Method for Determining the Optimal Number of Clusters Based on Agglomerative Hierarchical Clustering

Shibing Zhou, Zhenyuan Xu, and Fei Liu

***Abstract*— It is crucial to determine the optimal number of clusters for the clustering quality in cluster analysis. From the standpoint of sample geometry, two concepts, i.e., the sample clustering dispersion degree and the sample clustering synthesis degree, are defined, and a new clustering validity index is designed. Moreover, a method for determining the optimal number of clusters based on an agglomerative hierarchical clustering (AHC) algorithm is proposed. The new index and the method can evaluate the clustering results produced by the AHC and determine the optimal number of clusters for multiple types of datasets, such as linear, manifold, annular, and convex structures. Theoretical research and experimental results indicate the validity and good performance of the proposed index and the method.**

***Index Terms*— Cluster analysis, clustering validity index, hierarchical clustering, number of clusters.**

I. Introduction

CLUSTER analysis is an important research topic in the fields of pattern recognition and machine learning. Clustering is to divide samples into many clusters according to some similarity criteria so that the samples in the same cluster are as similar as possible and samples in different clusters are as distinct as possible. Researchers have come up with many clustering algorithms, which have been widely applied. They can be broadly classified into two groups: partitional and hierarchical. Partitional algorithms process input data and create a partition that groups the data into clusters. In contrast, hierarchical algorithms build a nested partition set called a cluster hierarchy. In general, a hierarchical clustering algorithm partitions a data set into various clusters via an agglomerative or a divisive approach based on a dendrogram [1]. Agglomerative clustering begins from the singleton clusters and obtains a hierarchy by successively merging clusters, whereas divisive clustering begins with a single cluster containing all the points and proceeds by iteratively splitting the clusters [2]. In hierarchical algorithms, agglomerative hierarchical clustering (AHC) has become one of the primary clustering algorithms due to less time complexity and better computational stability. Hierarchical algorithms can offer more clustering results than partitional algorithms, but how to obtain the best partition from numerous clustering results is a subject worthy of study. Because every layer of a cluster hierarchy is determined by a threshold value, which corresponds to the number of clusters, the optimal partitioning problem can be characterized as a question of determining the optimal clustering number. Some research has already been conducted to determine the optimal number of clusters in hierarchical clustering [2]–[9]. Gurrutxaga *et al.* [2] proposed the search over the extended partition set method and a new clustering validity index called context-independent optimality and partiality to find the best partition in hierarchical clustering. Nasibov and Kandemir-Cavas [3] integrated the ordered weighted averaging (OWA) operator with hierarchical clustering to determine the distance between clusters, then calculated the root-mean-square standard deviation and R-squared validity indexes to evaluate the results of hierarchical clustering algorithms. However, they did not do enough research regarding tuning the optimal weights of the OWA operator to obtain the best clustering results. Chen *et al.* [8] proposed a hierarchical method, which first obtained the clustering feature via scanning the data set and agglomeratively generated hierarchical partitions of the data set, then incrementally constructed a curve of the clustering quality for diverse partitions, and finally used the partitions corresponding to the extremum of the curve to estimate the optimal number of clusters. They stated that the proposed method could be used to measure nonconvex structure data. However, they did not describe the detailed distribution of experimental datasets. Hu *et al.* [9] proposed a new clustering validity index to determine the optimal number of clusters in hierarchical clustering by defining the compactness and separability of clustering results. Because the paper did not specify the similarity measure method for constructing a similarity matrix, we cannot verify the experimental results.

Based on the AHC algorithm, we propose a new clustering validity index, which could effectively evaluate the clustering results of multiple types of data sets, such as linear, manifold, annular, and convex structures. Applying the new index, a method for determining the optimal number of clusters is proposed. Theoretical research and experimental results

Manuscript received November 8, 2015; revised February 26, 2016 and September 6, 2016, accepted September 6, 2016. Date of publication October 5, 2016; date of current version November 15, 2017. This work was supported in part by the Fundamental Research Funds for the Central Universities under Grant JUSRP11235, Grant JUSRP51510, and Grant JUSRP51635B, in part by the National Natural Science Foundation of China under Grant 61134007, Grant 61170121, and Grant 61300152, and in part by the Program for Innovative Research Team of Jiangnan University.

S. Zhou is with the Department of Computer Science and Technology, Jiangnan University, Wuxi 214122, China (e-mail: worldguard@sina.com).

Z. Xu is with the School of Science, Jiangnan University, Wuxi 214122, China (e-mail: xuzhenyuan1946@hotmail.com).

F. Liu is with the Institute of Automation, Jiangnan University, Wuxi 214122, China (e-mail: fliu@jiangnan.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNNLS.2016.2608001

Algorithm 1 AHC

Step 1: Initially, each data point forms a cluster by itself, i.e., $G_i = \{x_i\}(i = 1, 2, \ldots, n)$.
Step 2: Find the distances between each pair of objects of the clusters and construct the distance matrix $D = (D_{ij})_{cc}$, where c is the number of clusters.
Step 3: Merge the clusters that are closer to each other (suppose that they are G_p and G_q) into a new cluster G_r with their elements as $G_p \cup G_q$, and let $c = c - 1$.
Step 4: Check the number of clusters. If the class number c is greater than the desired number of clusters, go to Step 2; otherwise, proceed to Step 5.
Step 5: Output the clustering results.

indicate the validity and good performance of the proposed method.

The remainder of this paper is organized as follows. Section II presents a brief overview of the AHC algorithm. A new validity index is presented in Section III. A novel algorithm for determining the optimal number of clusters is described in Section IV. Experimental results for synthetic data sets and real datasets are presented in Section V. A discussion on extending the proposed method to other clustering algorithms is included in Section VI. Finally, the conclusion is provided in Section VII.

II. Agglomerative Hierarchical Clustering

In accordance with bottom–up and top–down methods, hierarchical clustering algorithms can be divided into agglomerative and divisive algorithms. This paper mainly focuses on the AHC algorithm. Agglomerative procedures start with n singleton clusters and then form a sequence by successively merging clusters. Considering that the data set is $\{x_1, x_2, \ldots, x_n\}$ and G_i is the ith cluster in the kth merging process, the AHC algorithm consists by Algorithm 1 [10].

The AHC algorithm has three main distance measures: single linkage, complete linkage, and average linkage [11]. The AHC algorithm is also called the nearest neighbor cluster algorithm when the single linkage measure is used for the distance between clusters. Supposing that we have n samples and seek to form c clusters using a single linkage, the computational complexity is as follows [12], [13].

1) Once and for all, calculate $n(n-1)$ interpoint distances; the time complexity is $O(n^2)$.
2) Place the results in an interpoint distance table; the space complexity is $O(n^2)$.
3) Step through the complete list to find the minimum distance pair for the first merging and keep the index of the smallest distance; thus, for the first agglomerative step, the time complexity is $O(n(n-1)) = O(n^2)$.
4) Step through the unused distances in the list and find the smallest one in different clusters for an arbitrary agglomerative step; the time complexity is $O(n^2)$.

Usually, c is far less than $n(c \ll n)$. Therefore, with single linkage, the space complexity of the AHC algorithm is $O(n^2)$ and the full time complexity of the AHC algorithm is $O((n-c)n^2) = O(n^3)$.

In most cases, the AHC algorithm with single linkage is relatively stable and effective for linear, manifold, annular, and convex data. Thus, we use the single linkage measure for the cluster analysis of datasets in our experiments.

III. CSP: A New Clustering Validity Index

The evaluation of clustering results is called clustering validity analysis. In general, to show the internal structure of a data set, a good clustering partition should make the samples in the same cluster as similar as possible and samples in different clusters as distinct as possible. From the perspective of the distance measure, the optimal clustering partition is to minimize the intracluster distance and maximize the intercluster distance simultaneously. Among the existing validity indexes, there are many indexes that can analyze the clustering results and determine the optimal number of clusters. Good performance indexes mainly include the Calinski–Harabasz (CH) index [14], Davies–Bouldin (DB) index [15], silhouette (Sil) index [16], Krzanowski–Lai (KL) index [17], and Weighted inter–intra (Wint) index [18], [19]. In these indexes, the CH index, DB index, and KL index calculate the means of the samples as the clustering centers, which may be invalid for nonconvex structure data sets. The Sil index presents the clustering validity of a single sample, which may neglect the common characteristic of all samples in each cluster. Moreover, there are some drawbacks in the definition of intersimilarity and intrasimilarity for the Wint index, which may be invalid for nonconvex data sets. Due to the defects of these indexes, it is difficult to find the optimal number of clusters for nonconvex structure data sets correctly. To solve the problem, we design a new clustering validity index called the compact-separate proportion (CSP) index. The CSP index can evaluate the clustering results produced by the AHC algorithm and determine the optimal number of clusters for linear, manifold, annular, and convex data sets.

A. Definitions of CSP Index and Related Concepts

Supposing that D_{nn} is the distance matrix of data set X, by using the AHC algorithm, we obtain the dendrogram of data set X, which is $H = \{H_1, H_2, \ldots, H_n\}$. In the dendrogram, any layer, such as H_k, includes c clusters and each cluster contains n_i samples. We use intracluster compactness to measure the similarity of the samples in the same cluster and use intercluster separation to measure the similarity of the samples in different clusters.

Definition 1: Assume that H_k is a layer of the dendrogram produced by the AHC algorithm and there are c clusters $\{G_1, G_2, \ldots, G_c\}$ in H_k, each cluster containing n_i samples, $i = 1, 2, \ldots, c$. We take the average weight of the minimum spanning tree for all samples in the ith cluster as the intracluster compactness $cd(i)$, which is defined as follows:

$$cd(i) = \frac{W(G_i)}{n_i - 1} \tag{1}$$

where $W(G_i)$ is the weight of the minimum spanning tree for all samples in the ith cluster.

Definition 2: Assume that H_k is a layer of the dendrogram produced by the AHC algorithm, and there are c clusters $\{G_1, G_2, \ldots, G_c\}$ in H_k, each cluster containing n_i samples, $i = 1, 2, \ldots, c$. We take the minimum value of minimum distances between the samples in cluster i and the samples in other clusters as the intercluster separation $sd(i)$, which is defined as follows:

$$sd(i) = \min_{1 \le j \le c, j \ne i} \{\min\{\text{dist}(x_i, x_j) | x_i \in G_i, x_j \in G_j\}\} \quad (2)$$

where $\text{dist}(x_i, x_j)$ is the Euclidean distance of intercluster samples x_i and x_j, i.e., $\text{dist}(x_i, x_j) = \|x_i - x_j\|$; $\|\cdot\|$ represents Euclidean distance.

Definition 3: Assume that H_k is a layer of the dendrogram produced by the AHC algorithm, and there are c clusters $\{G_1, G_2, \ldots, G_c\}$ in H_k, each cluster containing n_i samples, $i = 1, 2, \ldots, c$. We take the difference between the intercluster separation and intracluster compactness of cluster i as the clustering dispersion degree $\text{sscd}(i)$, which is defined as follows:

$$\begin{aligned} \text{sscd}(i) &= sd(i) - cd(i) \\ &= \min_{1 \le j \le c, j \ne i} \{\min\{\text{dist}(x_i, x_j) | x_i \in G_i, x_j \in G_j\}\} \\ &\quad - \frac{W(G_i)}{n_i - 1}. \end{aligned} \quad (3)$$

Definition 4: Assume that H_k is a layer of the dendrogram produced by the AHC algorithm, and there are c clusters $\{G_1, G_2, \ldots, G_c\}$ in H_k, each cluster containing n_i samples, $i = 1, 2, \ldots, c$. We take the sum of the intercluster separation and the intracluster compactness of cluster i as the clustering synthesis degree $\text{sacd}(i)$, which is defined as follows:

$$\begin{aligned} \text{sacd}(i) &= sd(i) + cd(i) \\ &= \min_{1 \le j \le c, j \ne i} \{\min\{\text{dist}(x_i, x_j) | x_i \in G_i, x_j \in G_j\}\} \\ &\quad + \frac{W(G_i)}{n_i - 1}. \end{aligned} \quad (4)$$

Definition 5: Assume that H_k is a layer of the dendrogram produced by the AHC algorithm, and there are c clusters $\{G_1, G_2, \ldots, G_c\}$ in H_k, each cluster containing n_i samples, $i = 1, 2, \ldots, c$. We take the ratio of the clustering dispersion degree and the clustering synthesis degree for cluster i as the CSP index $\text{CSP}(i)$, which is defined as follows:

$$\begin{aligned} &\text{CSP}(i) \\ &= \frac{\text{sscd}(i)}{\text{sacd}(i)} = \frac{sd(i) - cd(i)}{sd(i) + cd(i)} \\ &= \frac{\min_{1 \le j \le c, j \ne i} \{\min\{\text{dist}(x_i, x_j) | x_i \in G_i, x_j \in G_j\}\} - \frac{W(G_i)}{n_i - 1}}{\min_{1 \le j \le c, j \ne i} \{\min\{\text{dist}(x_i, x_j) | x_i \in G_i, x_j \in G_j\}\} + \frac{W(G_i)}{n_i - 1}}. \end{aligned} \quad (5)$$

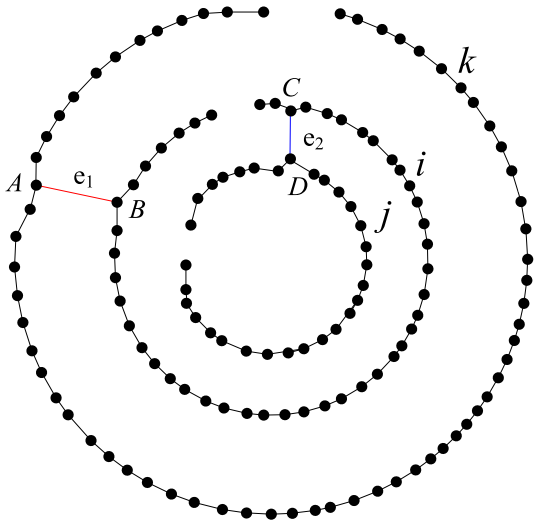

Fig. 1. Distribution diagram of clustering structure.

B. Analysis of CSP Index

To reflect the intracluster compactness and intercluster separability of data, we propose the CSP index. Based on the geometry structure of samples, the CSP index takes the samples in one cluster as the research object and analyzes the validity of clustering results. To explain the meaning of the CSP index and related concepts conveniently, we illustrate the CSP index by using the distribution diagram of Fig. 1.

In Fig. 1, all samples of the data set are distributed in three approximate circles and can be clustered into three clusters, denoted by j, i, and k. We can use the AHC algorithm to cluster the data set. We treat the data points as the nodes in Fig. 1 and use edges to form a path between the nodes in every cluster. When the single linkage metric is used to measure the distance between clusters, the nearest neighbor nodes determine the two nearest clusters. The merging of cluster i and cluster j corresponds to adding an edge between the nearest pair of nodes in cluster i and cluster j. Because edges linking clusters always go between distinct clusters, the resulting graph never has any closed loops or circuits. If it is allowed to continue until all clusters are linked, the result is a minimum spanning tree [13]. To better explain the clustering structure of Fig. 1, using the terminology of graph theory, we present the following definition and theorems, where $G = (V, E)$ denotes the graph with vertex set V and edge set E, $|V|$ is the number of vertexes and $|E|$ is the number of edges.

Definition 6: If $T = (V_T, E_T)$ is a subtree of graph $G = (V, E)$, T is the c-near neighbor minimum spanning tree if and only if it satisfies the following properties, where c is the number of branches.

1) *Minimality:* T is the minimum spanning tree of induced subgraph $G[V_T]$.
2) *Near neighbor:* $\text{Max}\{W(e) | e \in E_T\} \le W'(c)$, where $W(e)$ is the weight value of an edge, and $W'(c)$ is the cth largest weight value in the minimum spanning tree of graph G.

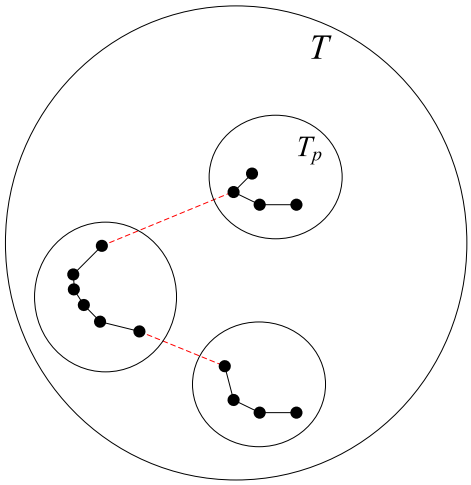

Fig. 2. Diagram of connected branches.

Theorem 1: Assume $T = (V_T, E_T)$ is the minimum spanning tree of graph $G = (V, E)$. For any $1 \leq c \leq |V|$, we remove the edges whose weight values are greater than $W'(c)$ from T and produce c connected branches, which are marked as $T_k = (V_k, E_k)$ $(k = 1, 2, \ldots, c)$. The resulting branch T_k is the c-near neighbor minimum spanning tree of graph G.

Proof: Because weights of edges removed from T are all greater than $W'(c)$, $\text{Max}\{W(e)|e \in E_k\} \leq W'(c)$, i.e., T_k satisfies the near neighbor condition of Definition 6. We need to prove the minimality, which could be proved by contradiction. Assuming that one of the c connected branches is not the c-near neighbor minimum spanning tree of graph G, we take it as $T_p = (V_p, E_p)$, which is not a minimum spanning tree. The induced subgraph $G[V_p]$ must have the minimum spanning tree; we take it as $T_q = (V_p, E_q)$, $W(T_q) < W(T_p)$. To understand the notations easily, we illustrate the proving process by using the diagram of Fig. 2. When c connected branches $T_k = (V_k, E_k)$ $(k = 1, 2, \ldots, p, \ldots, c)$ reconnect the $c - 1$ removed edges whose weight values are greater than $W'(c)$, the produced spanning tree T is a minimum spanning tree of graph G. From Fig. 2, we can see that the red broken lines connect every connected branch and produce the minimum spanning tree T. We set $W(T) = W(T_1) + W(T_2)+, \ldots, +W(T_p)+, \ldots, +W(T_c) + W'$, where W' represents the total weight values of $c-1$ removed edges. However, $W(T_1) + W(T_2)+, \ldots, +W(T_p)+, \ldots, +W(T_c) + W' > W(T_1) + W(T_2)+, \ldots, +W(T_q)+, \ldots, +W(T_c) + W'$ shows that T is not a minimum spanning tree of graph G. This is in contradiction with the assumption that T is the minimum spanning tree of graph G. Thus, the assumption that T_p is not the c-near neighbor minimum spanning tree of graph G does not hold. Thus, we prove that each of the c connected branches is the c-near neighbor minimum spanning tree of graph G.

Theorem 1 shows that after obtaining the minimum spanning tree T of a connected weighted graph for the data set, by removing $c - 1$ largest edges from T, we obtain a group of c-near neighbor minimum spanning trees of graph G, i.e., we obtain c clusters. $c - 1$ largest edges are $c - 1$ minimum distances between two clusters. In the clustering structure of Fig. 1, we show the minimum spanning tree of all samples in the data set. If samples are clustered into three clusters, only edge AB (the minimum distance between cluster i and cluster k) and edge CD (the minimum distance between cluster i and cluster j) need to be removed from the minimum spanning tree. Cluster j is the nearest cluster to cluster i. If cluster j meets the requirement of intercluster separability, other clusters will also meet the requirement. Moreover, if samples of cluster i could not be clustered into cluster i, cluster j will be their best choice. Therefore, the research of cluster i and its nearest neighbor cluster j has important significance.

The standard of clustering validity is to make the clustering results achieve intracluster compactness and intercluster separability. From the perspective of intracluster compactness, based on the distance measure, we expect that the intracluster distance of cluster i is as small as possible. Neither minimum distance nor maximum distance of samples in the same cluster is representative. From the above research, we know that the samples in cluster i could make up a c-near neighbor minimum spanning tree. It is suitable to use the average weight value of the minimum spanning tree for samples in cluster i to measure the intracluster compactness of cluster i. On the other hand, from the perspective of intercluster separability, we expect the intercluster distance between cluster i and its nearest neighbor cluster j to be as large as possible. It is not suitable to take the average distance between samples in cluster i and cluster j as the intercluster separability of two clusters. In general, minimum intercluster distance plays a great role in clustering partition. Therefore, we use minimum intercluster distance e_2 between cluster i and cluster j to measure the intercluster separability of cluster i. To take both of the factors into account, we use a linear combination of them to make the index function an increasing function. We use the clustering dispersion degree $\text{sscd}(i)$, which is equal to $sd(i) + (-cd(i))$, to evaluate the clustering results. It is obvious that the larger $\text{sscd}(i)$ is, the better the clustering effect could be. To make CSP index analyze the clustering validity of all clusters and make it invariable to the dimension, we put forward the concept of the clustering synthesis degree. We divide the clustering dispersion degree by the clustering synthesis degree to make the CSP index dimensionless. It is for such a normalizing factor that we propose the clustering synthesis degree. Moreover, we use the clustering synthesis degree to balance the following $\text{avgCSP}(k)$ function, which affects the determination of the optimal number of clusters directly. The clustering synthesis degree can prevent the CSP index from being too large or too small and help avoid extreme cases that affect the $\text{avgCSP}(k)$ function excessively.

Theorem 2: If H_k is a layer of the dendrogram produced by the AHC algorithm and there are c clusters $\{G_1, G_2, \ldots, G_c\}$ in H_k, each cluster containing n_i samples, $i = 1, 2, \ldots, c$, the intercluster separability of cluster i is greater than or equal to intracluster compactness, i.e., $sd(i) \geq cd(i)$.

Proof: Because the intercluster separability of cluster i is the minimum intercluster distance between cluster i and its nearest neighbor cluster j, which is one of the $c - 1$ largest edges

during the process of producing a group of c-near neighbor minimum spanning trees, if $T' = (V_{T'}, E_{T'})$ is the minimum spanning tree of a connected and weighted graph for samples in cluster i and the weight values are the distances of samples, it is obvious that $sd(i) \geq \text{Max}\{W(e)|e \in E_{T'}\}$, where $W(e)$ is the weight value of an edge. Because the average weight value $cd(i) \leq \text{Max}\{W(e)|e \in E_{T'}\}$, $sd(i) \geq cd(i)$.

Deduction 1: The CSP index value is in the range of [0, 1].

Proof: According to Theorem 2, $sd(i) - cd(i) \geq 0$; applying Definition 5, we draw the conclusion that $\text{CSP}(i) \in [0, 1]$. When $sd(i) = cd(i)$, $\text{CSP}(i) = 0$. When there is only one sample in cluster i, i.e., $cd(i) = 0$, $\text{CSP}(i) = 1$.

The above analysis shows that the CSP index has certain rationality.

C. CSP Index and Determination of the Optimal Number of Clusters

The CSP index shows the clustering validity of the samples in one cluster. The higher the value of CSP, the better the clustering result of the samples in one cluster. We analyze the clustering effects of a data set by calculating the average CSP index value of all clusters in the data set. The higher the average value is, the better the clustering result would be. The clustering number that corresponds to the maximum average value is the optimal number of clusters. Furthermore, we obtain the following formulas, where n is the number of data points, $\text{avgCSP}(k)$ is the average CSP value in the case where data points are clustered into k clusters, and k_{opt} is the optimal number of clusters:

$$\text{avgCSP}(k) = \frac{1}{k}\sum_{i=1}^{k}\text{CSP}(i) \tag{6}$$

$$k_{\text{opt}} = \arg\max_{2\leq k<n}\{\text{avgCSP}(k)\}. \tag{7}$$

IV. Algorithm for Determining the Optimal Number of Clusters

Based on the AHC algorithm with single linkage and CSP clustering validity index, the optimal number of clusters determination (ONCD) algorithm is summarized in Algorithm 2.

The AHC algorithm with single linkage includes two phases. The first phase is to create a hierarchical cluster tree; the second phase is to construct clusters from the hierarchical cluster tree. The first phase is also the process of producing the minimum spanning tree. From the hierarchical cluster tree, we can find enough information regarding all the edges that form the minimum spanning tree in sequence. During the second phase, the weight $W(G_i)$ of all samples' minimum spanning tree in the ith cluster and the intercluster separation $sd(i)$ can easily be calculated. To understand the notion easily, we illustrate the calculation process by using the diagram of Fig. 3. From the hierarchical cluster tree, we can construct the ith cluster, in which there are data points A, B, C, and D, and we can obtain the distance of edges AB, BC, and CD. It is easy to calculate the weight $W(G_i)$, whose value is $e_1 + e_2 + e_3$. Thus, the intracluster compactness $cd(i)$ can be calculated.

Algorithm 2 ONCD

Step 1: Select the search range $[k_{\min}, k_{\max}]$ for clustering numbers.
Step 2: Use the single linkage measure to create a hierarchical cluster tree.
Step 3: For $k = k_{\min}$ to $k_{\max}$
  Step a: Construct k clusters from the hierarchical cluster tree.
  Step b: Use (1) to calculate the intra-cluster compactness of a single cluster based on the hierarchical cluster tree.
  Step c: Use (2) to calculate the inter-cluster separation of a single cluster based on the hierarchical cluster tree.
  Step d: Use (5) to calculate the CSP index value of a single cluster.
  Step e: Use (6) to calculate the average value of the CSP index.
Step 4: Use (7) to calculate the optimal number of clusters.
Step 5: Output the optimal number of clusters, validity index values and clustering results.

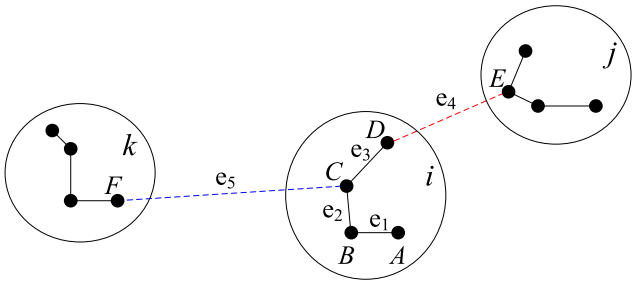

Fig. 3. Constructing clusters from the hierarchical cluster tree.

From the hierarchical cluster tree, we can easily obtain the first point that will connect the ith cluster. From Fig. 3, we can see that point E is the first point, which will connect the ith cluster. We conclude that the distance e_4 of edge DE is the minimum value of minimum distances between the samples in cluster i and the samples in other clusters, because the minimum distances between clusters must be included in the minimum spanning tree of all data points, and the first point that will connect the ith cluster is the minimum distance between cluster i and other clusters according to the AHC algorithm with single linkage. Point F is not the first point that will connect the ith cluster and the distance e_5 of edge CF is greater than the distance e_4 of edge DE; thus, edge CF is not the minimum distance between cluster i and other clusters. From the above analysis, the intercluster separation $sd(i)$ can be easily calculated. In Fig. 3, $sd(i) = e_4$.

In essence, the proposed ONCD algorithm includes two processes; one is the clustering process of using the AHC algorithm with single linkage, while the other is the evaluating process of using the CSP index. The clustering process of using AHC algorithm with single linkage is also the process

Algorithm 3 ONCD2

Step 1: Select the search range $[k_{\min}, k_{\max}]$ for clustering numbers.
Step 2: Use the single linkage measure to create a hierarchical cluster tree.
Step 3: For $k = k_{\min}$ to $k_{\max}$
  Step a: Construct k clusters from the hierarchical cluster tree.
  Step b: Calculate the value of a certain index.
Step 4: Calculate the optimal number of clusters according to a series of the certain index values.
Step 5: Output the optimal number of clusters, validity index values and clustering results.

of producing the minimum spanning tree, from which we can easily obtain the weight of the minimum spanning tree for all samples in each cluster and the minimum intercluster distance of each cluster. The time complexity of Step 2 is $O(n^3)$. In step 3, the time complexity of Step a is $O(n)$, Step b is $O(n)$, Step c is $O(c)$, Step d is $O(c)$, and Step e is $O(c)$. Usually, we set $k_{\min} = 2$ and $k_{\max} = \lfloor\sqrt{n}\rfloor$ according to the commonly used rule of thumb $k_{\max} \leq \sqrt{n}$ [20]–[22]; thus, the time complexity of Step 3 is

$$O\left(\sum_{c=2}^{\sqrt{n}}(n+n+c+c+c)\right) = O(\sqrt{n}*n) = O(n^{3/2}).$$

The time complexity of Step 4 is $O(\sqrt{n}) = O(n^{1/2})$; the total time complexity of ONCD is $O(n^3)$.

V. Experimental Studies

To verify the performance of the CSP index and ONCD algorithm, this paper uses 18 synthetic data sets and three real data sets to test the CSP index and compares it with other indexes, including CH, DB, Sil, KL, Wint, and V_λ [9]. Moreover, we use adjusted Rand index [23] to evaluate the partition of the eighteen synthetic data sets and the three real data sets. Based on the AHC algorithm with single linkage and the CH index, DB index, Sil index, KL index, Wint index, or V_λ index, the optimal number of clusters determination (ONCD2) algorithm can be described in Algorithm 3.

Usually, the search range for clustering numbers is [2, $k_{\max}$]; we choose $k_{\max} = \lfloor\sqrt{n}\rfloor$. Moreover, we use Pearson's correlation coefficient as the similarity measure to construct the similarity matrix of the V_λ index.

In the following experimental results, we set the following rule: the optimal clustering number is obtained by using the CSP index, which means that the ONCD algorithm is used, and the optimal clustering number is obtained by using another index, which means that the ONCD2 algorithm, based on that other index, is used.

In all tables except Tables III and VIII, we use the indexes to denote the algorithms. For example, CSP denotes the ONCD algorithm and CH denotes the ONCD2 algorithm based on the CH index.

TABLE I
Experimental Results of Optimal Clustering Numbers

Dataset	Sample number	Right NC	Optimal number of clusters						
			CH	DB	Sil	KL	Wint	V_λ	CSP
Parallel2	500	2	**2**	14	**2**	8	9	**2**	**2**
Parallel4	990	4	**4**	14	**4**	**4**	3	2	**4**
Parallel6	1800	6	2	39	2	36	3	2	**6**
Cross2	600	2	**2**	6	**2**	12	4	4	**2**
Cross3	510	3	**3**	22	2	**3**	4	22	**3**
Cross5	1320	5	**5**	25	3	13	4	17	**5**
Segment6	600	6	**6**	12	**6**	2	3	2	**6**
Semicircle2	800	2	13	8	6	13	3	27	**2**
Semicircle3	1200	3	10	32	2	28	4	34	**3**
Semicircle4	1960	4	40	43	40	37	25	2	**4**
Spiral2	1586	2	7	32	7	6	24	**2**	**2**
Circle2	800	2	19	20	20	19	7	9	**2**
Circle3	1800	3	42	42	42	8	34	16	**3**
Circle4	3000	4	51	51	51	51	2	2	**4**
Circle5	3600	5	58	58	59	16	58	2	**5**
Ring2	1100	2	16	31	17	10	3	9	**2**
Norm3	600	3	**3**	**3**	**3**	**3**	**3**	2	**3**
Norm4	800	4	**4**	**4**	**4**	**4**	3	3	**4**

A. Experiments Using Synthetic Data Sets

The experiments include 18 synthetic data sets, which comprise 2-D random numbers generated by computer simulation. These data sets are Parallel2, Parallel4, Parallel6, Cross2, Cross3, Cross5, Segment6, Semicircle2, Semicircle3, Semicircle4, Spiral2, Circle2, Circle3, Circle4, Circle5, Ring2, Norm3, and Norm4. In these data sets, the structures of the Parallel2, Parallel4, Parallel6, Cross2, Cross3, Cross5, and Segment6 data sets are linear, the structures of the Semicircle2, Semicircle3, Semicircle4, and Spiral2 data sets are manifold, the structures of the Circle2, Circle3, Circle4, Circle5, and Ring2 data sets are annular, and the structures of the Norm3 and Norm4 data sets are convex. The structure distributions of the 18 data sets are shown in Fig. 4.

Regarding the Parallel2 data set, its right clustering number is 2. The experimental results of using seven validity indexes to determine the optimal clustering numbers are shown in Fig. 5, where we see that the optimal clustering numbers obtained by the CH index, Sil index, V_λ index, and CSP index are correct, the DB index obtains the optimal clustering number 14, the KL index obtains the optimal clustering number 8, and the Wint index obtains the optimal clustering number 9. The DB, KL, and Wint indexes obtain the wrong optimal numbers of clusters.

Data set information and the experimental results of using the seven validity indexes to evaluate the optimal numbers of clusters are shown in Table I. NC denotes the number of clusters. From this table, we find that the CSP index could obtain the correct optimal numbers of clusters for the 18 synthetic data sets, the CH index is effective for the Parallel2, Parallel4, Cross2, Cross3, Cross5, Segment6, Norm3, and Norm4 data sets, the Sil index is effective for the Parallel2, Parallel4, Cross2, Segment6, Norm3, and Norm4 data sets, the KL index is effective for the Parallel4, Cross3, Norm3, and Norm4 data sets, the V_λ index could obtain the correct optimal numbers of clusters for the Parallel2 and Spiral2 data sets, the DB index could obtain the correct optimal numbers of clusters

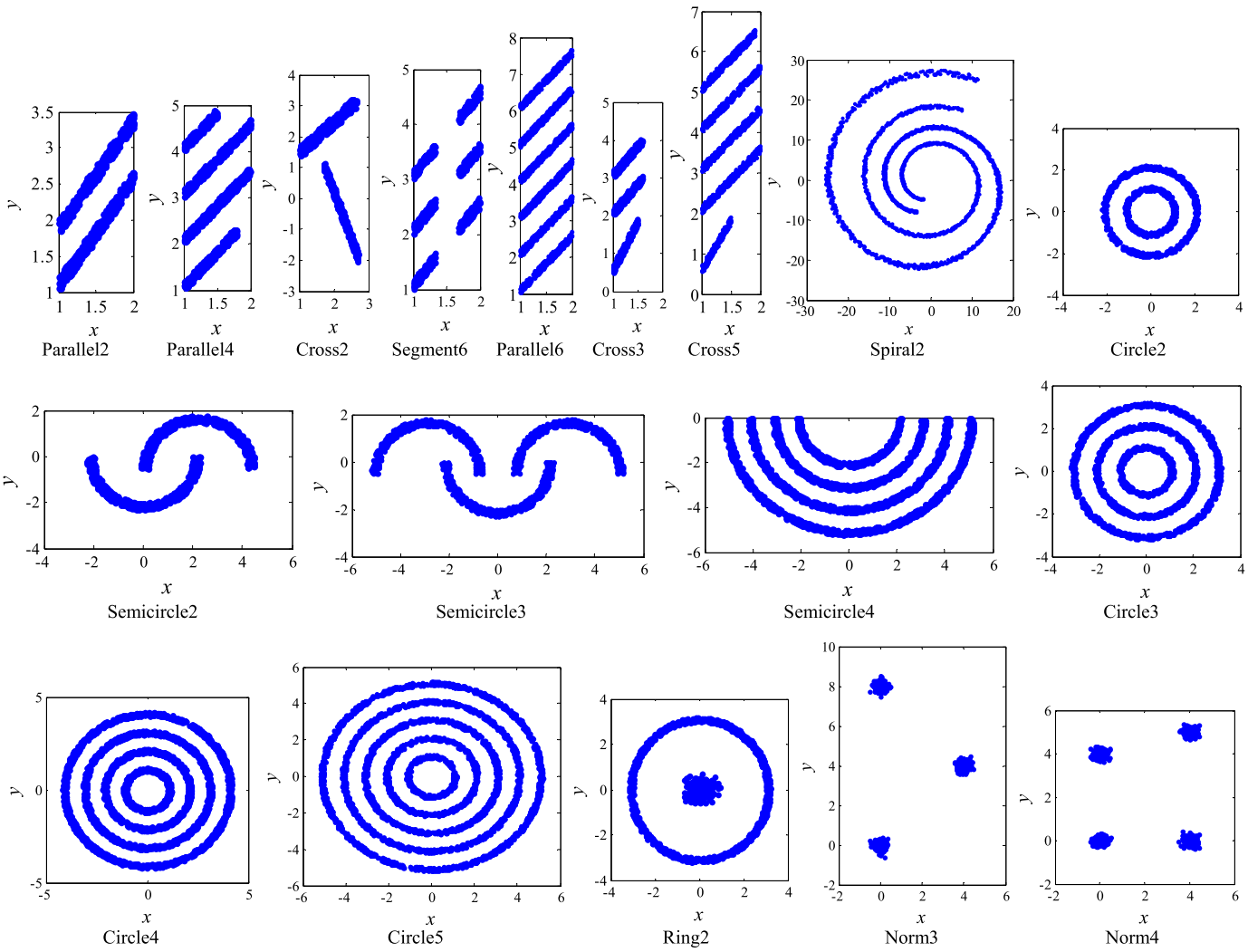

Fig. 4. Structure distribution of eighteen synthetic data sets.

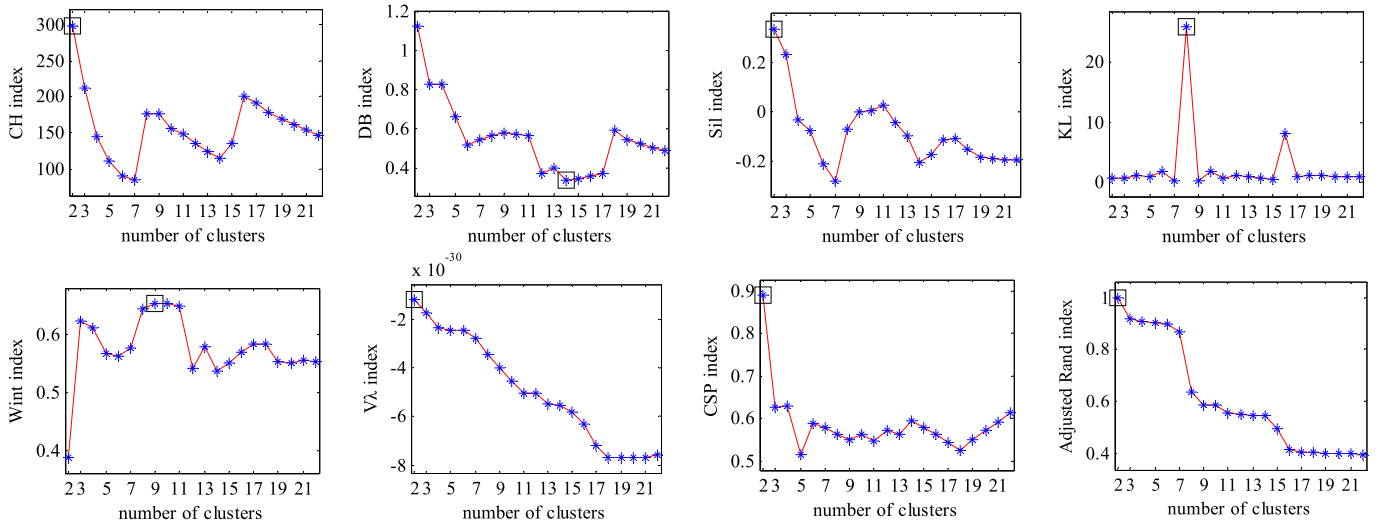

Fig. 5. Clustering number-index relationship graph of Parallel2.

for the Norm3 and Norm4 data sets, and the Wint index is only effective for the Norm3 data set.

Based on the AHC algorithm with single linkage, the experimental results of using the adjusted Rand index to evaluate the partition of the Parallel2 data set are shown in Fig. 5, where we see that the optimal value of the adjusted Rand index is 1 when the number of clusters is 2. This shows that the Parallel2 data set is correctly partitioned for the optimal clustering number.

TABLE II
TIME (SECONDS) OF THE ALGORITHMS USING SYNTHETIC DATA SETS

Dataset		Time (S)						
		CH	DB	Sil	KL	Wint	V_λ	CSP
Parallel2	Mean	0.0400	0.0494	0.9979	0.0460	0.1858	0.2758	**0.0356**
	Std	0.0012	0.0029	0.0043	0.0016	0.0033	0.0018	0.0021
Parallel4	Mean	0.0960	0.1041	3.8810	0.0973	0.7590	1.0497	**0.0800**
	Std	0.0033	0.0032	0.0040	0.0023	0.0031	0.0059	0.0020
Parallel6	Mean	0.2755	0.2961	14.7237	0.2747	2.7060	3.4918	**0.2536**
	Std	0.0058	0.0023	0.0087	0.0025	0.0126	0.0103	0.0044
Cross2	Mean	0.0544	0.0621	1.4258	0.0584	0.2678	0.3669	**0.0474**
	Std	0.0017	0.0014	0.0029	0.0018	0.0050	0.0026	0.0019
Cross3	Mean	0.0478	0.0546	1.0095	0.0527	0.1948	0.2907	**0.0374**
	Std	0.0028	0.0012	0.0021	0.0027	0.0020	0.0023	0.0026
Cross5	Mean	0.1570	0.1719	7.4194	0.1592	1.3851	1.6758	**0.1449**
	Std	0.0023	0.0017	0.0133	0.0009	0.0263	0.0035	0.0024
Segment6	Mean	0.0525	0.0621	1.4017	0.0626	0.2599	0.3744	**0.0452**
	Std	0.0009	0.0027	0.0044	0.0023	0.0051	0.0042	0.0011
Semicircle2	Mean	0.0729	0.0848	2.5438	0.0748	0.4773	0.6807	**0.0664**
	Std	0.0014	0.0010	0.0033	0.0021	0.0038	0.0156	0.0022
Semicircle3	Mean	0.1293	0.1426	6.0852	0.1286	1.1348	1.4676	**0.1140**
	Std	0.0015	0.0009	0.0158	0.0026	0.0034	0.0206	0.0016
Semicircle4	Mean	0.2952	0.3205	18.8970	0.2920	3.3633	4.1265	**0.2664**
	Std	0.0057	0.0028	0.0123	0.0014	0.0127	0.0086	0.0048
Spiral2	Mean	0.1961	0.2149	10.7834	0.2012	2.2436	2.4300	**0.1724**
	Std	0.0025	0.0044	0.0066	0.0046	0.0347	0.0355	0.0031
Circle2	Mean	0.0771	0.0876	2.5252	0.0795	0.5019	0.7114	**0.0673**
	Std	0.0016	0.0013	0.0081	0.0019	0.0140	0.0052	0.0009
Circle3	Mean	0.2644	0.2946	15.6332	0.2682	2.8074	3.3933	**0.2481**
	Std	0.0029	0.0053	0.0400	0.0020	0.0177	0.0390	0.0024
Circle4	Mean	0.6242	0.7054	51.3719	0.6274	8.5777	9.1182	**0.5901**
	Std	0.0019	0.0036	0.2239	0.0024	0.0322	0.0111	0.0074
Circle5	Mean	0.8670	0.9399	82.9258	0.8651	13.0858	14.4668	**0.8187**
	Std	0.0022	0.0022	0.0406	0.0019	0.0270	0.0583	0.0022
Ring2	Mean	0.1110	0.1227	4.9342	0.1191	1.0572	1.1772	**0.0994**
	Std	0.0026	0.0021	0.0092	0.0034	0.0297	0.0235	0.0019
Norm3	Mean	0.0545	0.0611	1.4948	0.0547	0.2981	0.3865	**0.0472**
	Std	0.0014	0.0013	0.0072	0.0017	0.0032	0.0032	0.0009
Norm4	Mean	0.0795	0.0893	2.9823	0.0782	0.5303	0.6864	**0.0727**
	Std	0.0018	0.0015	0.0048	0.0006	0.0055	0.0037	0.0010

TABLE III
NUMBERS OF CLUSTERS OBTAINED BY DBSCAN FOR 18 SYNTHETIC DATA SETS

Dataset	Parallel2	Parallel4	Parallel6	Cross2	Cross3	Cross5	Segment6	Semicircle2	Semicircle3
DBSCAN NC	2	5	7	2	4	6	6	2	3
Dataset	Semicircle4	Spiral2	Circle2	Circle3	Circle4	Circle5	Ring2	Norm3	Norm4
DBSCAN NC	5	2	3	4	5	6	2	3	4

When given the optimal clustering numbers, the experimental results of the other 17 synthetic data sets are the same as those of the Parallel2 data set. Thus, we draw a conclusion that the 18 synthetic data sets are correctly partitioned when given the optimal numbers of clusters.

The running times of the algorithms for determining the optimal number of clusters using the 18 synthetic data sets, with the means and standard deviations obtained after executing every algorithm ten times, are listed in Table II. By observing Table II, we can find that the ONCD algorithm is faster than the ONCD2 algorithms based on the other indexes and that the ONCD2 algorithm based on the Sil index is the most time-consuming one.

To verify the effectiveness of the proposed algorithm, we study the DBSCAN [24] clustering algorithm. Based on a density method, DBSCAN is designed to discover clusters for arbitrary shape. By setting two parameters, i.e., Eps and MintPts, DBSCAN can obtain the number of clusters and clustering result for a data set. According to the parameter settings of reference [24], [25] for Eps and MintPts, we use DBSCAN to cluster the 18 synthetic data sets. The numbers of clusters obtained by DBSCAN are shown in Table III, where we can see that DBSCAN is effective for the Parallel2, Cross2, Segment6, Semicircle2, Semicircle3, Spiral2, Ring2, Norm3, and Norm4 data sets but is not effective for the other nine data sets.

For every synthetic data set, we use the random simulation method to generate ten data sets. According to the above experimental process, we conduct experiments for 180 data sets. The experimental results of the Parallel2 data set are

TABLE IV
OPTIMAL CLUSTERING NUMBERS OF PARALLEL2 DATA SET

Dataset	Right NC	Optimal number of clusters						
		CH	DB	Sil	KL	Wint	V_λ	CSP
Parallel2_1	2	**2**	14	**2**	8	9	**2**	**2**
Parallel2_2	2	8	14	**2**	5	6	4	**2**
Parallel2_3	2	13	21	5	11	4	**2**	**2**
Parallel2_4	2	9	16	**2**	14	5	3	**2**
Parallel2_5	2	13	15	**2**	10	6	**2**	**2**
Parallel2_6	2	5	15	**2**	3	7	**2**	**2**
Parallel2_7	2	20	15	**2**	20	10	**2**	**2**
Parallel2_8	2	12	22	8	15	3	**2**	**2**
Parallel2_9	2	18	21	**2**	14	6	**2**	**2**
Parallel2_10	2	9	19	9	10	7	**2**	**2**
NC Correct rate		10%	0%	70%	0%	0%	80%	100%

TABLE V
ACCURACY OF OPTIMAL CLUSTERING NUMBERS

Dataset	NC Correct rate						
	CH	DB	Sil	KL	Wint	V_λ	CSP
Parallel2	10%	0%	70%	0%	0%	80%	**100%**
Parallel4	70%	0%	30%	0%	0%	0%	**100%**
Parallel6	50%	0%	0%	0%	0%	0%	**100%**
Cross2	70%	0%	**100%**	0%	0%	0%	**100%**
Cross3	80%	0%	0%	30%	0%	0%	**100%**
Cross5	70%	0%	0%	0%	0%	10%	**90%**
Segment6	**100%**	0%	**100%**	10%	0%	0%	**100%**
Semicircle2	0%	0%	50%	0%	0%	20%	**100%**
Semicircle3	60%	0%	30%	0%	0%	0%	**100%**
Semicircle4	0%	0%	0%	10%	0%	0%	**100%**
Spiral2	0%	0%	40%	0%	0%	40%	**100%**
Circle2	0%	0%	0%	10%	0%	20%	**100%**
Circle3	0%	0%	0%	0%	0%	0%	**90%**
Circle4	0%	0%	0%	0%	0%	0%	**90%**
Circle5	0%	0%	0%	0%	0%	0%	**90%**
Ring2	0%	0%	0%	0%	0%	0%	**100%**
Norm3	**100%**	**100%**	**100%**	**100%**	**100%**	0%	**100%**
Norm4	**100%**	**100%**	**100%**	**100%**	20%	0%	**100%**

shown in Table IV and the accuracy of every algorithm is listed in Table V. From Table V, we can see that the CSP index is more stable and valid than the other six indexes.

B. Experiments Using Real Data Sets

The experiments include three real data sets: Vertebral Column (Column_2C), Statlog Heart (Heart), and Teaching Assistant Evaluation (Tae), which are from the UCI Machine Learning Repository (http://archive.ics.uci.edu/ml/). Data set information and the experimental results of using seven validity indexes to evaluate the optimal numbers of clusters are shown in Table VI. From this table, we find that the CSP index could obtain the correct optimal numbers of clusters for the three real data sets, the CH, Sil, and Wint indexes are effective for the Column_2C data set and Heart data set, and the DB, KL, and V_λ indexes are invalid for the three data sets.

Based on the AHC algorithm with single linkage, the experimental results of using the adjusted Rand index to evaluate the partition of the Column_2C data set are shown in Fig. 6, where we see that the value of the adjusted Rand index is not equal to 1 when the number of clusters is 2.

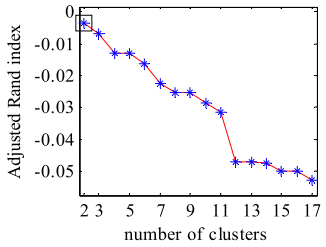

Fig. 6. Clustering number-index relationship of Column_2C.

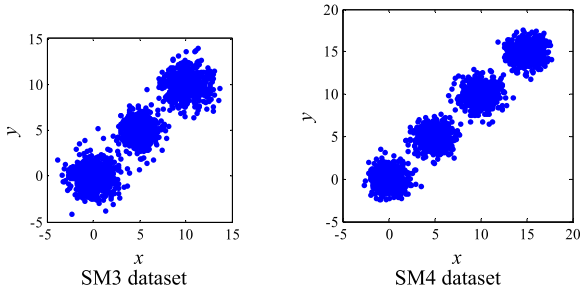

Fig. 7. Structure distributions of SM3 and SM4 data sets.

This shows that some samples of the Column_2C data set are not correctly partitioned for the optimal clustering number. When given the optimal clustering numbers, the experimental results of the other two data sets also show that the partitions of some samples are not correct. Although some samples of the three real data sets are not correctly partitioned, the CSP index could evaluate the clustering quality of data sets integrally and obtain the correct optimal numbers of clusters for the three real data sets.

The running times of the algorithms for determining the optimal number of clusters using the three real data sets, with the means and standard deviations obtained after executing every algorithm ten times, are listed in Table VII. By observing Table VII, we can find that the ONCD algorithm is faster than the ONCD2 algorithms based on the other indexes.

Moreover, we use DBSCAN to cluster the three real data sets. The numbers of clusters obtained by DBSCAN are shown in Table VIII, from which we can see that DBSCAN cannot cluster the Heart data set, can obtain the clustering number 1 for the Column_2C data set, and obtain the clustering number 2 for the Tae data set. DBSCAN is invalid for the three real data sets.

VI. DISCUSSION

The CSP index is a commonly designed validity index based on the idea of the nearest neighbor. The AHC algorithm with single linkage is relatively stable and effective for linear, manifold, annular, and convex data. Because two parameters of the CSP index can be obtained in the clustering process of using the AHC algorithm with single linkage, we propose the ONCD algorithm, which is valid and efficient for the above data sets. The AHC algorithm with single linkage has some limitations, such as having difficulties in clustering for overlapping data sets and showing a high complexity for large data sets. To extend our method, we can use the CSP index and

TABLE VI
OPTIMAL CLUSTERING NUMBERS OBTAINED BY SEVEN VALIDITY INDEXES FOR THREE REAL DATA SETS

Dataset	Sample number	Dimension	Real NC	Optimal number of clusters						
				CH	DB	Sil	KL	Wint	V_λ	CSP
Column_2C	310	6	2	**2**	11	**2**	12	**2**	17	**2**
Heart	270	13	2	**2**	9	**2**	5	**2**	16	**2**
Tae	151	5	3	5	10	5	5	4	10	**3**

TABLE VII
TIME (SECONDS) OF THE ALGORITHMS USING REAL DATA SETS

Dataset		Time (S)						
		CH	DB	Sil	KL	Wint	V_λ	CSP
Column_2C	Mean	0.0401	0.0391	0.4172	0.0408	0.0981	0.1283	**0.0339**
	Std	0.0006	0.0013	0.0018	0.0006	0.0048	0.0019	0.0022
Heart	Mean	0.0347	0.0334	0.3647	0.0361	0.0759	0.1047	**0.0253**
	Std	0.0024	0.0029	0.0029	0.0022	0.0039	0.0032	0.0021
Tae	Mean	0.0171	0.0195	0.1272	0.0169	0.0400	0.0415	**0.0125**
	Std	0.0005	0.0002	0.0026	0.0001	0.0021	0.0030	0.0010

Algorithm 4 EONCD

Step 1: Select the search range $[k_{\min}, k_{\max}]$ for clustering numbers.
Step 2: For $k = k_{\min}$ to $k_{\max}$
  Step a: Use a certain clustering algorithm to cluster the sample dataset.
  Step b: Use (5) to calculate the CSP index value of a single cluster.
  Step c: Use (6) to calculate the average value of the CSP index.
Step 3: Use (7) to calculate the optimal number of clusters.
Step 4: Output the optimal number of clusters, validity index values and clustering results.

TABLE VIII
NUMBERS OF CLUSTERS OBTAINED BY DBSCAN

Dataset	Column_2C	Heart	Tae
DBSCAN NC	1	0	2

other clustering algorithms to determine the optimal numbers of clusters for more data sets. The extended ONCD (EONCD) algorithm can be described in Algorithm 4.

When we use the EONCD algorithm based on the K-means clustering [26] or affinity propagation clustering [27] algorithm, we can obtain the correct optimal numbers of clusters for the Norm3 and Norm4 data sets. We can also use this algorithm to obtain the correct optimal numbers of clusters for the SM3 and SM4 data sets with overlapping clusters. The optimal clustering numbers are 3 and 4 for the SM3 and SM4 data sets, respectively. The structure distributions of the SM3 and SM4 data sets are shown in Fig. 7.

When we use the EONCD algorithm based on the kernel K-means clustering [28] algorithm, we can obtain the correct optimal numbers of clusters for the Cross2, Cross3, Segment6, Ring2, Norm3, Norm4, SM3, and SM4 data sets by setting the Gaussian kernel function.

When we use the EONCD algorithm based on the spectral clustering [29] algorithm, we can obtain the correct optimal numbers of clusters for the Parallel2, Parallel4, Parallel6, Cross2, Cross3, Cross5, Segment6, Semicircle2, Semicircle3, Semicircle4, Spiral2, Circle2, Ring2, Norm3, and Norm4 data sets. Moreover, the spectral clustering algorithm can be used to cluster large data sets [30], [31]; thus, the EONCD algorithm may also be used for large data sets.

VII. CONCLUSION

The hierarchical clustering algorithm can obtain stable clustering results. Therefore, it has become one of the main clustering methods. From the perspective of sample geometry, this paper designs a new clustering validity index called the CSP index. Based on the AHC algorithm with single linkage and the CSP clustering validity index, the ONCD algorithm is proposed to analyze the clustering effect of data sets and determine the optimal number of clusters. Theoretical research and experimental results show that the proposed index and the method could evaluate the clustering results effectively and are suitable for determining the optimal number of clusters for linear, manifold, annular, and convex data sets.

ACKNOWLEDGMENT

The authors would like to thank the Associate Editor and anonymous reviewers for their insightful comments and valuable suggestions. They would also like to thank Dr. Kaijun Wang of Fujian Normal University for his help in some experiments.

REFERENCES

[1] C. Zhong, D. Miao, and P. Fränti, "Minimum spanning tree based split-and-merge: A hierarchical clustering method," *Inf. Sci.*, vol. 181, no. 16, pp. 3397–3410, Aug. 2011.
[2] I. Gurrutxaga *et al.*, "SEP/COP: An efficient method to find the best partition in hierarchical clustering based on a new cluster validity index," *Pattern Recognit.*, vol. 43, no. 10, pp. 3364–3373, Oct. 2010.
[3] E. Nasıbov and C. Kandemır-Cavas, "OWA-based linkage method in hierarchical clustering: Application on phylogenetic trees," *Expert Syst. Appl.*, vol. 38, no. 10, pp. 12684–12690, Sep. 2011.

[4] Y. Jung, H. Park, D. Z. Du, and B. L. Drake, "A decision criterion for the optimal number of clusters in hierarchical clustering," *J. Global Optim.*, vol. 25, no. 1, pp. 91–111, Jan. 2003.
[5] K. Mali and S. Mitra, "Clustering and its validation in a symbolic framework," *Pattern Recognit. Lett.*, vol. 24, no. 14, pp. 2367–2376, Oct. 2003.
[6] S. Yue, J.-S. Wang, T. Wu, and H. Wang, "A new separation measure for improving the effectiveness of validity indices," *Inf. Sci.*, vol. 180, no. 5, pp. 748–764, Mar. 2010.
[7] C. Zhong, D. Miao, R. Wang, and X. Zhou, "DIVFRP: An automatic divisive hierarchical clustering method based on the furthest reference points," *Pattern Recognit. Lett.*, vol. 29, no. 16, pp. 2067–2077, Dec. 2008.
[8] L. Chen, Q. Jiang, and S. Wang, "A hierarchical method for determining the number of clusters," *J. Softw.*, vol. 19, no. 1, pp. 62–72, Jan. 2008.
[9] X.-Q. Hu, R.-N. Ma, and B.-J. Zhong, "Study on validity of hierarchical clustering," *J. Shandong Univ.*, vol. 40, no. 5, pp. 146–149, Oct. 2010.
[10] I. Cattinelli, G. Valentini, E. Paulesu, and N. A. Borghese, "A novel approach to the problem of non-uniqueness of the solution in hierarchical clustering," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 7, pp. 1166–1173, Jul. 2013.
[11] C.-R. Lin and M.-S. Chen, "Combining partitional and hierarchical algorithms for robust and efficient data clustering with cohesion self-merging," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 2, pp. 145–159, Feb. 2005.
[12] R. R. Yager, "Intelligent control of the hierarchical agglomerative clustering process," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 30, no. 6, pp. 835–845, Dec. 2000.
[13] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. New York, NY, USA: Wiley, 2001, pp. 550–554.
[14] T. Caliński and J. Harabasz, "A dendrite method for cluster analysis," *Commun. Statist.*, vol. 3, no. 1, pp. 1–27, Jan. 1974.
[15] D. L. Davies and D. W. Bouldin, "A cluster separation measure," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 1, no. 2, pp. 224–227, Apr. 1979.
[16] P. J. Rousseeuw, "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis," *J. Comput. Appl. Math.*, vol. 20, no. 1, pp. 53–65, Nov. 1987.
[17] S. Dudoit and J. Fridlyand, "A prediction-based resampling method for estimating the number of clusters in a dataset," *Genome Biol.*, vol. 3, no. 7, pp. 1–21, Jun. 2002.
[18] E. Dimitriadou, S. Dolničar, and A. Weingessel, "An examination of indexes for determining the number of clusters in binary data sets," *Psychometrika*, vol. 67, no. 1, pp. 137–160, Jan. 2002.
[19] K.-J. Wang, J. Li, J.-Y. Zhang, and L.-X. Guo, "Experimental comparison of clusters number estimation for cluster analysis," *Comput. Eng.*, vol. 34, no. 9, pp. 198–199, May 2008.
[20] N. R. Pal and J. C. Bezdek, "On cluster validity for the fuzzy c-means model," *IEEE Trans. Fuzzy Syst.*, vol. 3, no. 3, pp. 370–379, Aug. 1995.
[21] J. C. Bezdek and N. R. Pal, "Some new indexes of cluster validity," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 28, no. 3, pp. 301–315, Jun. 1998.
[22] J. Yu and Q. Cheng, "Search range of the optimal number of clusters in fuzzy clustering," *Sci. China (Ser. E)*, vol. 32, no. 2, pp. 274–280, Apr. 2002.
[23] L. Hubert and P. Arabie, "Comparing partitions," *J. Classification*, vol. 2, no. 1, pp. 193–218, Dec. 1985.
[24] M. Ester, H. P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. 2nd Int. Conf. Knowl. Discovery Data Mining*, 1996, pp. 226–231.
[25] M. Daszykowski, B. Walczak, and D. L. Massart, "Looking for natural patterns in data: Part 1. Density-based approach," *Chemometrics Intell. Lab. Syst.*, vol. 56, no. 2, pp. 83–92, May 2001.
[26] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proc. 5th Berkeley Symp. Math. Statist. Probab.*, 1967, pp. 281–297.
[27] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, no. 5814, pp. 972–976, Feb. 2007.
[28] M. Gönen and A. A. Margolin, "Localized data fusion for kernel k-means clustering with application to cancer biology," in *Proc. NIPS*, Montreal, QC, Canada, 2014, pp. 1305–1313.
[29] U. von Luxburg, "A tutorial on spectral clustering," *Statist. Comput.*, vol. 17, no. 4, pp. 395–416, Dec. 2007.
[30] D. Cai and X. Chen, "Large scale spectral clustering via landmark-based sparse representation," *IEEE Trans. Cybern.*, vol. 45, no. 8, pp. 1669–1680, Aug. 2015.
[31] S. Ding, H. Jia, and Z. Shi, "Spectral clustering algorithm based on adaptive Nyström sampling for big data analysis," *J. Softw.*, vol. 25, no. 9, pp. 2037–2049, Sep. 2014.

Shibing Zhou received the Ph.D. degree in light industry information technology and engineering from Jiangnan University, Wuxi, China, in 2011.

He is currently a Lecturer with the Department of Computer Science and Technology, Jiangnan University. His current research interests include pattern recognition, artificial intelligence, and their applications.

Zhenyuan Xu received the M.S. degree in applied mathematics from Anhui University, Hefei, China, in 1981.

He is currently a Professor with the School of Science, Jiangnan University, Wuxi, China. His current research interests include artificial intelligence, chaos, and bioinformatics.

Fei Liu received the Ph.D. degree in control science and engineering from Zhejiang University, Hangzhou, China, in 2002.

He is currently a Professor with the Institute of Automation, Key Laboratory of Advanced Process Control for Light Industry (Ministry of Education), Jiangnan University, Wuxi, China. His current research interests include advanced control theory and application, batch process control engineering, statistical monitoring and diagnosis in industrial process, and intelligent technique and systems.