I.J. Image, Graphics and Signal Processing, 2011, 1, 31-37
Published Online February 2011 in MECS (http://www.mecs-press.org/)

Clustering Belief Functions using Extended Agglomerative Algorithm

Ying Peng
Postgraduate College, Academy of Equipment Command & Technology, Beijing, China
Email: py_hehe@yahoo.com.cn

Huairong Shen
Department of Space Equipment, Academy of Equipment Command & Technology, Beijing, China
Email: shenhuair@tom.com

Zenghui Hu
College of Electronic Science and Technology, National University of Defense Technology, Changsha, China
Email:zhhunudt@163.com

Yongyi Ma
Postgraduate College, Academy of Equipment Command & Technology, Beijing, China
Email: mymailmayy@163.com

***Abstract*—Clustering belief functions is not easy because of uncertainty and the unknown number of clusters. To overcome this problem, we extend agglomerative algorithm for clustering belief functions. By this extended algorithm, belief distance is taken as dissimilarity measure between two belief functions, and the complete-link algorithm is selected to calculate the dissimilarity between two clusters. Before every merging of two clusters, consistency test is executed. Only when the two clusters are consistent, they can merge, otherwise, dissimilarity between them is set to the largest value, which prevents them from merging and assists to determine the number of final clusters. Typical illustration shows same promising results. Firstly, the extended algorithm itself can determine the number of clusters instead of needing to set it in advance. Secondly, the extended algorithm can deal with belief functions with hidden conflict. At last, the algorithm extended is robust.**

***Index Terms*—belief functions, clustering, belief distance, agglomerative algorithm**

I. Introduction

Dempster-Shafer theory (DST) [1,2] is a mathematical tools developed in the 1970s for reasoning under uncertainty. Its strength exists in that it can efficiently cope with imprecise and uncertain information without prior information, thus it is extensively used in many fields, such as information fusion, uncertainty reasoning, pattern recognition, comprehensive diagnosis, etc. The information carrier in DST is belief function. Combination of belief functions is required for getting a fusion result. And combination is performed just on condition that the belief functions are related to the same event. However, it is usually happened that belief functions related to different events are mixed up, which prevents us from combining them directly. Consequently, it is necessary to distinguish which belief functions are reporting on which event.

Clustering of belief functions can resolve this partition problem. Clustering is an approach that partition data into different clusters. Data in the same cluster are more similar to each other than to members in other clusters. Clustering algorithms such as k-means [3] and hierarchical algorithm [4,5] are the most popular ones for clustering the usual data point. However, different from the usual data point, belief function is uncertain information which cannot be expressed with data point, and the dissimilarity measure between belief functions is more special. Hence, the clustering approaches for data point cannot be used directly to partition belief functions.

So far, the clustering approaches for belief functions can be classified into two main categories: (1) Direct clustering, such as approaches proposed by reference [6], reference [7] and [8]. Approach in reference [6] is based on decomposing of belief functions and Potts spins mean field theory. However, it is complicated and need to set the number of clusters in advance. Compared with this method, approaches in reference [7] and [8] are simpler. Approach in reference [7] is based on k-modes algorithm [9] and belief distance, but it still needs to set the number of clusters, and the results are non-unique due to the selecting of different initialization seed beliefs. Approach presented by reference [8] is based on belief distance and the number of clusters is depended on the threshold values. However, the threshold values are hard to

Corresponding author: Ying Peng.

determine. Besides, the approach cannot cope with belief functions with hidden conflict [10]. (2) Indirect clustering, it first needs to transform belief functions into Euclidean characters by TBM [11] or a probabilistic transformation *DSmP* [12], then clustering algorithms for usual data point can be used. Thereby, approaches in this category are simpler. While the problem is the possible inequality of transformation, and most of the clustering approaches for data point still need to set the number of clusters in advance except for hierarchical algorithm.

This paper is to develop a direct clustering approach which can simultaneously determine the appropriate number of clusters. The rest of the paper is organized as follows. Some definitions in DST are reviewed in Section II. Agglomerative algorithm which is a branch of hierarchical algorithm is presented in Section III. Then, in Section IV, the extended agglomerative algorithm for clustering belief functions is proposed. Three examples are illustrated to validate the effectiveness of the extended agglomerative algorithm in Section V. Finally, we conclude in Section VI.

II. Review of DST

We list a few definitions necessary in DST to avoid misunderstanding.

Definition 1. (Frame of discernment) The frame of discernment is a finite set of mutually exclusive elements, denoted as Ω hereafter.

Definition 2. (Basic belief assignment) A basic belief assignment (bba) is a mapping m from $2^{\Omega} \rightarrow [0,\ 1]$ that satisfies $\sum_{A\subseteq\Omega} m(A)=1$. The basic belief mass (bbm) $m(A)$, $A\subseteq\Omega$, is the value taken by the bba at A.

When $m(A)>0$, A is the focal elements of a bba.

Definition 3. (Dempster's rule) Let m_1 and m_2 be two bbas defined on frame Ω which are derived from two distinct sources. Dempster's rule of combination two belief functions is given by $m=m_1\oplus m_2$, where

$$m(A)=\frac{\sum_{A_i\cap B_j=A} m_1(A_i)m_2(B_j)}{1-k},\quad A\subseteq\Omega, A\neq\phi, k\neq 1 \quad (1)$$

$k=\sum_{A_i\cap B_j=\phi} m_1(A_i)m_2(B_j)$. k is the mass of the combined belief assigned to the emptyset before normalization and called conjunctive conflict for short in this paper.

Definition 4. （Belief distance）[13] Let m_1 and m_2 be two bbas defined on frame Ω, the distance between m_1 and m_2 is given by

$$d_{BPA}(m_1,m_2)=\sqrt{\frac{1}{2}\left(\left\langle \vec{m}_1,\vec{m}_1\right\rangle+\left\langle \vec{m}_2,\vec{m}_2\right\rangle-2\left\langle \vec{m}_1,\vec{m}_2\right\rangle\right)} \quad (2)$$

where $\left\langle \vec{m}_1,\vec{m}_2\right\rangle=\sum_{i=1}^{2^N}\sum_{j=1}^{2^N} m_1(A_i)m_2(B_j)\frac{|A_i\cap B_j|}{|A_i\cup B_j|}$. In the following, whenever we use d_{BPA} or $d_{BPA}(m_1,m_2)$, we always associate it with belief distance. And d_{BPA} meets $0\le d_{BPA}\le 1$.

III. Agglomerative Algorithm

Agglomerative clustering performs in a bottom-up fashion, which initially takes each data points as a cluster and then repeatedly merges clusters until all data points have been merged into a single cluster. The process allows us to decide which level is the most appropriate. Namely we could determine the number of clusters and the clusters by analyzing the hierarchical tree. The agglomerative clustering is more flexible than approaches that need to set the number of clusters first.

Suppose there are n data points. Steps of agglomerative algorithm are described as follows:

Step1 Every data point in a different cluster, and there are n clusters;

Step2 Calculate the dissimilarity between any two clusters;

Step3 Merge two clusters that have smallest dissimilarity, and the number of clusters subtracts 1;

Step4 Repeated step (2) and (3), stop when the number of clusters gets 1.

From the steps above, we can get that the key factor of agglomerative algorithm is the dissimilarity measure between two clusters. The most representative algorithms that measure the dissimilarity between two clusters are single-link [14], complete-link [15], average-link [16], etc.

IV. Extended Agglomerative Algorithm For Clustering Belief Functions

The standard agglomerative algorithm which is used for clustering of usual data points shows serious limitations for dealing with belief functions. Firstly, the belief function is uncertain data, so the dissimilarity measure between them is naturally different from that of between the usual data points. Secondly, smallest dissimilarity is not adequate for merging clusters of belief functions because of the possible hidden conflict [10].

To tackle these problems, the agglomerative algorithm is extended from two aspects: ① Belief distance is used to calculate the dissimilarity. ② Consistency test is put forward to control the merging of clusters. If two clusters with smallest dissimilarity are consistent, they can merge, vice versa. Consistency test brings two advantages: one is to avoid the hidden conflict in each cluster, another one benefits the determining the number of clusters. These two issues will be discussed in the following sections.

The dissimilarity measure and the consistency test are presented in section *A* and *B* respectively. Section *C* explains how to determine the number of clusters. The algorithm design is provided in section *D*.

A. Dissimilarity Measure

Belief distance defined by (2) is taken as the dissimilarity measure between two belief functions.

The complete-link algorithm is select to measure the dissimilarity between two clusters. Hence, the dissimilarity between two clusters is defined as follows.

$$d(r,s)=\max\left(d_{BPA}\left(e_{ri},e_{sj}\right)\right), i\in\left(1,\cdots,n_r\right), j\in\left(1,\cdots,n_s\right) \quad (3)$$

where d_{BPA} is the belief distance. n_r is the number of beliefs in cluster r and n_s is the number of beliefs in cluster s. e_{ri} is the *i*th belief in the cluster r and e_{sj} is the *j*th belief in the cluster s. The range of dissimilarity between clusters is 0 to 1.

It is clear that the dissimilarity between two clusters is the furthest dissimilarity between belief functions in these two clusters. Find two clusters that have smallest dissimilarity by: $d(k,l)=\min\limits_{r\neq s}\left(d(r,s)\right)$. Clusters k and l are called the furthest neighbors. And it is obviously, $0\leq d(k,l)\leq 1$.

B. Consistency test

In this section, two definitions that are important for consistency test are proposed first. One is leading element in a bba and another one is leading element in a cluster.

Definition 5. (Leading element in a bba) The leading element in a bba is the union of focal elements that get the highest mass except for the frame of discernment Ω when $m(\Omega)\neq 1$.

When $m(\Omega)=1$, the leading element in this bba is Ω.

Example below is enumerated to describe the leading elements concretely.

Example 1 Let $\Omega=\{\omega_1,\omega_2,\omega_3\}$ be a frame of discernment. Suppose four distinct belief functions, defined on Ω, are given by

$m_1(\{\omega_1\})=0.8, m_1(\{\omega_2\})=0.2$;

$m_2(\{\omega_1\})=0.4, m_2(\{\omega_2\})=0.4, m_2(\{\omega_3\})=0.2$;

$m_3(\{\omega_2,\omega_3\})=0.8, m_3(\Omega)=0.2$;

$m_4(\{\omega_1\})=0.4, m_4(\Omega)=0.6$.

The leading elements of bbas in example 1 are listed in Table I.

TABLE I.
THE LEADING ELEMENTS OF BBAS IN EXAMPLE 1

Bba	$\{m_1\}$	$\{m_2\}$	$\{m_3\}$	$\{m_4\}$
Leading Element	$\{\omega_1\}$	$\{\omega_1,\omega_2\}$	$\{\omega_2,\omega_3\}$	$\{\omega_1\}$

Definition 6. (Leading element in a cluster) The leading element in a cluster is the intersection of leading elements in bbas that belong to this cluster.

If m_1, m_2 and m_4 in example 1 are in one cluster C_1, the leading element in C_1 is $\{\omega_1\}=\{\omega_1\}\cap\{\omega_1,\omega_2\}\cap\{\omega_1\}$.

The definition of consistency test is given as follows.

Definition 7. (Consistency test) If the intersection of leading elements in two clusters is not null, the two clusters are consistent, otherwise, they are inconsistent.

If two clusters are inconsistent, the two clusters should not merge, i.e., they should be in different clusters. In example 1, since $\{\omega_1\}\cap\{\omega_2,\omega_3\}=\phi$, the third BBA cannot merge with clusters C_1.

Example below is used to show that the consistency test is helpful to avoid hidden conflict.

Example 2 Let $\Omega=\{\omega_1,\omega_2,\omega_3\}$ be a frame of discernment. Suppose three distinct belief functions, defined on Ω, are given by

$m_1(\{\omega_1,\omega_2\})=1$; $m_2(\{\omega_2,\omega_3\})=1$; $m_3(\{\omega_1,\omega_3\})=1$.

Belief distances between any two of them are equal, and any two of them are consistent, which makes them seem to refer to the same event. However, they should not be partitioned into one cluster, because there is hidden conflict among them, i.e., $\{\omega_1,\omega_2\}\cap\{\omega_2,\omega_3\}\cap\{\omega_1,\omega_3\}=\phi$.

If consistency test is executed, the cluster $C_1=\{m_1,m_2\}$ gets the leading element $\{\omega_2\}$ and the leading element of cluster $C_2=\{m_3\}$ is $\{\omega_1,\omega_3\}$. The two leading elements have null intersection $\phi=\{\omega_2\}\cap\{\omega_1,\omega_3\}$. Consequently, the two clusters C_1 and C_2 are inconsistent, which keeps them from merging.

Thereby, it is necessary to implement consistency test before every merging of two furthest neighbors. Just the two furthest neighbors are consistent, they can merge.

C. Determining the number of clusters

To determine the number of clusters without any priori information is a difficult problem. Therefore, a majority of approaches for clustering need to set the number of clusters in advance. Although agglomerative algorithm does not need to set the number of clusters in advance, it is still hard to get the appropriate number of clusters after the finishing of clustering.

In the extended algorithm, the inconsistent furthest neighbors cannot merge, and the largest dissimilarity value prevents two clusters from merging. So the problem of determining the number of clusters is easy to resolve with a simple step: reset the dissimilarity between any inconsistent furthest neighbors to 1 which is the largest value of dissimilarity between clusters.

Since the dissimilarities among inconsistent clusters are all 1, it is easy to get the number of clusters from hierarchical tree: when dissimilarities between any two clusters are 1, the clustering is finished and the current number of clusters is the final number of clusters. This is the main advantage of the extend algorithm.

D. Algorithm Design

Fig. 1 presents a flow chart of the extended algorithm. And nodes in Fig. 1 are interpreted as follows:

1. Dissimilarity = Dissimilarities between any two clusters are calculated.

2. Fur Nei = Two furthest neighbors are found.

3. Cons Text = Consistency test. Are the two furthest neighbors consistent?

4. Merge = Two furthest neighbors merge, and a new cluster c_l is get.

5. Lead Elem = Leading Element. Get the leading element in the new cluster c_l.

6. Dis 1 = Dissimilarity between two clusters is reset as 1.

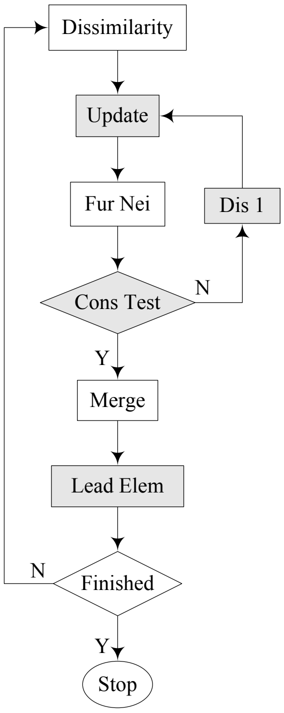

Figure 1. Flowchart of the extended agglomerative algorithm

7. Update = Update the dissimilarity between two clusters.

8. Finished = Does the clustering finish? If dissimilarities between any two clusters are 1, the clustering finishes.

It is obvious that, except for nodes with shadow, Fig. 1 expresses the standard agglomerative algorithm. And it is the very nodes with shadow make the algorithm capable of coping with clustering belief functions.

V. Experiments and Discussion

In this section, the validity of the extended algorithm will be validated by the following three typical examples. The first example is composed of Bayesian belief functions which are the simplest type of belief functions. Clustering belief functions of this category is relatively easy. The second example is made up of categorical beliefs, and it contains hidden conflict. Clustering belief functions of this category is some hard. Normal beliefs constitute the third example, which can be taken as the mix of the first two sorts of belief functions, and it is the most common one in real application.

We also give out the clustering results using the approaches in reference [7] and [8] for comparison. The selecting of these two approaches due to their similarities to our extended agglomerative algorithm: they both belong to direct clustering and belief distance is adopted as the dissimilarity measure. For approach in reference [7], we always set the right number of clusters that determined by our extended agglomerative algorithm, and the initialization seed beliefs (ISB) are specified. For approach in reference [8], threshold value ρ is needed to control clustering process. Hence, we always evaluate appropriate value for ρ in these three examples.

A. Bayesian belief functions

1) Experiment data

Example 3 Let $\Omega=\{\omega_1,\omega_2,\omega_3\}$ be a frame of discernment. Suppose ten distinct belief functions, defined on Ω, are given by

$e_1: m(\{\omega_1\})=0.5, m(\{\omega_2\})=0.2, m(\{\omega_3\})=0.3$;

$e_2: m(\{\omega_1\})=0.0, m(\{\omega_2\})=0.9, m(\{\omega_3\})=0.1$;

$e_3: m(\{\omega_1\})=0.55, m(\{\omega_2\})=0.1, m(\{\omega_3\})=0.35$;

$e_4: m(\{\omega_1\})=0.45, m(\{\omega_2\})=0.2, m(\{\omega_3\})=0.35$;

$e_5: m(\{\omega_1\})=0.1, m(\{\omega_2\})=0.2, m(\{\omega_3\})=0.7$;

$e_6: m(\{\omega_1\})=0.7, m(\{\omega_2\})=0.2, m(\{\omega_3\})=0.1$;

$e_7: m(\{\omega_1\})=0.6, m(\{\omega_2\})=0.2, m(\{\omega_3\})=0.2$;

$e_8: m(\{\omega_1\})=0.2, m(\{\omega_2\})=0.5, m(\{\omega_3\})=0.3$;

$e_9: m(\{\omega_1\})=0.5, m(\{\omega_2\})=0.4, m(\{\omega_3\})=0.1$;

$e_{10}: m(\{\omega_1\})=0.9, m(\{\omega_2\})=0.05, m(\{\omega_3\})=0.05$.

Analyzing the belief functions above, we can get that beliefs e_1, e_3 , e_4, e_6, e_7, e_9, e_{10} all assign the highest probability to hypothesis ω_1, so it is reasonable to consider that they concern the same event and partition them into one cluster. Beliefs e_2 and e_8 are both assign the highest probability to ω_2, so they are potentially partitioned into the same cluster. Only belief e_5 gives ω_3 the highest probability, which separates itself from other two clusters.

2) Result and discussion

Fig. 2 presents the hierarchical tree of example 3 produced by the extended agglomerative algorithm.

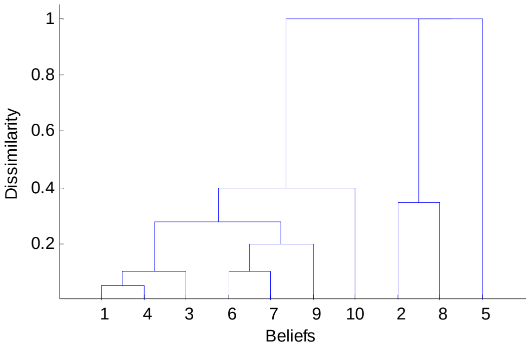

Figure 2. Hierarchical tree of example 3

It is clear that there are three clusters that dissimilarity between any two of them is 1. Therefore, we can easily determine that the number of clusters is 3. The three clusters are also gotten according to hierarchical tree. The clusters and the corresponding leading elements are listed in Table II.

TABLE II.
THE CLUSTERS AND THE CORRESPONDING LEADING ELEMENTS OF EXAMPLE 3 USING THE EXTENDED AGGLOMERATIVE ALGORITHM

Cluster	Leading Element
$C_1=\{e_1,e_3,e_4,e_6,e_7,e_9,e_{10}\}$	$\{\omega_1\}$
$C_2=\{e_2,e_8\}$	$\{\omega_2\}$
$C_3=\{e_5\}$	$\{\omega_3\}$

The clustering results listed in Table II go nicely with the analyses. And we can conclude from Table II that the leading element of each cluster is the event that this cluster concerns.

Table III lists the clustering results of example 3 by using approaches in reference [7] and [8].

TABLE III.
THE CLUSTERS OF EXAMPLE 3 USING APPROACH IN REFERENCE [7] AND [8]

Approach	Cluster
[7] ISB= $\{e_1,e_2,e_3\}$	$C_1=\{e_1,e_3,e_4,e_5,e_8,e_9\}$, $C_2=\{e_2\}$, $C_3=\{e_6,e_7,e_{10}\}$
[7] ISB= $\{e_1,e_8,e_5\}$	$C_1=\{e_1,e_3,e_4,e_6,e_7,e_9,e_{10}\}$, $C_2=\{e_2,e_8\}$, $C_3=\{e_5\}$
[8] $\rho=0.35$	$C_1=\{e_1,e_3,e_4,e_6,e_7,e_9,e_{10}\}$, $C_2=\{e_2,e_8\}$, $C_3=\{e_5\}$
[8] $\rho=0.38$	$C_1=\{e_1,e_3,e_4,e_5,e_7,e_8,e_9\}$, $C_2=\{e_2\}$, $C_3=\{e_6,e_{10}\}$

Utilizing the approach in reference [7], when ISB are $\{e_1,e_2,e_3\}$, we get the wrong clustering result. When ISB= $\{e_1,e_8,e_5\}$, we get the right clustering result. Consequently, the approach in reference [7] is unstable, i.e., the selecting of improper ISB can result in wrong clustering.

When threshold value $\rho=0.35$, approach in reference [8] produces the right clustering. When ρ changes a little, such as $\rho=0.38$, beliefs e_5 and e_8 are in the same cluster with beliefs e_1 , e_3 , e_4 . Obviously, the result is not right. Therefore, the approach in reference [8] is not robust.

B. *Belief functions with hidden conflict*

1) Experiment data

Example 4 Let $\Omega=\{\omega_1,\omega_2,\cdots,\omega_8\}$ be a frame of discernment. Suppose nine distinct belief functions, defined on Ω, are given by

$e_1:m(\{\omega_1,\omega_2\})=1$; $e_2:m(\{\omega_2,\omega_3\})=1$;

$e_3:m(\{\omega_1,\omega_3,\omega_8\})=1$; $e_4:m(\{\omega_4\})=1$;

$e_5:m(\{\omega_6\})=1$; $e_6:m(\{\omega_7,\omega_6\})=1$;

$e_7:m(\{\omega_4,\omega_5\})=1$; $e_8:m(\{\omega_1,\omega_2\})=1$;

$e_9:m(\{\omega_2,\omega_3\})=1$.

Obviously, e_1 and e_8 are identical ones, so they should be in the same cluster, we denote it by cluster c_1^*. e_2 and e_9 should also be in the same cluster because of their sameness, we denoted this cluster as cluster c_2^*. For cluster c_1^*, cluster c_2^*, and belief e_3, any two of them are consistent, which usually mistakes us with partitioning them into one cluster. However, there is a hidden conflict existing in these three ones, i.e., $\{\omega_1,\omega_2\}\cap\{\omega_2,\omega_3\}\cap\{\omega_1,\omega_3,\omega_8\}=\phi$. Hence, beliefs e_1 , e_2 , e_8 , e_9 and e_3 should not be partitioned into one cluster.

Belief e_5 ensures that hypothesis ω_6 is true hypothesis. Belief e_6 supports hypothesis ω_6 or ω_7 as true hypothesis. The two beliefs have intersection ω_6 , therefore, the two beliefs may concern to the same event, and may be in one cluster. Beliefs e_4 and e_7 have their intersection ω_4 as true hypothesis, therefore, the two beliefs may be in one cluster.

2) Result and discussion

Fig. 3 presents the hierarchical tree of example 4.

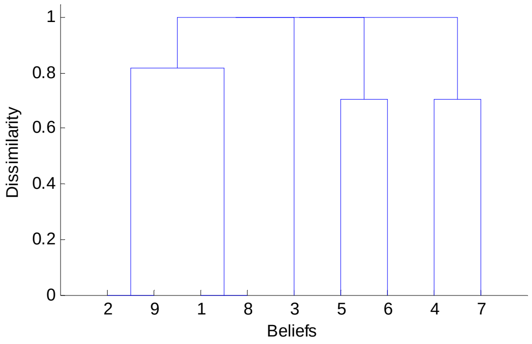

Figure 3. Hierarchical tree of example 4

As shown in Fig. 3, there are four clusters that dissimilarity between any two of them is 1. Consequently, we easily determine that the number of clusters is 4. The four clusters and the corresponding leading elements are listed in Table IV.

TABLE IV.
THE CLUSTERS AND THE CORRESPONDING LEADING ELEMENTS OF EXAMPLE 4 USING THE EXTENDED AGGLOMERATIVE ALGORITHM

Cluster	Leading Element
$C_1=\{e_5,e_6\}$	$\{\omega_6\}$
$C_2=\{e_4,e_7\}$	$\{\omega_4\}$
$C_3=\{e_2,e_9,e_1,e_8\}$	$\{\omega_2\}$
$C_4=\{e_3\}$	$\{\omega_1,\omega_3\}$

From the results listed in Table IV, we obtain that beliefs e_1 , e_2 , e_8 , e_9 and e_3 are in different clusters, which agrees with the analyses above. The extended agglomerative algorithm gives the reasonable clustering result.

Table V shows the clustering results of example 4 by using approaches in reference [7] and [8].

TABLE V.
THE CLUSTERS OF EXAMPLE 4 USING APPROACH IN REFERENCE [7] AND [8]

Approach	Cluster
[7] ISB= $\{e_1,e_2,e_3,e_4\}$	$C_1=\{e_5,e_6,e_1,e_8\}$, $C_2=\{e_4,e_7\}$, $C_3=\{e_2,e_9\}$, $C_4=\{e_3\}$
[7] ISB= $\{e_1,e_5,e_3,e_4\}$	$C_1=\{e_5,e_6\}$, $C_2=\{e_4,e_7\}$, $C_3=\{e_2,e_9,e_1,e_8\}$, $C_4=\{e_3\}$
[8] $\rho=0.71$	$C_1=\{e_5,e_6\}$, $C_2=\{e_4,e_7\}$, $C_3=\{e_2,e_9\}$, $C_4=\{e_1,e_8\}$ $C_5=\{e_3\}$
[8] $\rho=0.82$	$C_1=\{e_4,e_7\}$, $C_2=\{e_5,e_6\}$ $C_3=\{e_1,e_8,e_2,e_9,e_3\}$

For approach in reference [7], when ISB= $\{e_1,e_2,e_3,e_4\}$, we get the wrong clustering result. While when ISB= $\{e_1,e_5,e_3,e_4\}$, the right clustering is obtained.

For approach in reference [8], when $\rho=0.71$, we get five clusters in which $C_3=\{e_2,e_9\}$ and $C_4=\{e_1,e_8\}$. The result is not right. Beliefs in these two clusters have intersection on true hypothesis, so they concern the same event and should merge into one cluster. When $\rho=0.82$, beliefs e_1, e_2, e_8 , e_9 and e_3 are in the same cluster. The result is obviously wrong, which reflects that the approach in reference [8] cannot cope with belief functions with hidden conflict.

C. Normal belief functions

1) Experiment data

Example 5 Let $\Omega=\{\omega_1,\omega_2,\cdots,\omega_6\}$ be a frame of discernment. Suppose eight distinct belief functions, defined on Ω, are given by

$$e_1: m(\{\omega_1,\omega_2\})=0.8, m(\{\omega_3\})=0.1, m(\{\omega_6\})=0.1\ ;$$

$$e_2: m(\{\omega_2,\omega_3\})=0.85, m(\{\omega_1\})=0.15\ ;$$

$$e_3: m(\{\omega_1,\omega_3\})=0.6, m(\{\omega_4\})=0.2, m(\{\omega_5\})=0.2\ ;$$

$$e_4: m(\{\omega_4\})=0.7, m(\{\omega_2,\omega_5\})=0.2, m(\{\omega_3,\omega_6\})=0.1;$$

$$e_5: m(\{\omega_4,\omega_5\})=0.8, m(\{\omega_1\})=0.2\ ;$$

$$e_6: m(\{\omega_4\})=0.8, m(\{\omega_1,\omega_3\})=0.2\ ;$$

$$e_7: m(\{\omega_6\})=0.6, m(\{\omega_1,\omega_3\})=0.2, m(\Omega)=0.2\ ;$$

$$e_8: m(\{\omega_3\})=0.8, m(\{\omega_1\})=0.2\ .$$

Beliefs e_4, e_5 and e_6 have their intersection ω_4 as the true hypothesis, so they can be divided into the same cluster. Both beliefs e_1 and e_2 support ω_2 as possible true hypothesis, so they may be related to the same events. And so does for beliefs e_3 and e_8. Different from other beliefs, belief e_7 strongly supports ω_6 as true hypothesis, so we may consider e_7 describes a different event.

2) Result and discussion

Fig. 4 presents the hierarchical tree of example 5.

From Fig. 4 we can see that the appropriate number of clusters is 4. The clusters and the corresponding leading elements are given by Table VI.

The clustering result listed in Table VI agrees with the analyses.

Table VII shows the clustering results of example 5 by using approaches in reference [7] and [8].

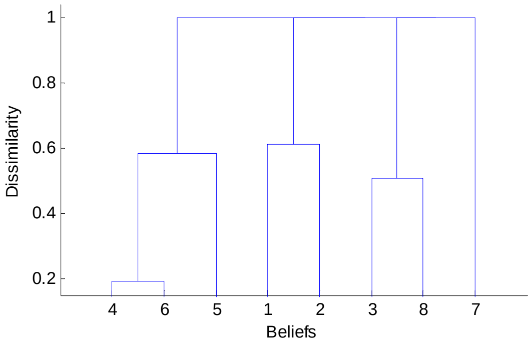

Figure 4. Hierarchical tree of example 5

TABLE VI.
THE CLUSTERS AND THE CORRESPONDING LEADING ELEMENTS OF EXAMPLE 5 USING THE EXTENDED AGGLOMERATIVE ALGORITHM

Cluster	Leading Element
$C_1=\{e_4,e_5,e_6\}$	$\{\omega_4\}$
$C_2=\{e_1,e_2\}$	$\{\omega_2\}$
$C_3=\{e_3,e_8\}$	$\{\omega_3\}$
$C_4=\{e_7\}$	$\{\omega_6\}$

TABLE VII.
THE CLUSTERS OF EXAMPLE 5 USING APPROACH IN REFERENCE [7] AND [8]

Approach	Cluster
[7] ISB= $\{e_1,e_2,e_3,e_4\}$	$C_1=\{e_4,e_5,e_6\}$, $C_2=\{e_1\}$, $C_3=\{e_2\}$, $C_4=\{e_3,e_7,e_8\}$
[7] ISB= $\{e_4,e_1,e_3,e_7\}$	$C_1=\{e_4,e_5,e_6\}$, $C_2=\{e_1,e_2\}$, $C_3=\{e_3,e_8\}$, $C_4=\{e_7\}$
[8] $\rho=0.55$	$C_1=\{e_3,e_4,e_5,e_6\}$, $C_2=\{e_1\}$, $C_3=\{e_2\}$, $C_4=\{e_7\}$, $C_5=\{e_8\}$
[8] $\rho=0.6$	$C_1=\{e_3,e_4,e_5,e_6\}$, $C_2=\{e_1\}$, $C_3=\{e_2,e_8\}$, $C_4=\{e_7\}$

For approach in reference [7], when ISB= $\{e_1,e_2,e_3,e_4\}$, belief e_7 is in the same cluster with beliefs e_3 and e_8. Belief e_7 strongly concerns hypotheses ω_6, whilst beliefs e_3 and e_8 have their intersection strongly concerns hypothesis ω_3. The two hypotheses are incompatible, so the clustering is still not right. When ISB= $\{e_4,e_1,e_3,e_7\}$, the right clustering is gotten.

For approach in reference [8], when $\rho=0.55$ and $\rho=0.6$, belief e_3 is always partitioned into the same cluster with beliefs e_4, e_5 and e_6. Belief e_3 strongly concerns hypotheses ω_1 and ω_3, whilst beliefs e_4, e_5

and e_6 have their intersection strongly concerns hypothesis ω_4. The hypotheses that they strongly concern are incompatible, so the clustering is still not right.

In all, above three examples demonstrate the superiority of our extended agglomerative algorithm in below three aspects:

(1) The extended agglomerative algorithm can determine the number of clusters by itself. It neither needs to set the number of clusters in advance nor needs to set any threshold value to control it.

(2) The extended agglomerative algorithm can avoid hidden conflict.

(3) The extended agglomerative algorithm is robust. The clustering result is regardless of the sequence of beliefs or any external control values.

VI. Conclusion

Clustering is necessary when belief functions related to different events are mixed up. However, most of clustering approaches for belief functions need to set the number of clusters in advance. Still, they may be unstable and cannot deal with the belief functions with hidden conflict. In order to solve these problems, an extended agglomerative algorithm is proposed for clustering all types of belief functions that are mixed up. Three typical examples product promising results which indicates the extended agglomerative algorithm is robust and can avoid the hidden conflict. What's more, the number of clusters is given out by algorithm itself instead of being set in advance. The extended agglomerative algorithm belongs to direct clustering approach, so it avoids the possible inequality transformation, and it is still simple.

References

[1] A. P. Dempster, "Upper and lower probabilities induced by a multi-valued mapping," Annals of Mathematical Statistics, vol. 38, pp. 325-339, 1967.

[2] G. Shafer, "A Mathematical Theory of Evidence," Princeton: Princeton University Press, 1976.

[3] J. MacQueen, "Some methods for classification and analysis of multivariate observations," In: Proc. of the Fifth Berkeley Symposium on Math, Stat. and Prob. vol. 1, 281-296, 1967.

[4] Guha,S., R. Rastogi, K. Shim, "CURE: An Efficient Clustering Algorithm for Large Databases," In Proc. Of ACM SIGMOD Intl. Conf. on Management of Data, pp. 73-82, 1998.

[5] Karypis, G., E. Han, and V. Kumar, "Chameleon: A hierarchical clustering algorithm using dynamic modeling," IEEE Computer, vol. 32, no. 8, pp. 68-75, 1999.

[6] J. Schubert, "Clustering decomposed belief functions using generalized weights of conflict," International Journal of Approximate Reasoning, vol. 48, no. 2, pp. 466-480, 2008.

[7] S. B. Hariz, Z. Elouedi, K. Mellouli, "Clustering approach using belief function theory," Lecture Notes in Computer Science, vol. 4183, no. 1, pp. 162-171, 2006.

[8] Ye Qing, Wu Xiaoping, Chen Zemao, "An approach for evidence clustering using generalized distance," Journal of electronics (china), 2009, 26(1): 18-23.

[9] Z. Huang, "Extensions to the k_means algorithm for clustering large data sets with categorical values," Data Mining Knowledge Discovery, vol. 2, no. 2, pp. 283-304, 1998.

[10] M. Daniel, "Associatively in combination of belief functions: a derivation of minC combination," Soft Computing, vol. 7, no. 5, pp. 288-296, 2003.

[11] Ph. Smets, R. Kennes, "The transferable belief model," Artificial Intelligence, vol. 66, no. 3, pp. 191-234, 1994.

[12] J. Dezert, F. Smarandache, "A new probabilistic transformation of belief mass assignment," 11th International Conference on Information Fusion, pp. 1-8, 2008.

[13] A. L. Jousselme, Dominic G, Bosse E, "A new distance between two bodies of evidence," Information Fusion, vol. 2, no. 2, pp. 91-111, 2001.

[14] R. Sibson, "SLINK: An optimally efficient algorithm for the single link cluster method," The Computer Journal, vol. 16, no. 1, pp. 30-34, 1973.

[15] D. Defays, "An efficient algorithm for a complete link method," The Computer Journal, vol. 20, no. 4, pp. 364-366, 1977.

[16] E. M. Voorhees, "Implementing agglomerative hierarchical clustering algorithms for use in document retrieval," Information Process Manage, vol. 22, no. 6, pp. 465-476, 1986.

Ying Peng was born in Huanggang, Hubei province, China, in 1982. She obtained her bachelor degree of engineering and Master degree of engineering in 2005 and 2008, respectively, from Academy of Equipment Command & Technology, Beijing, China.

She is now a Ph.D. candidate in Academy of Equipment Command & Technology. Her research interests include evidence theory, fault diagnosis, etc.

Huairong Shen was born in Shucheng, Anhui province, China, in 1954. He received Ph.D. degree of engineering in 1985 from National University of Defense Technology, Changsha, Hunan, China.

He is currently a professor and doctor supervisor of Aeronautics in Academy of Equipment Command & Technology. His research interests include fault diagnosis, Unmanned Aerial Vehicle technique, Avionic Material, etc.

Zenghui Hu was born in Pingxiang, Jiangxi province, in 1982. He obtained his bachelor degree of science and Master Degree of science in 2004 and 2006, respectively, from National University of Defense Technology, Changsha, Hunan, China.

He is now a Ph.D. candidate in the College of Electrical Science and Engineering, National University of Defense Technology. His research interests include blind source separation, array signal processing, evidence theory etc.

Yongyi Ma was born in Shucheng, Anhui province, in 1987. He obtained his bachelor degree of engineering in 2008 from Nanjing University of Aeronautics and Astronautics, Nanjing, China. He earned his Master degree of engineering in 2011 from Academy of Equipment Command & Technology, Beijing, China.

He is now an engineer. His research interests include fault diagnosis, evidence theory, etc.