

Hierarchical Trust-Tech-Enhanced K-Means Methods and Their Applications to Power Grids

HSIAO-DONG CHIANG¹ (Life Fellow, IEEE), TIAN-SHI XU²,
 XIAN-LONG LV², AND NA DONG¹

¹School of Electrical and Computer Engineering, Cornell University, Ithaca, NY 14853 USA

²School of Electrical and Information Engineering, Tianjin University, Tianjin 300072, China

CORRESPONDING AUTHOR: N. DONG (dongna@tju.edu.cn)

ABSTRACT K-means has been widely used in solving a wide range of clustering problems arising in engineering and industrial applications, but it still suffers from several issues. To address these issues, a hierarchical K-means method enhanced by Trust-Tech (H-KTT) is presented in this paper. The proposed H-KTT method is composed of two stages. The first stage of H-KTT is a hierarchical K-means (H-K-means) method for enhancing K-means with better initial points. Second, the H-K-means method is further enhanced to find multiple high-quality clustering results by the Trust-Tech methodology. The H-KTT method was evaluated on several test datasets including the clustering of Automatic Meter Reading (AMR), popular in power grids, with promising results. In particular, the evaluation results indicate that the proposed H-KTT method can significantly improve both the quality and stability of the clustering results by the K-means method. Furthermore, while the K-means gives stochastic clustering results, the proposed H-KTT method usually gives deterministic clustering results.

INDEX TERMS K-means, local optimal solution, hierarchical approach, Trust-Tech, dynamical system, electric grids, smart meters.

I. INTRODUCTION

ELECTRIC demand clustering can extract the power consumption patterns as well as the characteristics of electric users and can be very useful for consumer classification. Moreover, electric load clustering helps utilities better implement their energy policy and infrastructure planning strategies. Important applications of electric load clustering include the tariff design, anomaly detection, load forecasting, data security, and big data [1].

Clustering techniques such as the K-means algorithm and its variants have been applied to a variety of problems in power grids such as the following: (1) anomaly detection of online monitoring data of power equipment based on association rules and the clustering algorithm [6], [7], (2) development of day-ahead and hour-ahead bus load forecasting models [2], (1) forecasting hourly global solar radiation [3], (2) multi-resolution load profile clustering for smart metering data [4], (3) clustering load profiles for demand response applications [5], [8], (4) clustering load patterns with

different consumption behaviors for market strategies design [9], and (7) clustering load patterns with different consumption behaviors customer classification [10].

Clustering has become a practical technology in many applications. The prosperity of data mining technology further promotes the applications of clustering algorithms to a greater extent. Indeed, the K-means type of clustering algorithms are popular in real-world applications due to their ability to handle both numeric and categorical variables and their speed. For instance, the K-means algorithm has been applied to the clustering of mRNA databases, the automatic segmentation of brain tumors, the detection of forest fire smoke, single channel separation from mixed signals, digital pulse compression, subsequence clustering, and the management of large sizes of 3D data collections used for segmentation under image rotation, to name a few (see, for example, [11], [12], [13], [14], [15], [16], [17], [18], [19]).

While many clustering problems are usually nonlinear and non-convex, the K-means algorithm almost always converges

to a local ‘optimal’ solution. Moreover, the random initialization embedded in the K-means algorithm greatly affects its ability to deliver stable clustering results. It is well recognized that the K-means algorithm suffers from the following issues: (i) it is very sensitive to the initial points, (ii) it gives a local optimal solution, (iii) it is stochastic in obtaining a clustering solution, and (iv) its clustering results have poor stability.

In the past, a significant amount of research was directed toward improving the K-means type of clustering algorithms. Several meta-heuristics-based methods, such as the Genetic Algorithm, Particle Swarm Optimization (PSO), have been applied with the aim of providing high-quality initial points for the K-means type of algorithms. Due to the randomness of K-means algorithm and that of the Meta-heuristic-based method, the quality of the final clustering results remains unsatisfactory, and the issue of poor stability still remains. On the other hand, it is well recognized that the classical K-means algorithm is sensitive to the initial centroids, making the probability of finding appropriate initial centroids leading to high-quality clustering results low for large datasets. Hence, enhancing the performance of classical K-means for clustering large datasets is necessary (see, for example, [13], [14], [15], [16], [17], [18], [19], [20], [21]).

Another approach to enhance the K-means performance explores the hierarchical structure of AMI. In [29], a hierarchical K-means (H-K-means) clustering method was developed. This H-K-means method is different from the previously proposed “hierarchical K-means” methods. In previous studies, the “hierarchical” is based on the perspective of methodology, which usually refers to the aforementioned hierarchical clustering method. For example, the hierarchical clustering method is combined with K-means in different ways in [20] and [21]. The “hierarchical” is explored from the perspective of data in [29], which means the establishment of a hierarchical data structure before the process of K-means clustering. On this basis, the proposed H-K-means method was developed. In addition, extensive numerical studies on a large-scale AMI dataset have shown that the proposed H-K-means method possesses the following advantages: (1) it can significantly improve the quality of the clustering results given by classical K-means, (2) it can preserve the inherent speed advantage of classical K-means, (1) it is especially applicable to big data problems, and (2) it can effectively cluster large-scale load demand curves.

In this paper, the H-K-means method is extended to incorporate the capability of Trust-Tech methodology in computing a set of high-quality solutions; in particular, an integrated methodology, termed the H-KTT (hierarchical K-means enhanced by Trust-Tech), is developed. The Trust-Tech (TRansformation Under STability-reTaining Equilibria CHaracterization) methodology [22], [23], [24], [25], [26] is applied to find multiple high-quality clustering results. The resulting method, termed the H-KTT method, improves not only the quality of the clustering (by effectively providing multiple local optimal solutions) but also the stability of the clustering results via a systematic and deterministic

procedure. The Trust-Tech method has been shown to have the capability of enhancing other optimization methods to achieve better optimization results such as the EM method in [26], the PSO method in [27], and the branch and bound method in [28].

In summary, the proposed H-KTT methodology is developed to achieve the following desired capability:

- (i) It obtains multiple high-quality local optimal solutions (as compared with variants of K-means) or even the global optimal solution. This capability is enabled by the Trust-Tech methodology;
- (ii) It possesses stability in obtaining high-quality clustering results that are enabled by both the Trust-Tech method and by the hierarchical method;
- (iii) While the K-means gives stochastic clustering results, the proposed H-KTT method usually gives deterministic clustering results;
- (iv) It is effective and efficient in solving large-scale clustering problems that are enabled by the K-means algorithm, the Trust-Tech method, and by the hierarchical method.

To demonstrate that the proposed H-KTT can deliver high-degree stable clustering results, we employ the disagreement value as the measure to evaluate its stability in obtaining clustering results. In the numerical experimental results section, we will highlight the improvement made by the H-KTT method, as compared with several other methods that were proposed to enhance the K-means algorithm. It will be further shown in the numerical comparison studies that the proposed H-KTT method indeed obtains high-quality clustering results in a more stable manner. The experimental dataset we used to evaluate the proposed H-KTT method comes from the University of California Irvine datasets: the User Knowledge Modeling dataset, the Car Evaluation dataset, the Synthetic Control Chart time series dataset, and the Hill-Valley dataset [30]. Applications of the proposed H-KTT to power grids include (1) anomaly detection of online monitoring data of power equipment, multi-resolution load profile clustering, clustering load profiles for demand response applications, and clustering load patterns for market strategies design.

The rest of the paper is organized as follows. Section II gives the overview of the proposed method. Section III elaborates on the Trust-Tech methodology. Section IV presents the proposed Hierarchical K-Means enhanced by the Trust-Tech (H-KTT) method. Section V demonstrates the mechanism of locating multiple clustering solutions by Trust-Tech method. In Section VI, the data sets, server configurations, models, and evaluation indicators required for the experiment are introduced and the experimental results are analyzed. Finally, Section VII summarizes the paper and points out the future research directions and existing problems.

II. OVERVIEW OF THE PROPOSED METHOD

The clustering result of the K-means algorithm is not deterministic due to its random selection of initial centroids. Moreover, the K-means algorithm usually gets trapped near

a local optimal solution. These two problems become more pronounced when dealing with large datasets.

We propose to integrate the following two effective methods to enhance the K-means algorithm:

(i) Apply a hierarchical scheme to provide higher-quality initial points for the K-means algorithm to local optimal solutions; the resulting scheme is termed the hierarchical K-means algorithm (H-K algorithm) [31];

(ii) Apply the Trust-Tech method to each local optimal solution obtained to compute a set of high-quality optimal solutions.

For a given dataset, we can treat those points that are close to each other as a representative point with their centroid; then a reduced dataset (i.e., level-1 dataset) consisting of all the representative points can be obtained. The clustering result of this level-1 dataset by the K-means method can be expressed by the centroids of all the clusters. It is more effective to use these centroids as the initial state to the K-means algorithm for clustering of the original dataset instead of the ones obtained by random selection. On the basis of the above procedure, a given dataset can be clustered into several hierarchical levels, as shown in Fig. 1.

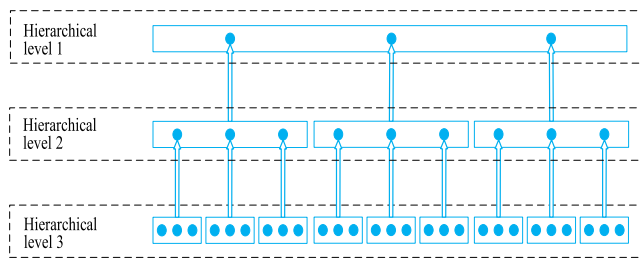


FIGURE 1. A hierarchical scheme generating multi-level reduced datasets.

Each hierarchical level consists of all the representative points of its next level except for the last level, which consists of all the data points from the original dataset. Then, at each hierarchical level, we can implement the K-means algorithm using the clustering result of its previous level as the initial state. Let the number of levels L be defined by the user or some criteria. The 1st level is the original dataset, and each subsequent level is constituted by a smaller dataset of its previous level. Based on this multilevel structure, the H-K-means method can be described by the following steps:

Step 1)

Set the original dataset as the 1st-level dataset and start from $i = 1$.

Step 2)

Establish the i th-level dataset based on the $(i - 1)$ th-level dataset (please see the latter part for details).

Step 3)

If $i = L$, go to Stage II; otherwise $i = i + 1$ and go to Step 2).

Stage II: Weighted clustering

Step 4)

If $i = L$, select K patterns among the i th-level dataset randomly; otherwise, use the K -obtained centroids in Step 5) instead.

Step 5)

Implement K-means clustering for the i th-level dataset and use the patterns given by Step 4) as the initial centroids. During each epoch of the K-means, calculate the centroid of each cluster

Step 6)

If $i = 1$, terminate the process and output the clustering results in Step 5) as the final results; otherwise, $i = i - 1$ and go to Step 4).

This process is then iteratively applied from the top level to the bottom one, where the original dataset has been clustered. This is the so-called hierarchical K-means method (HK-means method), which can achieve better clustering results by supplying better initial points than the ones given by a random selection for the K-means algorithm, as shown in [29]. Moreover, dealing with the simplified datasets from upper levels can reduce the computational burden to a certain extent as compared with the same number of iterations in the original dataset. Note that we use the K-means clustering to generate the corresponding simplified dataset for each hierarchical level.

A step-by-step description of the proposed H-KTT is presented below.

Stage 1 (Hierarchical datasets): Given a dataset and a user-defined number of levels, build the hierarchical datasets consisting of the Level 1 dataset, Level 2 dataset, and so on. Set $i = 1$ and set an initial guess.

Stage 2 (K-means stage): Starting from each given initial guess, apply the K-means method to the Level i dataset to obtain a fast clustering solution. When there are multiple initial guesses, multiple clustering solutions exist (when there are multiple initial guesses).

Stage 3 (Trust-Tech stage): Starting from each solution obtained at Stage 2, apply the Trust-Tech method to the Level i dataset to find a set of local optimal solution (i.e., a set of high-quality clustering solutions). If Level i is the original dataset, then stop, output the clustering solutions, and select the best one; otherwise, go to the next stage

Stage 4 (interface stage): Set $i = i + 1$ and set each clustering solution obtained at Stage 3 as an initial guess (or set the best clustering solution as the initial guess, depending on the user's preference) and go to Stage 2.

The proposed H-KTT method, composed of the above four stages integrating the K-means hierarchical method and the Trust-Tech method are able to compute a set of high-quality clustering solutions that may contain the global optimal one.

III. TRUST-TECH METHODOLOGY: ANALYTICAL ASPECTS

We propose to integrate the Trust-Tech method to improve the local optimal solutions obtained by the H-K-means method because of its following distinguishing features:

1) It can move from a local optimal solution obtained by the K-means algorithm and find a set of higher-quality local optimal solutions that may contain the global optimal solution.

2) It can reduce the effect of the random initialization of the K-means algorithm and find a set of high-quality local optimal solutions rather than the random ones.

The methodology Trust-Tech is a dynamical method designed to systematically compute multiple, local optimal solutions in a tier-by-tier manner [26], [27], [28], [29]. It includes the steps of first finding, in a deterministic manner, one local optimal solution, starting from an initial point, and then finding the nearby first-tier local optimal solution, starting from the previously found one, then finding the second-tier local optimal solutions until multiple or all the local optimal solutions are found. Finally, a high-quality local optimal solution, which can be the global optimal solution, is identified from the found local optimal solutions [26], [27], [28], [29].

From a theoretical viewpoint, the Trust-Tech methodology is built on the optimization theory, nonlinear dynamical systems, stability region theory, and characterization of the stability boundary. We explain the Trust-Tech framework in solving the following unconstrained nonlinear programming problem. Without loss of generality, an n -dimensional optimization problem can be formulated as:

$$\min_{x \in \mathfrak{N}^n} C(x) \quad (1)$$

where $C : \mathfrak{N}^n \rightarrow \mathfrak{R}$ is a function bounded below and possesses only finite local optimal solutions.

Instead of solving the unconstrained optimization problem described by (1) directly, we consider the corresponding dynamical system:

$$\dot{x}(t) = -\nabla C(x) \quad (2)$$

where $x \in \mathfrak{N}^n$. Recall that for a hyperbolic equilibrium point of (2), it is an (asymptotically) stable equilibrium point (SEP) if all the eigenvalues of its corresponding Jacobian have negative real parts; otherwise, it is an unstable equilibrium point (UEP). A hyperbolic equilibrium point \bar{x} of system (2) is called as **type-k** equilibrium point if the Jacobian matrix has $n - k$ eigenvalues with a negative real part and k eigenvalues with a positive real part. A hyperbolic equilibrium point \bar{x} of system (2) is called the source if it is of **type-n**. Hence, a source is an EP whose corresponding Jacobian matrix $Df(\bar{x})$ has all eigenvalues with a positive real part.

A useful concept for SEP is its stability region (also called the region of attraction). The stability region of a SEP is defined as the collection of state vectors whose corresponding trajectories converge to the SEP [24]. It has been shown that the Trust-Tech methodology performs the following transformations [25]:

(i) Transformation of a local optimal solution of a nonlinear optimization problem (1) into a stable equilibrium point (SEP) of a continuous nonlinear dynamical system (2) (also see Theorem 1 below).

Theorem 1 (Equilibrium Points and Local Optimum [22], [23]): If \bar{x} is a hyperbolic equilibrium point of gradient system (2), then \bar{x} is a SEP of system (2) if and only if $C(x)$ has an isolated minimum of the optimization problem (1).

(ii) Transformation of the search space of nonlinear optimization problems (1) into the union of closed stability regions of all the SEPs. Hence, the optimization problem (i.e., the problem of finding local optimal solutions) is transformed into the problem of finding stable equilibrium points.

Hence, the stability regions of SEPs play an important role in finding these local optimal solutions, as shown in the following theorem.

Theorem 2 (Characterization of the Stability Boundary [22], [23]): Suppose that all the equilibrium points of gradient system (2) are hyperbolic. Let $x_i, i = 1, 2, \dots$ be the equilibrium points on the stability boundary of a SEP, say x_s . Then, the stability boundary is contained in the union of the stable manifolds of the equilibrium points on the stability boundary.

Theorem 1 characterizes the relationship between the optimal solutions of the unconstrained optimization problem (1) and the SEPs of its corresponding dynamical system (2). Because of such correspondence, the problem of computing multiple local optimal solutions of the optimization problem (1) is then transformed into finding multiple SEPs of gradient system (2).

Before describing a procedure to locate multiple local optimal solutions, the following theorem shows that all the decomposition points can serve as the bridge linking two SEPs. Note that the decomposition point is a type-one unstable equilibrium point lying on the stability boundary with the following property.

Theorem 3 (Existence of Another Optimal Solution [23], [26]): Let x_s^1 be a stable equilibrium point (SEP) of dynamical system (2). If the stability boundary of the SEP exists and x_d is a decomposition point on the stability boundary, then there exists another SEP x_s^2 to which the one-dimensional unstable manifold of x_d converges.

Theorem 3 reveals a relationship between SEPs and decomposition points. The unstable manifold of a dynamic decomposition point converges to two SEPs (i.e., two local optimal solutions) of dynamical system (2). It asserts that two neighboring local optimal solutions are connected by the unstable manifold of the corresponding dynamic decomposition point.

Next, we present a Trust-Tech-based Dynamic Decomposition Point method (DDP) [23] for locating another local optimal solution from a local optimal solution of the unconstrained optimization problem (1) and proceed with the following key steps:

Step 1: (local method) A local optimal solution, say x_s , can be found by a local optimization method, such as an interior point method, a gradient-based method, or an SPQ method.

Step 2: (DDP method for escaping from a local optimal solution) Starting from an initial stable equilibrium point x_s (i.e., a local optimal solution), the TT moves along

a (given or desired) deterministic direction to find the corresponding dynamic decomposition point.

Step 3: (DDP method for another local optimal solution) Starts from the dynamic decomposition point and moves along the unstable manifold of the dynamic decomposition point, which will lead to another local optimal solution.

We note that, for a given local optimal solution, its corresponding first-tier local optimal solutions are defined as those optimal solutions whose corresponding stability boundaries have a non-empty intersection with the stability boundary of the local optimal solution [25], [26]. Similarly, its second-tier local optimal solutions are defined as those optimal solutions whose corresponding stability boundaries have a non-empty intersection with the stability boundary of first-tier local optimal solutions [25], [26]. See Fig. 2 for an illustration.

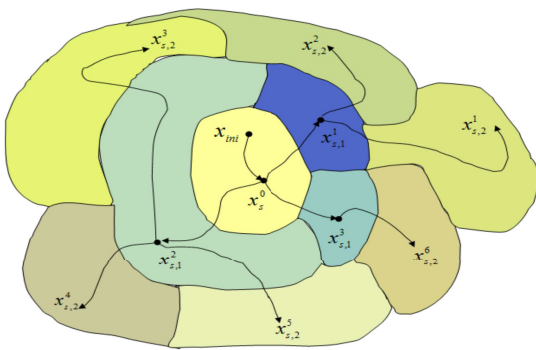


FIGURE 2. Given a local optimal solution (i.e., a tier-zero local optimal solution), the corresponding tier-1 local optimal solutions are $x_{s,1}^1, x_{s,1}^2, x_{s,1}^3$. Similarly, its tier-2 local optimal solutions are $x_{s,2}^1, x_{s,2}^2, x_{s,2}^3, x_{s,2}^4, x_{s,2}^5, x_{s,2}^6$. Note that the corresponding stability boundaries of tier-2 have a non-empty intersection with the stability boundaries of its tier-1 local optimal solutions.

IV. K-MEANS ENHANCED BY THE TRUST-TECH METHOD

We are now in a position to present the proposed Hierarchical K-Means enhanced by the Trust-Tech (H-KTT) method. The proposed H-KTT method is composed of three basic stages: (1) hierarchical datasets contain multi-level datasets, (2) the K-means method, which is designed for computing a fast local solution, and (1) the Trust-Tech method, which is designed for computing a set of high-quality multiple local optimal solutions from which the best local optimal solution is selected. (Alternatively, the top two best clustering solutions are selected.) These clustering solutions are mapped back to the next-level dataset (as designed in the hierarchical datasets of Stage 1) to be used as the initial guess for the K-means to cluster “this-level” dataset to obtain a good clustering solution of “this level” dataset. Again, the good clustering solution obtained is then sent to Stage 3 to be used as the initial condition for the Trust-Tech method to compute a set of high-quality clustering solutions. Then, the entire search procedure moves to the next level of the dataset

designed in Stage 1 and Stage 2 and Stage 3 are repeated (see Fig. 3 and Fig. 4 for an illustration).

To classify a dataset with n points (with d dimensions) into k clusters, the K-means clustering can be formulated as a nonlinear constrained optimization problem, which can be improved by Trust-Tech methodology as follows:

$$\begin{aligned} \min f(x) \\ \text{s.t. } h(x) = 0 \end{aligned} \quad (3)$$

where x is the variable vector consisting of integer variables and continuous variables, $f(x)$ is the objective function to be optimized, and $h(x)$ is the equality constraints of the optimization problem.

The clustering information of the dataset is expressed by the integer variables in a $k \times n$ matrix.

$$\begin{bmatrix} x_1 & x_2 & \cdots & x_n \\ x_{n+1} & x_{n+2} & \cdots & x_{n+n} \\ \vdots & \vdots & \vdots & \vdots \\ x_{(k-1)n+1} & x_{(k-1)n+2} & \cdots & x_{(k-1)n+n} \end{bmatrix}_{k \times n} \quad (4)$$

This kind of variable is used to indicate the centroid to which the points will associate. The integer variable in row i , column j , for instance, indicates that the j th point in the dataset belongs to the i th centroid when it equals 1 or indicates that the point does not belong to the centroid.

The pattern number of each cluster is expressed by the integer variable in a $1 \times k$ vector:

$$[x_{k \times n+1}, x_{k \times n+2}, \dots, x_{k \times n+k}]_{1 \times k} \quad (5)$$

These variables are also used to identify the total number of points in the clusters. The integer variable in column j , for instance, represents the total number of points in the j th cluster.

A. OBJECTIVE FUNCTION

1) TOTAL DISTANCE

The K-means algorithm minimizes the total distance from each point (of each cluster) to its centroid to separate the points with low similarity from the others. Here we take the Euclidean distance metric as an example. In general, the quality of the clustering results given by classical K-means can be measured by an objective function such as, among others, the summation of the square of the Euclidean distance between each pattern and its centroid:

$$f = \sum_{k=1}^K \sum_{i=1}^{n_k} |x_i^k - \omega_k|^2 \quad (6)$$

where K is the number of clusters, ω_k is the centroid of the k th cluster, n_k is the number of patterns belonging to the k th cluster, and x_i^k is the i th pattern belonging to the k th cluster.

To separate N patterns into K clusters, the basic procedure of the classical K-means algorithm is presented as follows.

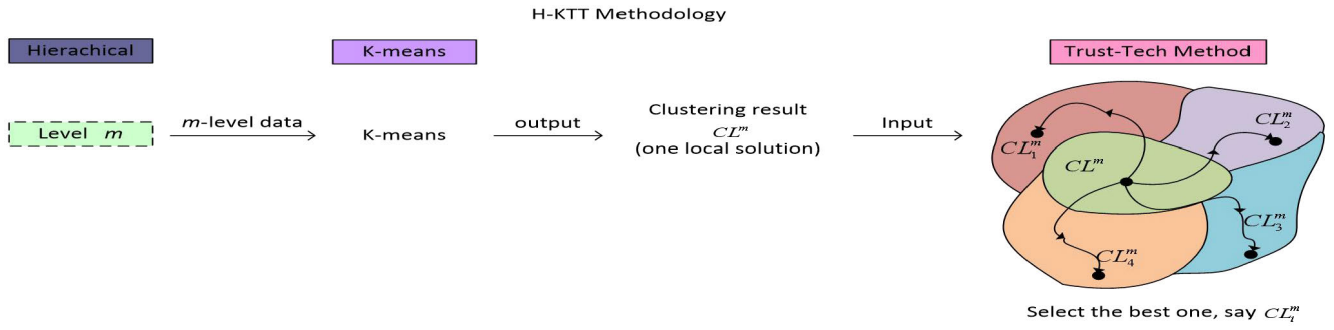


FIGURE 3. The m -level dataset is fed into the K-means method to solve for a fast local clustering solution. This solution is then used as an initial guess for the Trust-Tech method to compute a set of multiple local optimal solutions from which the best local optimal solution is selected and sent to the next-level dataset.

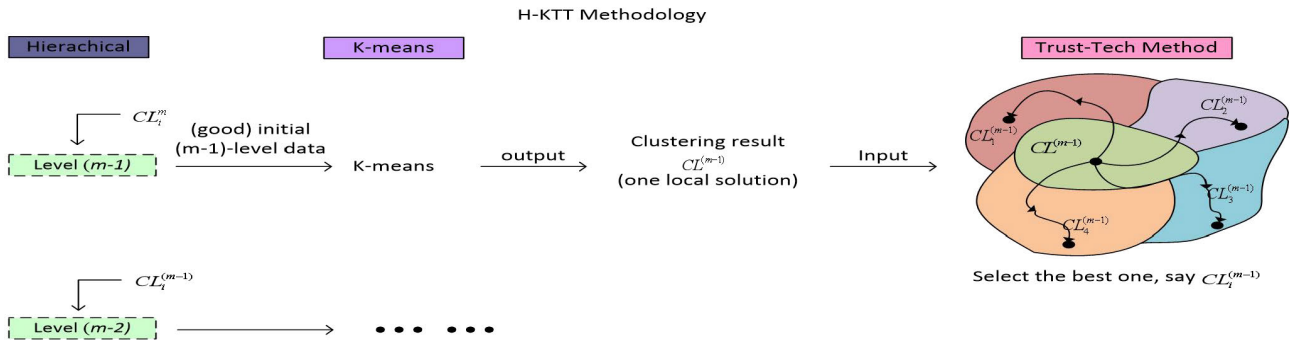


FIGURE 4. The best local optimal solution is mapped into the corresponding $(m-1)$ -level data, and then is fed into the K-means method to solve for a fast local clustering solution. This solution is then used as an initial guess for the Trust-Tech method to compute a set of multiple local optimal solutions from which the best local optimal solution is selected and sent to the next-level dataset. Alternatively, the top two best solutions can be selected (instead of the best one) and mapped to the next level as the initial condition for the K-means method to obtain a fast clustering solution, and this process continues.

Step 1) Select K patterns from the original dataset randomly as the initial centroids.

Step 2) For each pattern, calculate the distance between it and each centroid, and assign it to the nearest cluster.

Step 3) Update each centroid, say the k th centroid, by

$$\omega_k = \frac{1}{n_k} \sum_{i=1}^{n_k} x_i^k.$$

2) CONSTRAINT FUNCTIONS

(1) Clustering information

Since integer variables can only take the value of either 0 or 1, we express the penalty function to be $f_2 = \sum_{i=1}^{n \times k} x_i(1 - x_i)$.

(2) Number of patterns: The function for this cluster is $f_3 = \sum_{i=1}^k (x_{i+n \times k} - \sum_{j=1}^n x_{(i-1)n+j})^2$.

(3) Centroid coordinate:

$$f_4 = \sum_{m=1}^k \sum_{l=1}^d (x_{n \times k + k + (m-1)d+l} - \frac{1}{x_{n \times k + m}} \cdot \sum_{i=1}^n (D_{il} \cdot x_{n(m-1)+i}))^2 \quad (7)$$

where D_{il} represents the coordinate value of the l -th coordinate of the i th vector point in the dataset.

B. CONSTRAINTS

The integer variables corresponding to the clustering information must satisfy a set of constraints to ensure that each point in the dataset belongs to just one centroid.

$$\sum_{i=1}^k x_{n(i-1)+j} = 1, \quad j = 1, 2, \dots, n. \quad (8)$$

V. MULTIPLE CLUSTERING SOLUTIONS BY TRUST-TECH METHODS

The clustering result given by the hierarchical K-means method is used as the initial condition for the Trust-Tech method to compute tier-one local optimal solutions and a higher-tier local optimal solution. While K-means gives a local high-quality solution, Trust-Tech methodology can improve the solution by finding a set of local optimal solutions via the decomposition points of the found local optimal solution, as explained in the following.

To numerically illustrate Trust-Tech methodology, we use the two-dimensional unconstrained Six Hump Camel

problem as an example:

$$\min_{x \in \mathbb{R}^2} f(x) = 4x_1^2 - 2.1x_1^4 + \frac{x_1^6}{3} + x_1x_2 - 4x_2^2 + 4x_2^4 - 5 \leq x_1, \quad x_2 \leq 5 \quad (9)$$

To compute a multiple local optimal solution, we construct the following gradient dynamic system associated with optimization problem (9):

$$\begin{aligned} \dot{x}_1 &= -\frac{\partial f}{\partial x_1} = -8x_1 + 8.4x_1^3 - 2x_1^5 - x_2 \\ \dot{x}_2 &= -\frac{\partial f}{\partial x_2} = -x_1 + 8x_2 - 16x_2^3 \end{aligned} \quad (10)$$

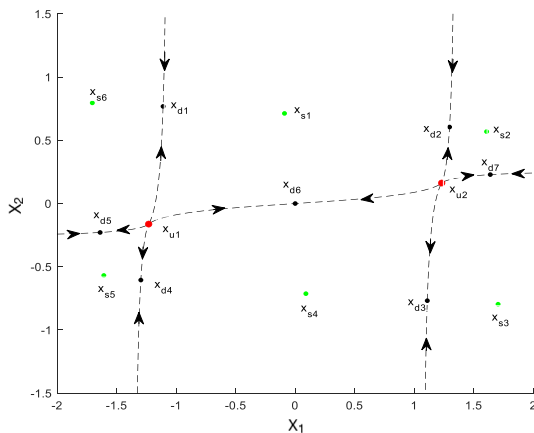


FIGURE 5. There are multiple stable equilibrium points of gradient system (10) and each one corresponds to a local optimal solution of optimization problem (9). The stability region and stability boundary of each stable equilibrium point are highlighted and they are explored to find multiple stable equilibrium points (i.e., multiple local optimal solutions).

The dashed black lines in Fig. 5 are the stability boundaries of dynamic system (10). x_{d1}, \dots, x_{d7} are the UEPs on the boundary and the lines marked with arrows represent the stable manifolds of these UEPs lying on the stability boundary. The stability boundary of (10) is indeed the union of the stable manifolds of the UEPs lying on the stability boundary.

According to Theorem 3, one can escape from a LOS via the type-one dynamic decomposition point lying on its quasi-stability boundary and move to a new LOS via either the unstable manifold of the dynamic decomposition point (DDP) or the unstable eigenvector and another local solver. This motivates the development of a decomposition point-based Trust-Tech method [22], [23]. Note that a type-one EP on the quasi-stability boundary $\partial A_q(x_s)$ is termed a *decomposition point* (with respect to x_s). The decomposition point serves as a bridge linking two SEPs, i.e., one can escape the current stability region via the direction from the SEP to the DDP and enter another stability region to compute the corresponding SEP (i.e., another local minimum) [22], [23],

[24], [25], [26], [27], [28]. However, the Trust-Tech decomposition point method may encounter numerical difficulties in computing exit points.

To resolve these difficulties, we next present a Trust-Tech source-point method [31]. Theorem 4 below establishes the existence of source points on the quasi-stability boundary and the relationship between a source point and the quasi-stability neighboring regions.

Theorem 4 (Existence of Source Points [31]): Assume the gradient dynamical system (2) with a SEP x_s whose stability boundary is non-empty. If the stability region $\partial A_p(x_s)$ is bounded, then at least one source must be on the stability boundary $\partial A_p(x_s)$.

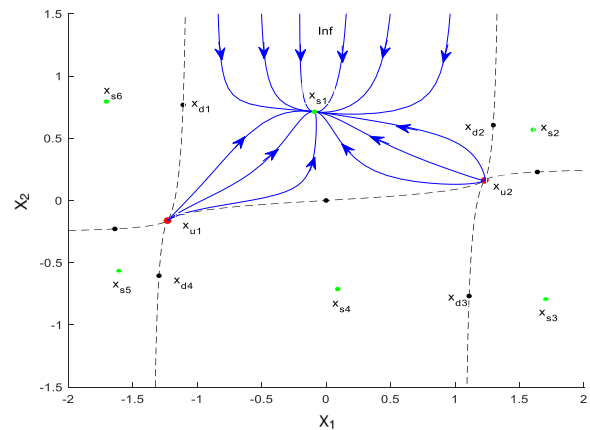


FIGURE 6. There are two source points (i.e., x_{u1}, x_{u2}) that lie on the stability boundary of SEP x_{s1} and other source points that lie in infinity.

To illustrate Theorem 4, we examine example (9) again. We note from Fig. 6 that there are two sources x_{u1} and x_{u2} lying on its stability boundary. Other source points lie in infinity.

Theorem 5 below shows that the unstable manifold of the sources converges to multiple stable equilibrium points (i.e., multiple local optimal solutions).

Theorem 5 (Unstable Manifold of the Source Point [31]): Let the gradient dynamic system (2) contain a SEP x_s whose quasi-stability boundary is non-empty. Let S_i be a source point lying on the quasi-stability boundary $\partial A_p(x_s)$. Then, the unstable manifold of the source intersects the quasi-stability region of every SEP whose quasi-stability boundary intersects the quasi-stability boundary of x_s .

To illustrate Theorem 6, we examine example (9) again and note from Fig. 7 that the blue curves are the unstable manifolds of the source points x_{u1} and the purple curves are the unstable manifolds of the source points x_{u2} . The unstable manifolds of x_{u1} intersect the quasi-stability regions of x_{s4}, x_{s5}, x_{s6} . Moreover, their quasi-stability boundaries indeed intersect the quasi-stability boundaries of x_{s1} at x_{u1} .

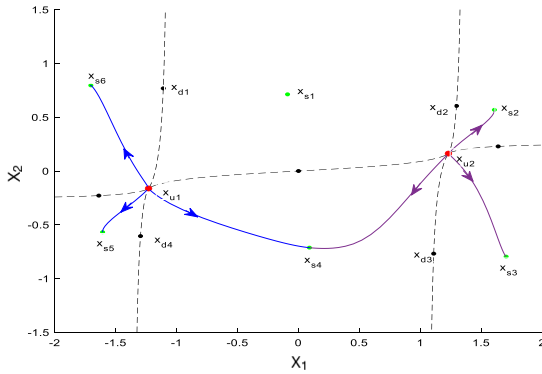


FIGURE 7. The unstable manifolds of x_{u1} intersect the quasi-stability regions of x_{s4} , x_{s5} , x_{s6} ; hence, a total of three local optimal solutions are obtained by following the unstable manifold of the source.

A. THE TRUST-TECH SOURCE POINT METHOD

Stage I: Local search: Given a nonlinear optimization problem (1) and an initial point, construct the corresponding nonlinear dynamical system (2). Apply a numerical integration method to (2) or employ a local search algorithm such as BFGS, trust region, and SQP to compute a local minimum.

Stage II: Exit the stability region. Starting from the SEP found in Stage I, compute a set of source points on its quasi-stability boundary of the LOS of (1).

Stage III: Enter the neighboring stability regions. Follow the unstable manifold of a source to enter neighboring stability regions.

Stage IV: Compute neighboring local minimal solutions. The unstable manifold of each source converges to neighboring SEPs; hence, a set of neighboring LOSs are found.

We note that when a source point cannot be found on the stability boundary, then the search procedure can switch to the decomposition point-based method to find multiple LOSs.

VI. EXPERIMENTAL RESULTS

Clustering techniques such as the K-means algorithm, and its variants have been applied to a variety of problems in power grids such as the seven applications described in the Introduction section. Indeed, electric load clustering is becoming more essential for its great potential in the analytics of consumers' energy consumption patterns and preferences through data mining, given the growing popularity of Automatic Meter Reading (AMR) in the smart grid paradigm. Important potential applications of electric load clustering include the tariff design, anomaly detection, load forecasting, data security, and big data [6].

Since the K-means algorithm suffers from the four issues described above, it will be shown in this section that these issues can be greatly improved or even removed by the proposed H-KTT method on several datasets that are publicly available.

A. STABILITY MEASUREMENT

The stability of a clustering algorithm with respect to small perturbations of the data or the parameters of the algorithm

(e.g., random initialization) is a desirable quality. Since initialization of the K-means algorithm is a random selection from the dataset, the clustering results are more likely to be unstable over different runs. To evaluate the stability of the algorithm, we use the entropy of similarity based on the cluster ensemble (see [11] for details), which means taking the average similarity between each pair of patterns based on all the tests from the cluster ensemble, then calculating the average entropy of the similarities as a measurement of the disagreement value for all the tests. It should be pointed out that stable clustering results give a low disagreement value based on the definition of the entropy function. Conversely, a median average similarity and thus, a high disagreement value, is usually caused by the poor stability of a clustering algorithm.

The experimental dataset we used to evaluate the proposed H-KTT method comes from the popular dataset: the University of California Irvine datasets [30]: the User Knowledge Modeling dataset, the Car Evaluation dataset, and the Synthetic Control Chart time series dataset.

B. CASE I: THE USER KNOWLEDGE MODELING DATASET

The User Knowledge Modeling dataset is a real dataset consisting of knowledge status about the subject of electrical DC machines of 403 undergraduate students at Gazi University in 2009. There are 5 attributes for each donor: STG (the degree of study time for goal object materials), SCG (the degree of the user repetition number for goal object materials), STR (the degree of user study time for related objects with a goal object), LPR (the user exam performance for related objects with a goal object), and PEG (the user exam performance for goal objects). The knowledge level of each student is put into four classes: very low, low, middle, and high. All the attributes were normalized into the range of [0, 10] for clustering proposes.

We first reduce the original dataset by clustering the patterns close to each other with their centroids and obtain a reduced dataset with 20 patterns as the first hierarchical level, then use the original dataset as the second hierarchical level. Level 1, 2 represents different levels of datasets. The 1st level is the original dataset, and each subsequent level is constituted by a smaller dataset of its previous level. And a given dataset can be clustered into several hierarchical levels, as shown in Fig. 1.

We then proceed as follows:

1. Cluster the first-level dataset using the K-means algorithm based on random selection of the initial points.
2. Cluster the second-level dataset using the K-means algorithm with the clustering result of the first level as the initial condition.

We improve the clustering results obtained by applying the H-KTT-means method. For computational efficiency, we apply the Trust-Tech method only at the first hierarchical level and apply the K-means method in successive hierarchical levels. In consideration of the randomness of the K-means algorithm, we set the number of test runs as 10.

All the clustering results obtained by the H-K-means method and by the H-KTT method (the Trust-Tech method is applied at the first level) are summarized in Table 1. In comparison, we also cluster the dataset using the original K-means method.

TABLE 1. Ensemble result for case I.

Index	K-means (e+003)	H-K-means		H-KTT (level 1)	
		Hierarchical level		Hierarchical level	
		1 (e+002)	2 (e+003)	1 (e+002)	2 (e+003)
1	7.2379	6.0269	7.1109	5.6996	6.8817
2	7.0625	6.0079	7.0307	5.6996	6.8817
3	7.1429	5.9248	6.9229	5.6996	6.8817
4	7.1378	5.8449	6.9068	5.6996	6.8817
5	7.1618	5.6996	6.8817	5.6996	6.8817
6	7.1348	6.3280	7.1299	5.6996	6.8817
7	7.2251	5.9248	6.9229	5.6996	6.8817
8	6.9994	5.6996	6.8817	5.6996	6.8817
9	7.1112	6.0079	7.0307	5.6996	6.8817
10	7.1138	6.0269	7.1109	5.6996	6.8817

We observe from the local optimal solutions obtained by the H-K-means that there exists a close relationship between the quality of the clustering results of hierarchical levels (1) and (2) among all the test runs. Moreover, the results obtained by the H-K-means were better (17% to 20% improvement) than the ones given by the K-means method, but the quality of the clustering results is still not the best, and the clustering results are quite unstable (fluctuating between 5.6996 to 6.3280). By contrast, the clustering results given by the H-KTT (which applied the Trust-Tech method at level 1) have the following highlights:

(i) All the clustering solutions obtained by the H-KTT (1), i.e., enhanced by the Trust-Tech method at the first level, are improved to the same high-quality local optimal solution 5.6996, which may be the global optimal solution.

(ii) While the clustering results by the K-means and H-K-means are fluctuating around their averages, the clustering results from the H-KTT are all the same, showing the capability of the proposed H-KTT method in producing consistent outputs.

To analyze the stability of each method, we calculate the disagreement value based on the average similarity between the patterns for the K-means, H-K-means and H-KTT (applied Trust-Tech at level 1), respectively. The disagreement value of each method is summarized in Table 2. We can see that only a small improvement was achieved by the H-K-means while Trust-Tech methodology had reduced the disagreement value to 0, making all the local optimal solutions given by the H-KTT improved to a deterministic one.

C. CASE II: THE CAR EVALUATION DATASET

The Car Evaluation dataset is a simple hierarchical decision model originally developed to demonstrate DEX. This model

TABLE 2. Stability analysis for case I.

Method	K-means	H-K-means	H-KTT (level 1)
Disagreement	0.303364	0.253407	0

evaluates 1728 cars according to the attributes. All the cars are to be clustered into 4 levels: unacceptable, acceptable, good, and very good, based on the given attributes. We numerically normalized all the attributes into the range of [0, 10]. First, we implement a 3-level H-K-means method (i.e., three-level reduced datasets were generated from the original dataset). The Level-3 dataset is the original one with 1728 patterns while the Level-2 dataset has 200 patterns and the Level-1 dataset has 20 patterns. Also, due to the randomness of the K-means method, we set the number of test runs to 20. All the local optimal solutions given by the H-K-means had been improved by Trust-Tech methodology in hierarchical levels (1) and (2) successively (see Table 3 and Fig. 5). The results of the K-means, H-K-means, H-KTT (1) (applied Trust-Tech at level 1) and H-KTT (2) (applied Trust-Tech at levels 1 and 2) are shown in Table 3. The following observations are obtained:

(i) While the clustering results of K-means fluctuates around its average, the clustering results of H-KTT (2) are not only deterministic values but also the best among the four methods.

(ii) The proposed H-KTT (1) and H-KTT (2) outperform the H-K-means and K-means methods in all 20 runs (see Table 3).

(iii) The H-K method outperforms the K-means method in all 20 runs.

Significant improvement has been achieved with application of the Trust-Tech method at the first hierarchical level (1). However, due to the diversity of the obtained clustering solutions, there are still some local optimal solutions (clustering results) whose quality can be further improved. Hence, H-KTT (2) (apply Trust-Tech at levels 1 and 2) is applied for further improvements. Although H-KTT (2) increases computational burden to a certain extent, all of the local optimal solutions by H-KTT (1) are all greatly improved. Moreover, H-KTT (2) is highly stable in the sense that the clustering results of the 20 test runs are the same, making it completely stable in this dataset.

The K-means, H-K-means, H-KTT (1) (applied Trust-Tech at level 1), and H-KTT (2) (applied Trust-Tech at levels 1 and 2) are shown in Table 4, which shows that the application of hierarchical and the Trust-Tech method at level (1) greatly improved the clustering results by 14.10% and 75.02%, respectively. Moreover, the disagreement value of the clustering results is greatly reduced, as all the local optimal solutions are closer to each other, as compared with that of the K-means. The disagreement value was also reduced to zero after we applied Trust-Tech at levels 1 and 2 successively, as all the clustering results obtained by the H-KTT (1) improved based on the optimal one (which can be the global optimal).

TABLE 3. The proposed H-KTT (1) and H-KTT (2) outperform the H-K-means and K-means methods for all 20 runs. The improvements show up in both the quality of solutions and the stability of the solutions. It is interesting to note that both the H-KTT (level 1) and H-KTT (levels 1,2) all give a constant value of 1.1126 for all 20 runs.

Index	K-means	H-K-means			H-KTT (level 1)			H-KTT (levels 1,2)		
	(e+005)	Hierarchical level			Hierarchical level			Hierarchical level		
		1 (e+002)	2 (e+004)	3 (e+005)	1 (e+002)	2 (e+004)	3 (e+005)	1 (e+002)	2 (e+004)	3 (e+005)
1	1.1414	6.7654	1.0791	1.1126	6.7654	1.0791	1.1126	6.7654	1.0791	1.1126
2	1.1517	7.5238	1.1358	1.1514	6.8594	1.0810	1.1134	6.8594	1.0790	1.1126
3	1.1467	7.6310	1.0961	1.1237	6.7606	1.0805	1.1134	6.7606	1.0790	1.1126
4	1.1585	7.4771	1.1348	1.1405	6.7396	1.0789	1.1126	6.7396	1.0789	1.1126
5	1.1559	7.0459	1.0815	1.1203	6.7418	1.0810	1.1134	6.7418	1.0790	1.1126
6	1.1467	7.4266	1.1307	1.1448	6.8617	1.0981	1.1134	6.8617	1.0790	1.1126
7	1.1538	7.4638	1.1642	1.1607	6.7910	1.0789	1.1126	6.7910	1.0789	1.1126
8	1.1819	7.8279	1.1214	1.1292	6.7396	1.0789	1.1126	6.7396	1.0789	1.1126
9	1.1519	7.3414	1.1555	1.1478	6.7396	1.0789	1.1126	6.7396	1.0789	1.1126
10	1.1429	7.5154	1.1016	1.1274	6.7654	1.0791	1.1126	6.7654	1.0791	1.1126
11	1.1406	7.5380	1.1214	1.1384	6.7396	1.0789	1.1126	6.7396	1.0789	1.1126
12	1.1538	7.1898	1.1453	1.1384	6.7654	1.0791	1.1126	6.7654	1.0791	1.1126
13	1.1510	7.6310	1.0961	1.1237	6.7606	1.0805	1.1134	6.7606	1.0790	1.1126
14	1.1585	7.0459	1.0815	1.1203	6.7418	1.0810	1.1134	6.7418	1.0790	1.1126
15	1.1428	7.3414	1.1555	1.1478	6.7396	1.0789	1.1126	6.7396	1.0789	1.1126
16	1.1355	7.5380	1.1214	1.1384	6.7396	1.0789	1.1126	6.7396	1.0789	1.1126
17	1.1527	7.4266	1.1307	1.1448	6.8617	1.0981	1.1134	6.8617	1.0790	1.1126
18	1.1614	6.7654	1.0791	1.1126	6.7654	1.0791	1.1126	6.7654	1.0791	1.1126
19	1.1717	7.4638	1.1642	1.1607	6.7910	1.0789	1.1126	6.7910	1.0789	1.1126
20	1.1675	7.8279	1.1214	1.1292	6.7396	1.0789	1.1126	6.7396	1.0789	1.1126
Ave.	1.1533	7.3893	1.1209	1.1356	6.7704	1.0813	1.1129	6.7704	1.0790	1.1126

TABLE 4. Stability assessment of the clustering results.

Method	K-means	H-K-means	H-KTT (level 1)	H-KTT (level 1,2)
Disagreement	0.450217	0.386746	0.112459	0

The Euclidean distance metric may not be suitable when it comes to the clustering of a set of vector data, such as sequences, time series, etc. At times, it is usually the fluctuation characteristics instead of the vector data amplitudes (or time series) that are of concern during the clustering process. We note that the different amplitudes of two serial data can cause a high Euclidean distance value, although their fluctuation characteristics are similar. In this situation, the cosine distance metric is used such as the following:

$$D(X, Y) = 1 - \cos(X, Y) \tag{11}$$

where $D(X, Y)$ represents the distance between series X and Y . Due to the usage of different distance metrics, the corresponding objective function of the optimization formulation built is modified accordingly.

We tested the time efficiency and memory cost of our proposed method (the number of test runs are set to 20). The results of the K-means, H-K-means, H-KTT (1) (applied Trust-Tech at level 1) and H-KTT (2) (applied Trust-Tech at

levels 1 and 2) are shown in Table 5. The following observations are obtained:

(i) The average time efficiency of the proposed H-KTT (1) and H-KTT (2) in all 20 runs is superior to H-K-means and K-means methods.

(ii) The average memory cost of the proposed H-KTT (1) and H-KTT (2) in all 20 runs is higher than H-K-means and K-means methods. But the increase in memory cost is neglectable.

D. CASE III: SYNTHETIC CONTROL CHART TIME SERIES DATASET

The synthetic control chart time series dataset contains 600 examples of control charts synthetically generated. The length of each is 60 and the amplitude of each lies in the range of 0 to 60. There are 6 different kinds of control charts: normal, cyclic, increasing trend, decreasing trend, upward shift, and downward shift.

Based on the cosine distance metric, we can apply the proposed clustering method to the 600 time series. To compute the metric, the sequences are viewed as vectors in an inner product space, and the cosine similarity is defined as the cosine of the angle between them, that is, the dot product of the vectors divided by the product of their lengths. It follows that the cosine similarity does not depend on the

TABLE 5. Test results of time efficiency and memory cost.

Index	K-means		H-K-means		H-KTT (level 1)		H-KTT (levels 1,2)	
	Time (e+002/ms)	Memory (e+002/MB)	Time (e+002/ms)	Memory (e+002/MB)	Time (e+002/ms)	Memory (e+002/MB)	Time (e+002/ms)	Memory (e+002/MB)
1	1.4384	1.030	1.3836	1.085	1.3370	1.125	1.3237	1.137
2	1.4091	1.121	1.3718	1.211	1.3317	1.242	1.2925	1.265
3	1.4092	1.085	1.3709	1.137	1.3552	1.173	1.3387	1.205
4	1.4991	1.061	1.4460	1.118	1.4303	1.144	1.4105	1.155
5	1.3792	1.096	1.3371	1.147	1.3132	1.180	1.2601	1.220
6	1.3792	1.053	1.3408	1.115	1.2920	1.137	1.2508	1.157
7	1.4991	1.043	1.4513	1.110	1.4064	1.122	1.3566	1.153
8	1.4631	1.097	1.4197	1.192	1.3748	1.213	1.3209	1.244
9	1.4092	1.075	1.3714	1.139	1.3533	1.169	1.3281	1.180
10	1.3792	1.065	1.3314	1.158	1.3072	1.193	1.2509	1.208
11	1.2893	1.097	1.2467	1.167	1.1982	1.178	1.1576	1.193
12	1.4092	1.085	1.3551	1.138	1.2995	1.166	1.2599	1.182
13	1.4692	1.068	1.4327	1.139	1.4023	1.173	1.3899	1.191
14	1.3792	1.076	1.3244	1.132	1.3111	1.143	1.2738	1.161
15	1.4092	1.064	1.3631	1.130	1.3084	1.147	1.2762	1.185
16	1.4692	1.056	1.4142	1.148	1.3994	1.183	1.3745	1.219
17	1.4692	1.084	1.4302	1.137	1.4201	1.169	1.3959	1.203
18	1.4392	1.072	1.3842	1.169	1.3711	1.195	1.3315	1.232
19	1.3495	1.069	1.3007	1.165	1.2518	1.178	1.1922	1.207
20	1.4991	1.110	1.4683	1.171	1.4075	1.193	1.4456	1.220
Ave.	1.4224	1.075	1.3772	1.145	1.34435	1.171	1.3115	1.196

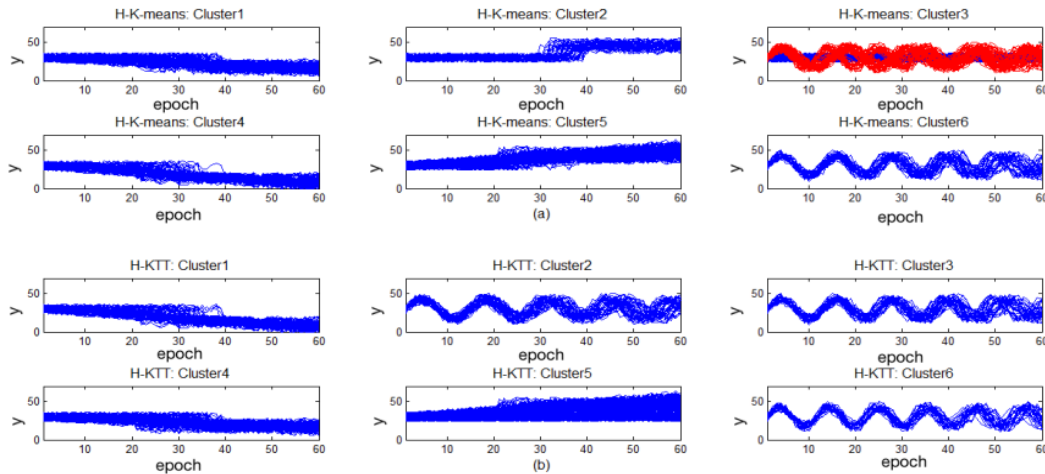


FIGURE 8. Clustering results by H-K-means and H-KTT for case III (refers to distance). The clustering results by the H-KTT are much improved.

magnitudes of the vectors, but only on their angle. The aim of the clustering process is to clearly distinguish all of the time series data as clearly as possible, so we set the number of clusters as 6 and used a simple dataset with 60 patterns as the first hierarchical level, the original 600 time series as the second hierarchical level, and also set the size of the cluster ensemble as 10. Again, we apply the K-means, K-means++ [32] and H-K-means first, then improve the results using Trust-Tech methodology (H-KTT) in the first

hierarchical level. This process was similar to the last two cases, so we omitted the details and summarize the numerical results in Tables 6 and 7.

We note from the ensemble result of the H-K-means that there are still two problems in the test cases: (i) the cyclic series cannot be clearly distinguished from the normal series, (ii) the increasing (or decreasing) trend series cannot be clearly distinguished from the upward (or downward) shift series.

TABLE 6. Ensemble result for case III. The stability of H-KTT (level 1) gives the best result since it gives consistent and smaller results.

Index	K-means	K-means++	H-K-means		H-KTT (level 1)	
			Hierarchical level		Hierarchical level	
			1	7	1	7
1	9.4085	8.8445	0.6760	10.5796	0.4935	8.0847
2	9.3644	9.9485	0.5694	9.2024	0.4935	8.0847
3	9.3644	9.4232	0.5571	8.6636	0.4935	8.0847
4	10.4793	8.8519	0.5740	9.2467	0.4935	8.0847
5	9.5684	10.583	0.5819	9.1588	0.4935	8.0847
6	10.4462	10.5058	0.5980	9.4058	0.4935	8.0847
7	10.0281	9.0946	0.6384	10.0979	0.4935	8.0847
8	9.5180	9.0505	0.5103	8.6372	0.4935	8.0847
9	10.7075	8.6405	0.5238	8.5741	0.4935	8.0847
10	10.6304	9.6407	0.5494	8.7892	0.4935	8.0847

TABLE 7. The disagreement index is greatly improved by the proposed H-KTT method, which give a value of zero.

Method	K-means	K-means++	H-K-means	H-KTT (level 1)
Disagreement	0.303314	0.281265	0.239651	0

Fig. 8(a) shows the clustering effect given by the H-K-means. Here, we took the result of ensemble test (1) as a representative case that had suffered from problems similar to those in the other 9 ensemble cases, although they were not exactly the same. We can find the problems mentioned above where a large number of the cyclic series with different frequencies (marked with red) and the normal series had been mixed in cluster (1), which also caused the numerical result in Table 6 to be of low quality. Moreover, the increasing trend series had been mixed with the upward shift series and the decreasing trend series had been mixed with the downward shift series. By comparison, Fig. 8(b) has shown the clustering effect given by H-KTT, in which all the cyclic series which had caused a confusion in Fig. 8(a) had been clearly separated.

VII. CONCLUSION

An integrated method, termed the H-KTT (hierarchical K-means enhanced by Trust-Tech), has been developed to achieve the following goals:

- (i) It computes multiple high-quality local optimal solutions in clustering or even computes the global optimal solution;
- (ii) It is not sensitive to initial guesses and possesses stability (i.e., consistency) in obtaining high-quality clustering results; in other words, while the K-means gives stochastic clustering results, the proposed H-KTT method usually gives deterministic clustering results.
- (iii) It is effective in solving large-scale clustering problems.

Numerical evaluation of the proposed method on four test datasets favors the accomplishment of the above claimed

advantages. One major disadvantage of the proposed method is that, compared with the K-means method, it is slower as it needs to escape from one local optimal solution to find another local optimal solution in a tier-by-tier manner. However, the pay-off in obtaining high-quality optimal solutions can be tremendous. Future work includes how to improve the computational speed of the proposed method when applied to applications where speed matters in addition to high-quality optimal solutions.

REFERENCES

- [1] C. Si, S. Xu, C. Wan, D. Chen, W. Cui, and J. Zhao, "Electric load clustering in smart grid: Methodologies, applications, and future trends," *J. Modern Power Syst. Clean Energy*, vol. 9, no. 2, pp. 237–252, 2021.
- [2] I. P. Panapakidis, "Clustering based day-ahead and hour-ahead bus load forecasting models," *Int. J. Elect. Power Energy Syst.*, vol. 80, pp. 171–178, Sep. 2016.
- [3] K. Benmouiza and A. Chekneane, "Forecasting hourly global solar radiation using hybrid K-means and nonlinear autoregressive neural network models," *Energy Convers. Manag.*, vol. 75, pp. 561–569, Nov. 2013.
- [4] R. Li, F. Li, and N. D. Smith, "Multi-resolution load profile clustering for smart metering data," *IEEE Trans. Power Syst.*, vol. 31, no. 6, pp. 4473–4482, Nov. 2016.
- [5] S. Lin, F. Li, E. Tian, Y. Fu, and D. Li, "Clustering load profiles for demand response applications," *IEEE Trans. Smart Grid*, vol. 10, no. 2, pp. 1599–1607, Mar. 2019.
- [6] Y.-X. Cai, L.-J. Cai, and Z. Lu, "Anomaly detection of online monitoring data of power equipment based on association rules and clustering algorithm," in *Proc. 2nd Annu. Int. Conf. Electron., Electr. Eng. Inf. Sci. (EEEIS)*, Xi'an, China, 2017, pp. 289–298.
- [7] E. W. S. Angelos, O. R. Saavedra, O. A. C. Cortés, and A. N. De Souza, "Detection and identification of abnormalities in customer consumptions in power distribution systems," *IEEE Trans. Power Del.*, vol. 26, no. 4, pp. 2436–2442, Oct. 2011.
- [8] S. Valero, M. Ortiz, C. Senabre, C. Alvarez, F. Franco, and A. Gabaldon, "Methods for customer and demand response policies selection in new electricity markets," *IET Gener., Transmiss. Distrib.*, vol. 1, no. 1, pp. 104–110, 2007.
- [9] R. F. Chang and C. N. Lu, "Load profiling and its applications in power market," in *Proc. IEEE Power Eng. Soc. Gen. Meeting*, Toronto, ON, Canada, Jul. 2003, pp. 974–978.
- [10] J.-H. Shin, B.-J. Yi, Y.-I. Kim, H.-G. Lee, and K. H. Ryu, "Spatiotemporal load-analysis model for electric power distribution facilities using consumer meter-reading data," *IEEE Trans. Power Del.*, vol. 26, no. 2, pp. 736–743, Apr. 2011.
- [11] W. Bi, M. Cai, M. Liu, and G. Li, "A big data clustering algorithm for mitigating the risk of customer churn," *IEEE Trans. Ind. Informat.*, vol. 12, no. 3, pp. 1270–1281, Jun. 2016.
- [12] J.-S. Wang and J.-C. Chiang, "A cluster validity measure with outlier detection for support vector clustering," *IEEE Trans. Syst., Man, Cybern., B*, vol. 38, no. 1, pp. 78–89, Feb. 2008.
- [13] L. I. Kuncheva and D. P. Vetrov, "Evaluation of stability of K-means cluster ensembles with respect to random initialization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 11, pp. 1798–1808, Nov. 2006.
- [14] D. Charalampidis, "A modified K-means algorithm for circular invariant clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 12, pp. 1856–1865, Dec. 2005.
- [15] J. Z. Huang, M. K. Ng, H. Rong, and Z. Li, "Automated variable weighting in K-means type clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 5, pp. 657–668, May 2005.
- [16] M.-C. Su and C.-H. Chou, "A modified version of the K-means algorithm with a distance based on cluster symmetry," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 6, pp. 674–680, Jun. 2001.
- [17] Z. Huang, "Extensions to the K-means algorithms for clustering large data sets with categorical values," *Data Mining Knowl. Discovery*, vol. 2, no. 3, pp. 283–304, 1998.
- [18] I. P. Panapakidis, T. A. Papadopoulos, G. C. Christoforidis, and G. K. Papagiannis, "Pattern recognition algorithms for electricity load curve analysis of buildings," *Energy Buildings*, vol. 73, pp. 137–145, Apr. 2014.

- [19] D. Arthur and S. Vassilvitskii, "K-Means++: The advantages of careful seeding," in *Proc. 18th Annu. ACM-SIAM Symp. Discrete Algorithms*, New Orleans, LA, USA, 2013, pp. 1027–1035.
- [20] Y.-C.-F. Wang and D. Casasent, "Hierarchical K-means clustering using new support vector machines for multi-class classification," in *Proc. IEEE Int. Joint Conf. Neural Netw.*, Vancouver, BC, Canada, Jul. 2006, pp. 3457–3464.
- [21] B. Chen, R. Harrison, Y. Pan, and P. C. Tai, "Novel hybrid hierarchical-K-means clustering method (H-K-means) for microarray analysis," in *Proc. IEEE Comput. Syst. Bioinf. Conf.-Workshops (CSBW)*, Aug. 2005, pp. 105–108.
- [22] H.-D. Chiang and C.-C. Chu, "A systematic search method for obtaining multiple local optimal solutions of nonlinear programming problems," *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol. 43, no. 2, pp. 99–109, Feb. 1996.
- [23] J. Lee and H.-D. Chiang, "A dynamical trajectory-based methodology for systematically computing multiple optimal solutions of general nonlinear programming problems," *IEEE Trans. Autom. Control*, vol. 49, no. 6, pp. 888–899, Jun. 2004.
- [24] H. D. Chiang and L. F. C. Alberto, *Stability Regions of Nonlinear Dynamical Systems: Theory, Estimation, and Applications*. Cambridge, U.K.: Cambridge Univ. Press, 2015.
- [25] B. Wang and H.-D. Chiang, "ELITE: Ensemble of optimal input-pruned neural networks using TRUST-TECH," *IEEE Trans. Neural Netw.*, vol. 22, no. 1, pp. 96–109, Jan. 2011.
- [26] C. K. Reddy, H. D. Chiang, and B. Rajaratnam, "TRUST-TECH-based expectation maximization for learning finite mixture models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 7, pp. 1146–1157, Jul. 2008.
- [27] Y.-F. Zhang and H.-D. Chiang, "A novel consensus-based particle swarm optimization-assisted trust-tech methodology for large-scale global optimization," *IEEE Trans. Cybern.*, vol. 47, no. 9, pp. 2717–2729, Sep. 2017.
- [28] H. D. T. C. Wang, "A novel TRUST-TECH guided branch-and-bound method for nonlinear integer programming," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 45, no. 10, pp. 1361–1372, Oct. 2015.
- [29] T.-S. Xu, H.-D. Chiang, G.-Y. Liu, and C.-W. Tan, "Hierarchical K-means method for clustering large-scale advanced metering infrastructure data," *IEEE Trans. Power Del.*, vol. 32, no. 2, pp. 609–616, Apr. 2017.
- [30] D. Dua and C. Graff. (2019). *UCI Machine Learning Repository*. [Online]. Available: <http://archive.ics.uci.edu/ml>
- [31] L. Zeng, H.-D. Chiang, D. Liang, M. Xia, and N. Dong, "Trust-tech source-point method for systematically computing multiple local optimal solutions: Theory and method," *IEEE Trans. Cybern.*, vol. 52, no. 11, pp. 11686–11697, Nov. 2022.
- [32] D. Arthur and S. Vassilvitskii, "K-means++: The advantages of careful seeding," in *Proc. Symp. Discrete Algorithms*, 2007, pp. 1027–1035.

HSIAO-DONG CHIANG (Life Fellow, IEEE) is currently a Professor with the School of Electrical and Computer Engineering, Cornell University, Ithaca, NY, USA. He is also the Founder of Bigwood Systems, Inc., and Global Optimal Technology, Inc., Ithaca, NY, USA. He and his research team have published more than 400 papers in refereed journals and conference proceedings. His publications earned a H-index of 58. He holds 18 U.S. and overseas patents and several consultant positions. He is the author of two books *Direct Methods for Stability Analysis of Electric Power Systems: Theoretical Foundations, BCU Methodologies, and Applications* (Hoboken, NJ, USA: Wiley, 2011) and with Luis F. C. Alberto *Stability Regions of Nonlinear Dynamical Systems: Theory, Estimation, and Applications* (Cambridge, U.K.: Cambridge University Press, 2015). His research interests include nonlinear system theory, nonlinear computation and optimization, and their practical applications to electric power systems and energy management systems.

He has served as an associate editor for several IEEE TRANSACTIONS and IEEE journals and a Board Member for IEE Japan.

TIAN-SHI XU received the B.Sc. and M.Sc. degrees in electrical engineering and automation from Tianjin University, Tianjin, China, in 2013 and 2015, respectively. His areas of research interests include power systems and distribution system analysis, load management, computational techniques, Trust-Tech methodology, and artificial-intelligence applications.

XIAN-LONG LV received the B.S. and M.S. degrees in electrical engineering from the University of Jinan, Jinan, China, in 2018 and 2015, respectively. He is currently pursuing the Ph.D. degree in electrical engineering with the School of Electrical Engineering and Automation, Tianjin University, Tianjin, China.

His research interests include nonlinear systems theory and global optimization methods and their applications to machine learning, power systems, and computer vision.

NA DONG received the Ph.D. degree in control theory and control application from Nankai University, China, in 2011.

She is currently an Associate Professor with the School of Electrical and Information Engineering, Tianjin University, China. Her current research interests include intelligent control algorithms, heuristic optimization algorithm, neural networks, data-driven control, deep learning, and image processing.