Symposium on Mathematical Statistics and Probability 5.1, Berkeley, CA, USA, 7 January 1967; pp. 281–297.
18. Kaur, P.; Goyal, M.; Lu, J. Data mining driven agents for predicting online auction's end price. In Proceedings of the 2011 IEEE Symposium on Computational Intelligence and Data Mining (CIDM), Paris, France, 11–15 April 2011; p. 141.
19. Understanding Clustering with the K-Means Algorithm in Machine Learning: (With Examples). Available online: https://www.analyticsvidhya.com/blog/2021/11/understanding-k-means-clustering-in-machine-learningwith-examples/ (accessed on 20 January 2023).
20. Luo, X.; Zhu, X.; Lim, E.G. A parametric bootstrap algorithm for cluster number determination of load pattern categorization. *Energy* **2019**, *180*, 50–60. [CrossRef]
21. Battaglia, O.R.; Paola, B.D.; Fazio, C. A New Approach to Investigate Students' Behavior by Using Cluster Analysis as an Unsupervised Methodology in the Field of Education. *Appl. Math.* **2016**, *7*, 141–147. [CrossRef]
22. Nobis, M.; Schmitt, C.; Schemm, R.; Schnettler, A. Pan-European CVaR-Constrained Stochastic Unit Commitment in Day-Ahead and Intraday Electricity Markets. *Energies* **2020**, *13*, 2339. [CrossRef]

23. Ocampo, E.; Huang, Y.; Kuo, C.-C. Feasible Reserve in Day-Ahead Unit Commitment Using Scenario-Based Optimization. *Energies* **2020**, *13*, 5239. [CrossRef]
24. Heuberger, C.F.; Staffell, I.; Shah, N.; Mac Dowell, N. A systems approach to quantifying the value of power generation and energy storage technologies in future electricity networks. *Comput. Chem. Eng.* **2017**, *107*, 247–256 [CrossRef]
25. Khan, K.; Rehman, S.U.; Aziz, K.; Fong, S.; Sarasvady, S. DBSCAN: Past, present and future. In Proceedings of the ICADIWT, Bangalore, India, 17–19 February 2014; pp. 232–238.
26. Ester, M.; Sander, H.K.J.; Xu, X. A density-based algorithm for discovering clusters in large spatial databases. In Proceedings of the KDD '96 Proceeding of the Second International Conference on Knowledge Discovery and Data Mining, Oregon, Portland, 2–4 August 1996; pp. 226–231.
27. Gaonkar, M.N.; Sawant, K. Auto Eps DBSCAN: DBSCAN with Eps automatic for large dataset. *Int. J. Adv. Comput. Theory Eng.* **2013**, *2*, 11–16.
28. Japan Meteorological Agency. Available online: https://www.data.jma.go.jp/obd/stats/etrn/index.php (accessed on 20 January 2023).
29. Okinawa Electric Power | Electricity Forecast. Available online: https://www.okiden.co.jp/denki2/dl (accessed 20 January 2023).
30. Pandi, H.; Wang, Y.; Qiu, T.; Dvorkin, Y.; Kirschen, D.S. Near-Optimal Method for Siting and Sizing of Distributed Storage in a Transmission Network. *IEEE Trans. Power Syst.* **2015**, *30*, 2288–2300.
31. Aoyagi, H.; Isomura, R.; Mandal, P.; Krishna, N.; Senjyu, T.; Takahashi, A.H. Optimum Capacity and Placement of Storage Batteries Considering Photoaics. *Sustainability* **2019**, *11*, 2556. [CrossRef]
32. Blanco, R.F.; Dvorkin, Y.; Xu, B.; Wang, Y.; Kirsche, D.N. Optimal energy storage siting and sizing: A WECC case study. In Proceedings of the 2017 IEEE Power & Energy Society General Meeting, Chicago, IL, USA, 17 July 2017; p. 1.
33. Çunkaş, M.; Altun, A.A. Long Term Electricity Demand Forecasting in Turkey Using Artificial Neural Networks. *Energy Sources Part Econ. Plan. Policy* **2010**, *5*, 279–289. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.