IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received May 25, 2020, accepted June 16, 2020, date of publication June 26, 2020, date of current version July 8, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3005195

A Manufacturing Network Modeling and Evolution Characterizing Approach for Self-Organization Among Distributed MSMEs Under Social Manufacturing Context

JIAJUN LIU AND PINGYU JIANG
State Key Laboratory for Manufacturing Systems Engineering, Xi'an Jiaotong University, Xi'an 710049, China

Corresponding author: Pingyu Jiang (pjiang@mail.xjtu.edu.cn)

This work was supported in part by the Innovation Method Fund of China under Grant 2016IM010100, and in part by the National Natural Science Foundation of China under Grant 71571142.

ABSTRACT In the manufacturing industry, cross-enterprise resource sharing has emerged among micro-and-small-scale manufacturing enterprises (called social manufacturing nodes, SMNs) with similar manufacturing resources. In this context, social manufacturing is proposed to promote resource sharing among SMNs through order sharing in manufacturing communities (MCs) on the network platform. In social manufacturing, SMNs are geographically distributed and peer-to-peer, and MCs are formed by the self-organization among SMNs. However, because of the distributed and peer-to-peer characteristic of SMNs, the efficiency of SMNs self-organizing into MCs is relatively low, and the scope of self-organization is also relatively narrow. After MCs are formed, the structure of MCs is time-varying, and the evolution information of MCs is helpful for the smooth operation of the network platform. For these problems, this paper proposes a manufacturing network modeling and evolution characterizing approach. Firstly, distributed SMNs are clustered into overlapping MCs by the speaker-listener label propagation algorithm. Based on the clustering result, SMNs are recommended to each other as potential partners, by which they can quickly self-organize into MCs. On the other hand, seven fundamental events are defined to characterize the evolution of MCs on the network platform. From the evolution of MCs, the manager of the network platform get useful information for the smooth operation of the network platform. The feasibility of the proposed approach is verified by a simulation case.

INDEX TERMS Micro-and-small-scale manufacturing enterprises, social manufacturing, self-organization, overlapping manufacturing community, evolution.

I. INTRODUCTION

The application of information and communication technology in the manufacturing industry has expanded the resources sharing of manufacturing enterprises from internal to cross-enterprise [1]. Along with this process, many manufacturing models have also been proposed to solve the problem of resource sharing under different market demands, including virtual enterprise [2], agile manufacturing [3], service-oriented manufacturing [4], cloud manufacturing [5], etc. These manufacturing models have one thing in common, that is, they all integrate manufacturing resources through network platforms, such as virtual enterprise platforms [6], cloud manufacturing service platforms [7]. These platforms realize the cross-enterprise sharing of manufacturing resources to a certain extent, which improve the resource utilization and flexibility of manufacturing enterprises. However, the focus of these platforms is to achieve resource sharing among enterprises with different manufacturing resources through outsourcing, crowdsourcing, or crowdfunding.

Recently, cross-enterprise resource sharing among enterprises with similar manufacturing resources has emerged in the manufacturing industry. Micro-and-small-scale manufacturing enterprises (MSMEs) form physical groups to share their similar manufacturing resources through order sharing [8]. In this way, MSMEs can quickly respond to

The associate editor coordinating the review of this manuscript and approving it for publication was Hocine Cherifi.

service-oriented, personalized, diverse and dynamic market demands. This trend has emerged in many field of the manufacturing industry, such as, apparel industry [9], [10], 3D printing [11], and the emerging shared factory [12]. In this context, a new service-oriented advanced manufacturing paradigm called social manufacturing is proposed [13]–[15]. Its purpose is to integrate socialized manufacturing resources from distributed micro-and-small-scale manufacturing enterprises (called social manufacturing nodes, SMNs) and realize the cross-enterprise resource sharing among them by physical groups (called manufacturing communities, MCs) [15], [16]. Within an MC, SMNs share their similar manufacturing resources by order sharing. Between different MCs, SMNs share different types of manufacturing resources through outsourcing, crowdsourcing, etc. Through this cross-enterprise manufacturing resource sharing, the resource utilization and flexibility of SMNs are further improved. Like the aforementioned manufacturing model, the cross-enterprise resource sharing in social manufacturing is also implemented through a network platform [17].

In social manufacturing, the practitioners are the amount of geographically distributed SMNs and they self-organize into manufacturing communities (MCs) [8]. However, SMNs are geographically distributed all over the world so that they do not know each other well. On the other hand, SMNs are usually in a peer-to-peer relationship, which means there are no core-enterprises to act as organizers of MCs [17]. All these characteristics lead to that the efficiency of SMNs self-organization to form MCs is relatively low on the network platform, and the scope of self-organization is also relatively narrow. So, in order to help SMNs quickly identify potential partners to form MCs, it is necessary to recommend MCs for SMNs. After that, SMNs can quickly self-organize into real MCs. In addition, in the actual operation of social manufacturing, SMNs will participate in different MCs at different times so that the structure of MCs changes dynamically with time. Characterizing the evaluation of MCs contributes to the smooth operation of the network platform in social manufacturing. However, in the manufacturing industry, there is little research on the evaluation of MCs.

For the above problems, this study focused on clustering SMNs into overlapping MCs and characterizing the evolution of MCs. Firstly, several definitions of social manufacturing are given, at the same time, the time-varying characteristic of MCs is clarified and MCs are further divided into transient manufacturing communities (TMCs) and dynamic manufacturing communities (DMCs). Then, a domain ontology is constructed to describe the manufacturing interests (MIs) of SMNs. Based on this, SMNs select the MI features they are interested in to form their MI entities. Then, a feature-based approach is adopted to calculate MI similarity between SMNs according to their MI entities, and a threshold is given to judge whether there are MIRs between SMNs. After that, SMNs are clustered into overlapping TMCs by the speaker-listener label propagation algorithm (SLPA). Furthermore, DMCs are formally described, and seven fundamental events are defined to characterize the evolution of DMCs. The purpose of the above research is to improve the efficiency of SMNs self-organizing into MCs on network platform, and at the same time, provide guidance for the operation of network platform by characterizing the evaluation of MCs.

The rest of this paper is arranged as follows. After a brief review of concept subsumption based cross-ontology semantic similarity calculation, label propagation-based overlapping communities detecting in the static network and community evolution characterizing in the dynamic network in Section 2. In Section 3, the formation of overlapping TMCs is elaborated, and the evolution of DMCs is characterized. Section 4 shows a demonstrative case study. Discussion and conclusion are presented in Section 5 and Section 6, respectively.

II. LITERATURE REVIEW

A. CONCEPT SUBSUMPTION BASED CROSS-ONTOLOGY SEMANTIC SIMILARITY CALCULATION

Most of the existing research calculated the ontology-based semantic similarity according to ontological structure, concept subsumption, information content, or with a hybrid approach [18], [19]. For examples, Al-Mubaid H. *et al.* used the minimum path length and the taxonomical depth in ontological structure to calculate the semantic similarity between concepts [20]. Pirró G. *et al.* calculated the semantic similarity between words by concept subsumption and information content [21]. Further, Gao J. B. *et al.* combined the weighted shortest path length and information content to evaluate the semantic similarity between concepts [22]. In the above methods, concept subsumption is the most suitable for cross-ontology semantic similarity calculation, and it was first implemented by Rodríguez and Egenhofer [23], who computed semantic similarity between independent ontologies by the weighted sum of words matching, feature matching and semantic-neighborhood matching. Based on this work, Petrakis *et al.* [24] proposed an X-Similarity method to calculate cross-ontology semantic similarity between ontology concepts based on the matching between synsets and term description sets. Afterward, Sánchez *et al.* [19] proposed a feature-based approach to assess normalized dissimilarity between ontology concepts based on their taxonomical features, and this approach could also be used in cross-ontology. But the premise to apply the above methods was that all ontologies must be predefined complete with no features or terms missing. To enable the similarity estimation across multiple ontologies when there is a term missing on some ontology, Batet *et al.* [25] proposed several heuristics based on three cases by comparing concept pairs on ontologies. Further, in order to consider implicit evidence, Solé-Ribalta *et al.* [26] combined concept subsumption and ontological structure between ontologies to realize multi-ontology similarity assessment. In this paper, because the feature-based approach [19] can obtain normalized semantic similarity and it is effective and convenient, we adopt this approach to calculate MI similarity between MI ontologies of SMNs.

B. LABEL PROPAGATION-BASED OVERLAPPING COMMUNITY DETECTING IN STATIC NETWORK

Palla *et al.* [27] first introduced the clique percolation method (CPM) to detect overlapping communities. And then, many approaches were proposed for overlapping community detection, including LFM [28], SLPA [29], CDAEO [30], OSLOM [31], Game [32] etc. According to the detection style, Xie *et al.* [33] divided into these approaches into five categories, i.e., clique percolation, line graph and link partitioning, local expansion and optimization, fuzzy detection, and agent-based and dynamical algorithms. Among these approaches, the speaker-listener label propagation algorithm (SLPA) in link partitioning has better performance in detecting overlapping communities, which is an extension of the label propagation algorithm (LPA) [34] and community overlap propagation algorithm (COPRA) [35]. LPA is first proposed by Raghavan *et al.* [34] to fast detect community based on network structure, but it wasn't suitable for overlapping community detecting. Then, Leung *et al.* [36] examined the performance of the label propagation algorithm by comparing asynchronous updating with synchronous updating, and found that synchronous updating was more stable in community detecting. In order to improve the quality of communities detected by the label propagation algorithm, Szymanski [37] introduced new update and label propagation rules for label propagation algorithm. Gregory [35] enabled the label propagation algorithm to find overlapping communities by extending the label and propagation step, whose key was that each vertex could belong to up to a given number of communities. Following the idea in [34, 35], Xie *et al.* [29] proposed the speaker-listener label propagation algorithm (SLPA) by introducing listener rule and speaker rule into LPA, which enabled each vertex to accumulate label knowledge so that the result of overlapping community detecting was more stable. Furthermore, Xie and Szymanski [38] proposed the LabelRank algorithm where four operators, i.e. propagation, inflation, cutoff, conditional update, were introduced into label propagation algorithm to control and stabilize its propagation process. Inspired by the confidence of human communication, Dai *et al.* [39] presented a multi-label propagation algorithm (MLPA) to detect overlapping communities, whose propagation process was guided by propagating intensity. In this study, because SLPA has an attractive potential to fast detect overlapping communities, it is introduced to cluster SMNs into overlapping MCs based on MIRs between SMNs.

C. COMMUNITY EVOLUTION CHARACTERIZING IN DYNAMIC NETWORK

Palla *et al.* [40] firstly defined six fundamental events to characterize the dynamic change of a community in structure, and they proposed two basic quantities (the size and the age of a community) to quantify and characterize the community evolution. Lin *et al.* [41] proposed the FacetNet framework to characterize evolution in a unified process by Bayesian theory, where a Dirichlet distribution-based probabilistic model was used to capture community evolution. Asur *et al.* [42] introduced five fundamental events to characterize the evolutionary behavior of communities and four basic transformations to characterize the behavioral patterns of individuals over time, and then, a series of indexes were proposed to characterize the behavior of communities and individuals. Greene *et al.* [43] defined six fundamental events to track the evolution of dynamic communities over time, which was motivated by using a heuristic threshold-based method to realize many-to-many matching between communities at different time steps. Takaffoli *et al.* [44] modeled community evolution in social networks based on five fundamental events and defined two basic quantities (lifetime and member fluctuation) to characterize a community. Saganowski *et al.* [45] defined seven fundamental events, and adopted a group evolution discovery (GED) method to track community evolution. Oliveira *et al.* [46] introduced two time window models to characterize the evolution of customer communities: one is the landmark window model for characterizing the long-term evolution of communities over the entire period, and the other is sliding window model for characterizing current evolution of communities at fixed time span. Diakidis *et al.* [47] applied the GED method to track community evolution, at the same time, extracted three community features to predict the continuation, shrinking, growth and dissolution of community. From the above literature, it can be seen that the evolution of communities in a dynamic network is characterized by defining fundamental events. SMNs will join different MCs at different times, which makes MCs overlap with each other during evolution. Therefore, the heuristic threshold-based method is adopted to realize many-to-many matching between TMCs and DMCs.

III. METHODOLOGY

A. PRELIMINARIES

1) DEFINITIONS

This section elaborates six definitions related to this paper, including social manufacturing (SMNs), manufacturing interest (MI), manufacturing interest relationship (MIR), manufacturing community (MC), and social manufacturing network (SMNet). In addition, the relationships among the above six definitions is also clarified.

Definition 1: Social manufacturing nodes (SMNs) mean socialized micro-and-small-scale manufacturing enterprises that are geographically distributed and own diverse manufacturing resources. Those SMNs are capable to provide specialized manufacturing services for customers/prosumers, and are willing to share their orders with other SMNs through the network platform.

Definition 2: Manufacturing interest (MI) means the preference of an SMN in providing manufacturing services during a period. MI of an SMN depends on which socialized manufacturing resources the SMN owns. And it is also closely related to how the SMN to manage its socialized

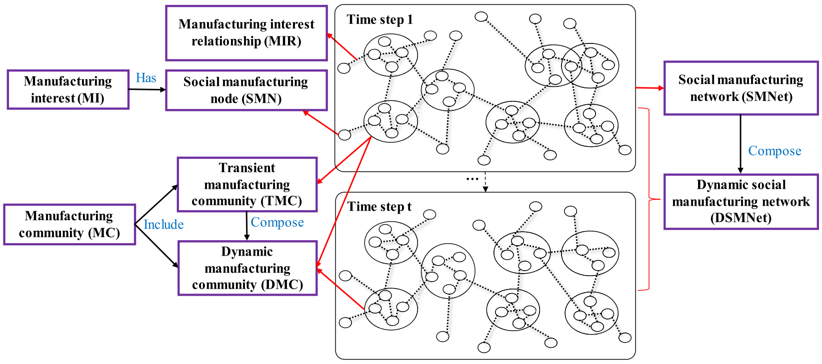

FIGURE 1. The relationships among the above six definitions.

manufacturing resources. MI has time-varying characteristic, whose reasons include three aspects: market demands change over time, socialized manufacturing resources of SMN transform with time, and SMN subjectively change its MIs at different times.

Definition 3: Manufacturing interest relationship (MIR) is defined as a kind of social relationship between two SMNs. and it depends on the MI similarity between two SMNs. This paper judges whether there is a relationship bewteen two SMNs by a given threshold $\theta \in [0, 1]$. If MI similarity between two SMNs exceeds the threshold $\theta \in [0, 1]$, we consider there is an MIR between them, because the greater MI similarity between two SMNs, the more likely they are to form a MC. And MIRs between SMNs are dynamic since MIs of SMNs change over time.

Definition 4: Manufacturing community (MC) is defined as a physical group composed of SMNs with common MIs. MC is formed by self-organization between SMNs. In essence, the self-organization is a spontaneous process where SMNs pursuit more business interests. Therefore, MC can be regarded as a relatively stable community of interest where SMNs share their socialized manufacturing resources by sharing their product orders. SMNs in MCs autonomously coordinates socialized manufacturing resources to complete product orders, such as distributing orders according to the real-time manufacturing capacities of SMNs. During the operation of an MC, SMNs will join or leave the MC in different periods because of the time-varying characteristic of MIs, which leads to the MC structure changing over time, including its members and common MIs. For analyzing the evolution of MCs, MCs are further divided into transient MCs (TMCs) and dynamic MCs (DMCs). TMCs refer to a snapshot of MCs at one time step and they are composed of SMNs and their MIRs at the time step. TMCs reflect the static structure of MCs at one time step. DMCs cover the whole lifecycle of MCs and are composed of TMCs at successive time steps. DMCs reflect the evolution of MCs during their lifecycle.

Definition 5: Social manufacturing network (SMNet) is an MI-based transient network which is a snapshot of SMNs, TMCs and MIRs between SMNs at one time step. From the view of complex network theory, SMNs are abstracted as the nodes of SMNet, and MIRs between SMNs are considered as the edges. SMNet can be modeled as $SMNet^t = \{SMN, E\}$, where $SMN = \{SMN_1, SMN_2, \ldots, SMN_n\}$ represents the set of SMNs, $E = \{e_1, e_2, \ldots, e_m\}$ represents the set of MIRs between SMNs, t is the time step corresponding to $SMNet^t$. TMCs are the clusters in $SMNet^t$, and they can be formulated as a set $TMC^t = \left\{TMC_1^t, TMC_2^t, \ldots, TMC_k^t\right\}$, where k is the number of TMCs in $SMNet^t$ at the time step t, $TMC_k^t = \left\{SMN_1, SMN_2, \ldots, SMN_p\right\}$ denotes a TMC and p is the number of SMNs contained in TMC_k^t.

Definition 6: Dynamic social manufacturing network (DSMNet) is the set of SMNet at successive time steps. In DSMNet, SMNs, MIs of SMNs and MIRs between SMNs are all time-varying. DSMNet can be formally described as $DSMNet = \left\{SMNet^1, SMNet^2, \ldots, SMNet^t\right\}$, where $SMNet^t$ means the network snapshot at time step t. FIGURE 1 presents the relationships among the above definitions. Social manufacturing nodes (SMNs) have diverse manufacturing interests (MIs) in provide manufacturing services, and there are a mass of underlying manufacturing interest relationships (MIRs) between SMNs. Based on this, SMNs with similar MIs self-organize into overlapping manufacturing communities (MCs) to share their socialized manufacturing resources. Considering the time-varying characteristic of MIs, MCs are divided into transient MC (TMC) and dynamic MC (DMC). SMNs, TMCs and MIRs between SMNs constitute social manufacturing network (SMNet). And the set of SMNet at successive time steps constitutes dynamic social manufacturing network (DSMNet).

2) RESEARCH IDEA

The purpose of this paper is to cluster distributed SMNs into MCs, and then characterize the evolution of MCs. The research idea of this paper is presented in FIGURE 1. Based on the above six definitions, firstly, a domain ontology is used to describe the MIs of SMNs. Then, the MI similarity between SMNs is calculated through a feature-based approach, based on that, a manufacturing interest relationship network is built by a given threshold $\theta \in [0, 1]$. Afterward, SMNs are clustered into overlapping MCs based on the speaker-listener

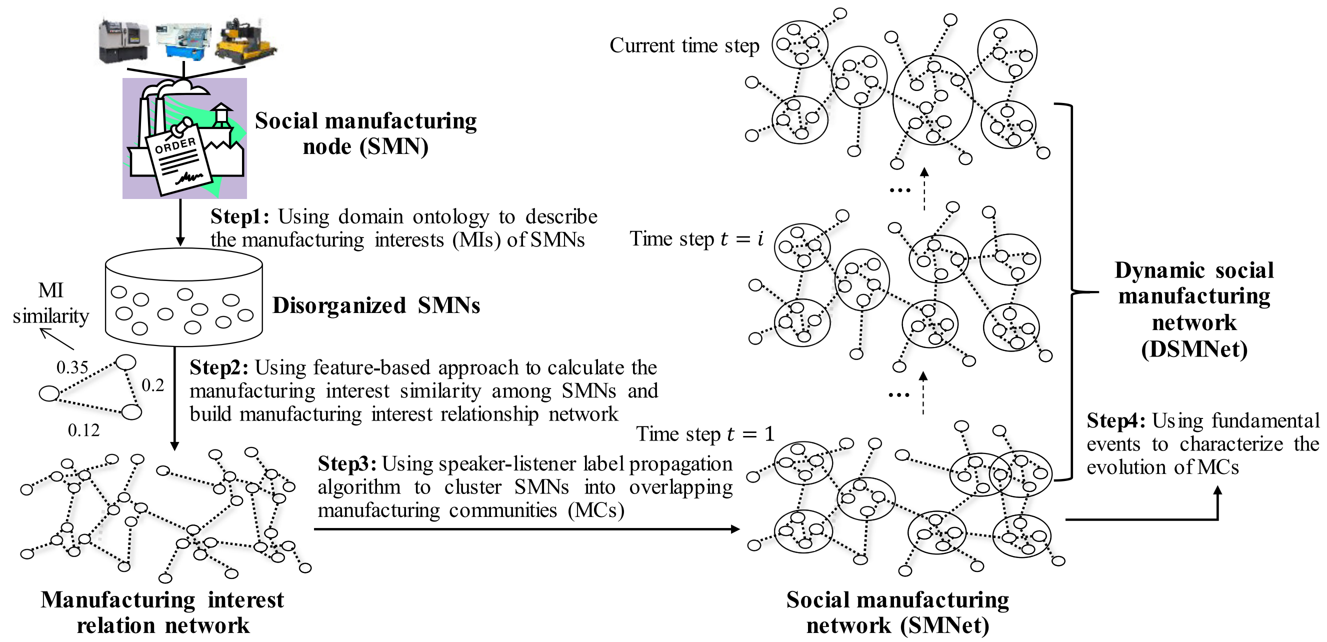

FIGURE 2. The research idea of this paper.

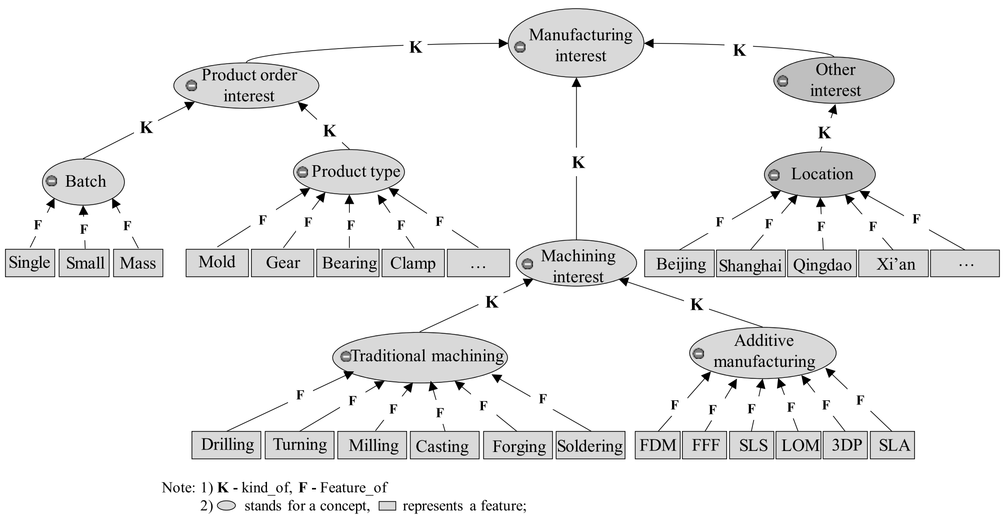

FIGURE 3. MI domain ontology.

label propagation algorithm (SLPA). According to the clustering result, SMNs in the same MCs will be recommended to each other as potential partners, by which SMNs can quickly self-organize into diverse MCs. During the operation of the network platform in social manufacturing, MCs are time-varying and overlap with each other. Finally, in order to characterize the evolution of MCs, seven fundamental events are defined.

B. FORMING TRANSIENT MANUFACTURING COMMUNITY

1) MI DOMAIN ONTOLOGY

The MIs of SMNs are described as a domain ontology, and it is composed of a series of concepts, features and their relationships, as shown in FIGURE 3.

MI domain ontology includes three rough concepts, i.e. product order interest, machining interest and other interest. The three rough concepts are further divided into refined concepts, for example product order interest is divided into batch and product type. Every refined concept can be described by one or more MI features, such as the refined concept "batch" corresponds to three MI features: single, small and mass.

When SMNs participate in the network platform, SMNs select the MI features in MI domain ontology to form their MI entities. The MI entity of SMN_i can be formally described as

$$MI(SMN_i) = \{single, small, \ldots, turning, Beijing\} \quad (1)$$

TABLE 1. The pseudo-code of SLPA-based overlapping TMCs clustering.

Algorithm 1. the pseudo-code of SLPA-based overlapping TMCs clustering
Step 1. The memory of each SMN is initialized with a unique label, the SMN's id.
Step 2. SMNs are selected as listeners at propagation step p in turn.
Step 3. Each neighbor of the selected SMN is considered as a speaker and sends out a single label from its memory at propagation step p -1 based on speaking rule.
Step 4. The selected SMN accepts one label from the sent label set according to the listening rule.
Step 5. Repeat Step 2 to Step 4 until the termination condition is achieved.
Step 6. Execute post-processing to output overlapping TMCs.

where $MI(SMN_i)$ means the MI entity of SMN_i, sin gle, $small$, $\ldots$, $turning$, $Beijing$ are the MI features of MI entity $MI(SMN_i)$.

2) NORMALIZED MI SIMILARITY CALCULATION

Based on the MI entities of SMNs, the feature-based approach [19] is used to calculate MI similarity between SMNs. MI similarity between SMN_1 and SMN_2 is calculated as (2) and (3), as shown at the bottom of the page, where $Dis(SMN_1, SMN_2)$ denotes normalized MI dissimilarity between SMN_1 and SMN_2, $|MI(SMN_1)\backslash MI(SMN_2)|$ means the number of MI features in $MI(SMN_1)$ but not in $MI(SMN_2)$, $|MI(SMN_1)/MI(SMN_2)|$ means the number of MI features in $MI(SMN_2)$ but not in $MI(SMN_1)$, $|MI(SMN_1) \cap MI(SMN_2)|$ means the number of MI features belonging to both $MI(SMN_1)$ and $MI(SMN_2)$, $Sim(SMN_1, SMN_2)$ is the normalized MI similarity between $MI(SMN_1)$ and $MI(SMN_2)$.

3) CLUSTERING SMNs INTO OVERLAPPING TMCs

After MI similarity between SMNs is obtained, by the given threshold $\theta \in [0, 1]$, MI relationships between SMNs are obtained. Based on that, SMNs can be clustered into default TMCs according to their MIRs. During this process, the overlap among TMCs must take into account because SMNs can join different TMCs based on their diverse MIs. Therefore, the speaker-listener label propagation algorithm (SLPA) [29], [48] is adopted.

In the SLPA algorithm, each SMN is associated with one or more labels. A label corresponds to an SMN's id, and SMNs with the same label belong to the same TMC. An SMN can hold multiple labels in the propagation process of SLPA according to the underlying network structure, which means that an SMN can belong to many TMCs. The propagation process of SLPA is realized by mimicking human communication behavior. At the same time, in order to take into account old label knowledge observed in the past, each SMN has a memory to store the label set observed from the first time step to the current time step.

At each propagation step, SMNs are selected as listeners in turn. Each neighbor of the selected SMN is considered as a speaker. And it sends out a single label based on speaking rule: randomly sending a label from its memory with a probability that is proportional to the frequency of the label in its memory. After that, the selected SMN will accept one label from the sent label set by listening rule: selecting the most popular label from the sent label set. The algorithm terminates when a predefined maximum number of iterations T is reached. After the algorithm is terminated, post-processing is executed to output overlapping TMCs. In post-processing, a threshold $r \in [0, 1]$ is set to determine whether a label in the SMN memory is saved. If the probability of seeing a label is less than the threshold, the label will be removed from the SMN memory. When this process is completed, SMNs with the same label in their memories are divided into the same TMC. If there are multiple labels in the memory of an SMN, it is considered to belong to many TMCs. The pseudo-code of SLPA-based overlapping MCs clustering is depicted in Table 1.

C. CHARACTERIZING THE EVOLUTION OF DYNAMIC MANUFACTURING COMMUNITIES

1) CLUSTERING SMNs INTO OVERLAPPING TMCs

After SMNs are given some recommended TMCs, SMNs will quickly self-organize into real TMCs. In the operation of the network platform in social manufacturing, SMNs have different MIs at different time steps and will join different TMCs. At each time step, SMNs, TMCs and MIRs between SMNs constitute a SMNet. DSMNet is a set of SMNets at successive time intervals. DMCs are obtained by combining TMCs at different time steps in chronological order. A DMC can be regarded as a set of TMCs that have been detected from the initial time step of DSMNet to the current time step. In each DMC, there is a TMC in the set that appears at the most recent time step, and it is considered as the front of the DMC. And the front of a DMC is constantly updated with its evolution. DMCs can be formally described as

$$DMC = \{DMC_1, DMC_2, \ldots, DMC_h\} \tag{4}$$

$$Dis(SMN_1, SMN_2) = \log_2\left(1 + \frac{|MI(SMN_1)\backslash MI(SMN_2)| + |MI(SMN_1)/MI(SMN_2)|}{|MI(SMN_1)\backslash MI(SMN_2)| + |MI(SMN_1)/MI(SMN_2)| + |MI(SMN_1) \cap MI(SMN_2)|}\right) \tag{2}$$

$$Sim(SMN_1, SMN_2) = 1 - Dis(SMN_1, SMN_2) \tag{3}$$

$$DMC_h = \left\{TMC_h^e, \ldots, TMC_h^d\right\}, \ o \leq t \wedge d \leq t \wedge o \leq d \quad (5)$$
$$FMC = \{FMC_1, FMC_2, \ldots, FMC_h\} \quad (6)$$

where DMC is the set of DMCs in DSMNet, DMC_h refers to a DMC, h denotes the number of DMCs in DSMNet, e is the time step that DMC_h emerges, TMC_h^e is the TMC of DMC_h at time step e, d is the most recent time step that DMC_h is observed, TMC_h^d is the TMC of DMC_h at time step d, t is the current time step, FMC is the set of fronts of DMCs, FMC_h is the front of DMC_h. At the time step d, TMC_h^d is considered as the front FMC_h of DMC_h. By equation (4), equation (5) and equation (6), DMCs can be clearly presented.

2) FUNDAMENTAL EVENTS

The evolution of DMCs means the continuous changes of DMC structures over time, which is characterized by a series of fundamental events. The fundamental events are used to connect TMCs to DMCs, which reflects the evolutionary relationships between DMCs.

There is a broad consensus on fundamental events that characterize the evolution of dynamic communities. Palla *et al.* [40] considered the fundamental events in community evolution as growth, contraction, merging, splitting, birth and death. Asur *et al.* [42] defined five fundamental events for the evolution of the dynamic community, i.e. continue, k-merge, k-split, form and dissolve. Takaffoli *et al.* [44] regarded form, dissolve, survive, split and merge as fundamental events to characterize the evolution of dynamic communities. Greene *et al.* [43] adopted six fundamental events, including birth, death, merging, splitting, expansion and contraction, to track the evolution of dynamic communities. According to the above research, for the evolution of DMCs, we define seven fundamental events, i.e. birth, death, merging, splitting, expansion, contraction and continuation. The seven fundamental events are elaborated as follows:

Birth: Birth of a DMC means that the DMC emerges at the current time step, but it has not been identified at the previous time steps. It indicates that some SMNs with similar MIs self-organize into a new TMCs at the current time step.

Death: The death of a DMC means that it can't be detected any more in DSMNet at serval consecutive time steps. It indicates that SMNs in the DMC have changed their MIs so that they have low MI similarity, which drives SMNs out of the DMC.

Merging: When a TMC is observed to match two separate DMCs at the current time step, the two DMCs merge into one DMC. The merging of two DMCs means that SMNs in the two DMCs have changed their MIs towards a common direction, and they tend to provide similar manufacturing services at the current time step. The reason may be that SMNs can earn more business benefits by providing such manufacturing services.

Splitting: A DMC is considered to have a splitting when its front matches two TMCs at the current time step. The splitting of a DMC implies that SMNs in the DMC have been distracted from their common MIs, which has led SMNs to part ways and join different DMCs.

Expansion: The expansion of a DMC occurs when it contains more SMNs at the current time step than the most recent time step (growth $> 10\%$). The main motivation behind this is that more and more SMNs have changed or expanded their MIs to the common MIs of the DMC for more business benefits.

Contraction: If the number of SMNs in a DMC at the current time step is less than the most recent time step, the DMC is believed to have experienced a contraction (reduction $> 10\%$). It indicates that in the DMC, competition among SMNs has become increasingly fierce, which forces some SMNs to change their MIs. Another reason for the contraction may be that market demands for manufacturing services provided by the DMC are declining.

Continuation: The continuation of a DMC means that the change in the number of SMNs in the DMC is relatively stable (no more than 10%) from the most recent time step to the current time step. It implies that the DMC is running smoothly so that most SMNs in DMC have not changed their MIs.

3) CHARACTERIZING THE EVOLUTION OF DMCs

Based on the seven fundamental events, the evolution of DMCs can be characterized by continuously adopting them to connect TMCs to DMCs from the initial time step of DSMNet to the current time step. The key is to match TMCs at current time step with DMCs and map the seven fundamental events with the matching result.

To match TMCs with DMCs, a heuristic threshold-based method is applied [43]. This method judges whether there is a match between a TMC and the front of a DMC by calculating the similarity between them. The front of a DMC refers to the TMC that appears in DMC at the recent time step. If their similarity exceeds a threshold $\mu \in [0, 1]$, the TMC is considered to match the DMC. And the TMC will be added to the DMC, at the same time, it will be updated as the front of the DMC. The similarity between a TMC and the front of a DMC is calculated as:

$$sim(FMC_h, TMC_k^t) = \frac{\left|FMC_h \cap TMC_k^t\right|}{\left|FMC_h \cup TMC_k^t\right|} \quad (7)$$

where FMC_h is the front of DMC_h, TMC_k^t is a TMC identified at the current time step t, $sim(FMC_h, TMC_k^t)$ denotes the similarity between FMC_h and TMC_k^t, $\left|FMC_h \cap TMC_k^t\right|$ is the number of SMNs belonging to both FMC_h and TMC_k^t, $\left|FMC_h \cup TMC_k^t\right|$ is the number of SMNs that are the members of FMC_h, or TMC_k^t, or both. If $sim(FMC_h, TMC_k^t)$ is greater than the threshold μ, TMC_k^t will be added to FMC_h and updated as the front of FMC_h.Table 2 shows the pseudo-code of adopting the heuristic threshold-based method to match TMCs with DMCs.

After that, fundamental events can be mapped to the above matching results to characterize the evolution of DMCs. If a TMC is matched with no DMC, there is a "birth" of a new DMC. If a TMC is matched with two or more DMCs,

TABLE 2. The pseudo-code of adopting heuristic threshold-based method to match TMCs with DMCs.

Algorithm2. the pseudo-code of adopting the heuristic threshold-based method to match TMCs with DMCs
Step 1. Apply the SLPA algorithm to obtain TMC^1 from $SMNet^1$ at time step 1, and initialize DMC by creating a new DMC for each $TMC_i^1 \in TMC^1$.
Step 2. For each subsequent time step $t>1$, obtain TMC^t from $SMNet^t$.
Step 3. Process each $TMC_k^t \in TMC^t$ as follows:
1) Match all DMCs in DMC by equation (7), if $sim(TMC_k^t, FMC_h) > \mu$, there is a match between TMC_k^t and FMC_h.
2) If there is no match, create a new DMC containing TMC_k^t.
3) Otherwise, add TMC_k^t to all matching DMCs
Step 4. Update the set of fronts for all DMCs to be the latest matched TMC. And create a split DMC if one existing DMC has been matched 2 or more TMCs.
Step 5. Repeat Step2 to Step 4 until all SMNets at all time steps have been processed.

TABLE 3. Refined concepts and their corresponding MI features in MI domain ontology.

Refined concept	MI feature	Number
Batch	Single, Small, Mass	3
Product type	Accumulator, Bearing, Blade, Bolt nut, Chain, Clamp, Clutch, Crank set, Cutter, Gear, Motor, Mold, Pump, Piston group, Fan, Tyre, Valve	17
Traditional machining	Drilling, Turning, Milling, Casting, Forging, Soldering	6
Additive manufacturing	FDM, FFF, SLS, LOM, 3DP, SLA	6
Location	Beijing, Tianjin, Chongqing, Shanghai, Macao, Hongkong, Shijiazuang, Zhengzhou, Xi'an, Taiyuan, Jinan, Lanzhou, Shenyang, Changchun, Haerbin, Kunming, Guiyang, Fuzhou, Guangzhou, Haikou, Taibei, Chengdu, Wuhan, Changsha, Nanchang, Heifei, Nanjing, Hangzhou, Xining, Abroad	29

TABLE 4. The fragment of MI entities of SMNs.

SMN	MI entity
SMN1	[Mass', 'Clutch', 'Fan', 'Gear', 'Crank set', 'Blade', 'Turning', 'SLS', 'Guiyang', 'Xining', 'Zhengzhou', "Xi'an", 'Abroad', 'Macao', 'Guangzhou', 'Haerbin', 'Hangzhou', 'Chongqing']
SMN2	['Mass', 'Gear', 'Accumulator', 'Forging', 'Turning', 'Drilling', 'FFF', 'Fuzhou', 'Guiyang', 'Jinan', 'Shijiazuang', 'Beijing']
SMN3	['Mass', 'Motor', 'Valve', 'Casting', 'Drilling', 'SLA', 'Abroad', 'Beijing', 'Shanghai', 'Kunming', 'Changsha', 'Guiyang']
SMN4	['Single', 'Small', 'Mold', 'Pump', 'Chain', 'Blade', 'Forging', 'Soldering', 'Milling', 'FFF', 'SLS', 'Wuhan', 'Changsha', 'Beijing', 'Shanghai', 'Shijiazuang', 'Tianjin', 'Nanjing', 'Nanchang', "Xi'an"]
SMN5	['Mass', 'Valve', 'Tyre', 'Bearing', 'Motor', 'Blade', 'Casting', 'Drilling', 'FDM', 'LOM', 'Taiyuan', 'Chengdu', 'Xining', 'Tianjin']
…	…

there is a "merging" among DMCs. If a DMC is matched with no TMC at serval consecutive time steps, it indicates the "death" of the DMC. If a TMC is matched with only a DMC, there is an "expansion", a "contraction" or a "continuation" of the DMC, which can be judged by calculating changes in the number of SMNs in the DMC. If two or more TMCs are matched with only a DMC, there is a "splitting" in the DMC. Note that if DMCs only appear at one time step, the DMCs will be ignored and removed from the set of DMCs.

IV. CASE STUDY

In this section, a simulation case is run to verify the feasibility of the approaches proposed in this paper. The implementation process of the case is divided into five stages, i.e. constructing MI domain ontology, forming MI entities of SMNs, calculating MI similarity between SMNs, clustering SMNs into TMCs, and simulating the evolution of DMCs.

According to MI domain ontology in FIGURE 3, MI domain ontology of this case is shown in Table 3, including 3 rough concepts (i.e., product order interest, machining interest, and other interest), 5 refined concepts, and 61 MI features. Each refined concept contains multiple MI features.

Based on the MI domain ontology, SMNs can select the MI features that they are interested in to form their MI entities. We randomly produce 100 SMNs with 100 MI entities, and Table 4 lists the fragment of MI entities of SMNs.

After that, according to equation (2) and equation (3), the MI dissimilarity between SMN_1 and SMN_2 is calculated $Dis(SMN_1, SMN_2) = \log_2\left(1+\frac{14+8}{14+8+4}\right) = 0.885$, so their MI similarity is $Sim(SMN_1, SMN_2) = 1 - Dis(SMN_1, SMN_2) = 0.115$. Correspondingly, MI similarity between other SMNs is obtained in the same way. Based on MI similarity, we set a threshold $\theta = 0.24$ to determine MIRs between SMNs. After MIRs between SMNs are obtained, SMNs are clustered into overlapping TMCs through SLPA algorithm. In this case, the predefined maximum number of iterations is set as $T = 100$, and the threshold for

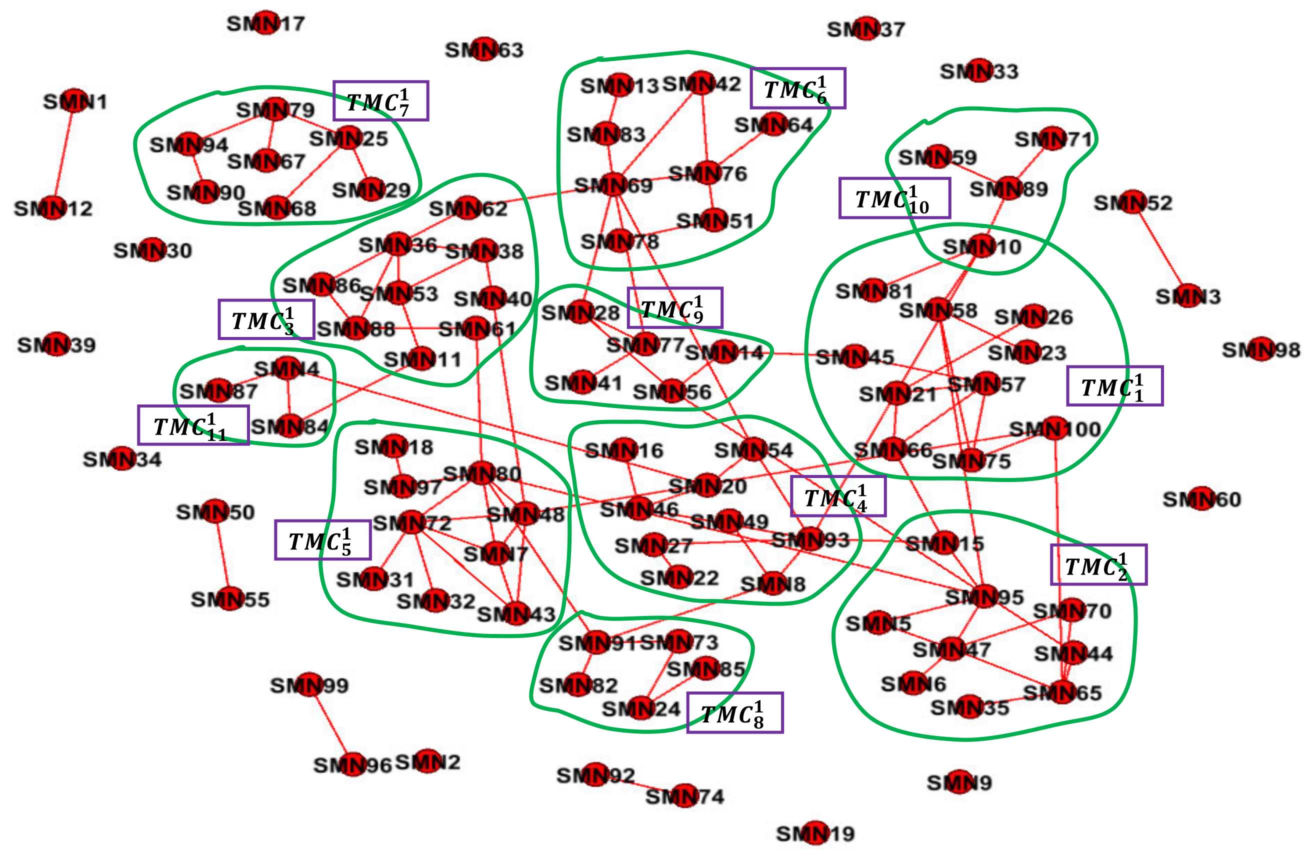

FIGURE 4. The final result of clustering SMNs into overlapping TMCs by SLPA algorithm.

TABLE 5. The member of TMCs and their common MI features at time step 1.

TMC	Member	Common MI feature
TMC_1^1	SMN10 SMN58 SMN81 SMN45 SMN66 SMN21 SMN26 SMN57 SMN23 SMN100 SMN75	SLA, Accumulator, Changsha
TMC_2^1	SMN5 SMN47 SMN95 SMN6 SMN15 SMN35 SMN65 SMN44 SMN70	Mass, Casting, SLA
TMC_3^1	SMN11 SMN53 SMN36 SMN38 SMN62 SMN86 SMN88 SMN40 SMN61	Single, Taiyuan, Bolt nut, Gear, Turning, Casting, Abroad
TMC_4^1	SMN20 SMN8 SMN49 SMN93 SMN16 SMN46 SMN54 SMN22 SMN27	Small, Cutter, FDM, Heifei
TMC_5^1	SMN7 SMN43 SMN48 SMN72 SMN80 SMN18 SMN97 SMN31 SMN32	Small, Soldering, SLA, Haikou
TMC_6^1	SMN13 SMN83 SMN69 SMN42 SMN76 SMN51 SMN78 SMN64	Single, Chengdu, Drilling, Milling, FFF, Shijiazuang
TMC_7^1	SMN25 SMN29 SMN68 SMN79 SMN67 SMN94 SMN90	Mass, Drilling, Turning, Hangzhou
TMC_8^1	SMN91 SMN24 SMN73 SMN85 SMN82	Small, Valve, Turning, Casting, Chongqing, Guangzhou, Abroad
TMC_9^1	SMN14 SMN56 SMN28 SMN77 SMN41	Shanghai, Gear, Milling
TMC_{10}^1	SMN10 SMN89 SMN59 SMN71	Mass, Chain, Guangzhou
TMC_{11}^1	SMN4 SMN84 SMN87	Single, Chain, Pump, FFF, Nanjing

post-processing is set as $r = 0.45$. In order to get better overlapping TMCs, we run SLPA algorithm 5 times and select the result with the most TMCs to be the output result. At the same time, we delete TMCs with less than three members from the output result to obtain the final result. The final result is shown in FIGURE 4. It can be clearly seen that 100 SMNs is clustered into 11 overlapping TMCs, and SMN 10 are clustered into both TMC_{10}^1 and TMC_1^1.

In order to uncover MI information in a TMC, firstly, we calculate the intersection of MI features between any

TABLE 6. TMCs and their common MI features at time step 2.

TMC	Member	Common MI feature
TMC_1^2	SMN5 SMN95 SMN10 SMN58 SMN81 SMN23 SMN47 SMN70 SMN100 SMN75	SLA, Mass, Changsha
TMC_2^2	SMN7 SMN76 SMN69 SMN42 SMN51 SMN78 SMN64 SMN83	Chengdu, Single, Drilling
TMC_3^2	SMN9 SMN94 SMN19 SMN25 SMN68 SMN79 SMN67 SMN90	Mass, Drilling, Hangzhou
TMC_4^2	SMN43 SMN72 SMN32 SMN48 SMN80 SMN61 SMN97	Single, Small, SLA
TMC_5^2	SMN31 SMN11 SMN53 SMN84 SMN18 SMN38 SMN40	SLS, Single, Casting, Nanjing
TMC_6^2	SMN93 SMN15 SMN22 SMN20 SMN46 SMN54 SMN27	Small, Heifei, Soldering, FDM, LOM
TMC_7^2	SMN13 SMN63 SMN73 SMN73 SMN91 SMN82	Small, Valve, Single, Macao
TMC_8^2	SMN17 SMN29 SMN41 SMN28 SMN56 SMN77	Turning, Shanghai, Mass, Heifei
TMC_9^2	SMN21 SMN26 SMN57 SMN66 SMN45	Accumulator, Zhengzhou, Haerbin
TMC_{10}^2	SMN16 SMN88 SMN36 SMN62 SMN86	Taiyuan, Single, Gear, Turning, Abroad
TMC_{11}^2	SMN12 SMN24 SMN85	Small, Turning, Beijing, Chongqing, Guangzhou, Xining
TMC_{12}^2	SMN89 SMN59 SMN71	Small, Mass, Chain, Guangzhou
TMC_{13}^2	SMN35 SMN65 SMN44	Mass, Milling, Nanchang

TABLE 7. TMCs and their common MI features at time step 3.

TMC	Member	Common MI feature
TMC_1^3	SMN5 SMN95 SMN10 SMN31 SMN58 SMN81 SMN43 SMN29 SMN21 SMN66 SMN36 SMN92 SMN72 SMN35 SMN37 SMN57 SMN48 SMN45 SMN80 SMN100 SMN75 SMN61 SMN91 SMN74 SMN97 SMN82	Small, SLA, Single
TMC_2^3	SMN7 SMN76 SMN17 SMN41 SMN20 SMN54 SMN78 SMN28 SMN56 SMN77 SMN42 SMN33 SMN69 SMN40 SMN51 SMN62 SMN64 SMN83	Chengdu, Single, Milling, Shijiazuang
TMC_3^3	SMN12 SMN24 SMN13 SMN63 SMN73 SMN85 SMN34	Small, Valve, Turning, Lanzhou
TMC_4^3	SMN9 SMN94 SMN67 SMN25 SMN79 SMN90	Small, Mass, Drilling
TMC_5^3	SMN56 SMN65 SMN44 SMN47 SMN70	Mass, Casting, Valve, Turning, FDM, SLA
TMC_6^3	SMN11 SMN53 SMN84 SMN18 SMN38	Single, Milling, Casting, SLS, Nanjing, Abroad
TMC_7^3	SMN89 SMN35 SMN59 SMN71	Small, Mass, Guangzhou
TMC_8^3	SMN2 SMN23 SMN8 SMN49	Jinan, Mass, Casting, Forging, SLA, Beijing, Xi'an, Guiyang, Wuhan
TMC_9^3	SMN16 SMN88 SMN86	Fan, Taiyuan
TMC_{10}^3	SMN15 SMN22 SMN30	Mass, Casting, Nanchang
TMC_{11}^3	SMN93 SMN46 SMN27	Small, Cutter, FDM, LOM

two SMNs in the TMC and the frequency of each MI feature occurred in all intersections. Afterward, we rank MI features based on their frequency and select MI features with larger frequencies as the common MI features of the TMC. To some extent, common MI features indicate what MI features SMNs in TMCs are interested in, which is why they self-organize themselves together. Table 5 shows the member of TMCs and their common MI features. Taking TMC_1^1 as an example, from Table 5, we can deduce that TMC_1^1 prefers to provide SLA additive manufacturing service and supply accumulator product for the customers located at Changsha.

In order to simulate the evolution of DMCs, we use the above SMNs and their MIRs as the initial snapshot of DSMNet at time step 1, and obtain five snapshots of DSMNet corresponding to other five time steps by selecting 20 SMNs in turn to change their MI features. For example, at time step 2, the MI features of the first 20 SMNs are changed. Afterward, for the snapshots of DSMNet, we also adopt SLPA algorithm to cluster SMNs into overlapping TMCs. The clustering results at six time steps are respectively shown in Table 5, Table 6, Table 7, Table 8, Table 9 and Table 10.

Then, according to equation (7), we calculate the similarity between TMCs and the fronts of DMCs. Afterward, according to the research of Greene *et al.* [43], we set matching threshold $\mu = 0.3$ to decide whether a TMC matches a DMC.

TABLE 8. TMCs and their common MI features at time step 4.

TMC	Member	Common MI feature
TMC_1^4	SMN4 SMN54 SMN87 SMN5 SMN95 SMN7 SMN76 SMN11 SMN43 SMN84 SMN17 SMN51 SMN36 SMN28 SMN56 SMN77 SMN42 SMN33 SMN69 SMN41 SMN40 SMN83 SMN50 SMN62 SMN64	Single, Chengdu, Milling
TMC_2^4	SMN15 SMN22 SMN19 SMN44 SMN66 SMN30 SMN35 SMN37 SMN57 SMN65 SMN38 SMN53 SMN70 SMN45 SMN75 SMN47 SMN68 SMN99 SMN100 SMN96	Mass, Casting, Nanchang
TMC_3^4	SMN13 SMN63 SMN73 SMN16 SMN88 SMN72 SMN34 SMN61 SMN80 SMN91 SMN97 SMN82 SMN86	Macao, Small, Single
TMC_4^4	SMN9 SMN94 SMN12 SMN24 SMN14 SMN55 SMN52 SMN85 SMN48 SMN49 SMN90	Small, Mass, Beijing
TMC_5^4	SMN93 SMN20 SMN46 SMN21 SMN60 SMN67 SMN25 SMN79 SMN27	Small, Mass, FDM, Shanghai
TMC_6^4	SMN10 SMN31 SMN58 SMN59 SMN81 SMN89 SMN75 SMN71	Mass, Chain, Small, Soldering, FFF, Guangzhou, Haikou, Changsha
TMC_7^4	SMN2 SMN23 SMN8 SMN78	Mass, Forging
TMC_8^4	SMN29 SMN92 SMN74	LOM, Chongqing, Abroad

TABLE 9. TMCs and their common MI features at time step 5.

TMC	Member	Common MI feature
TMC_1^5	SMN61 SMN2 SMN4 SMN63 SMN74 SMN87 SMN6 SMN9 SMN72 SMN80 SMN94 SMN12 SMN24 SMN13 SMN52 SMN73 SMN85 SMN34 SMN48 SMN49 SMN90	Small, Beijing, Turning
TMC_2^5	SMN61 SMN4 SMN63 SMN74 SMN87 SMN6 SMN9 SMN72 SMN80 SMN94 SMN12 SMN24 SMN13 SMN14 SMN55 SMN52 SMN73 SMN85 SMN34 SMN48 SMN49 SMN90	Mass, Chengdu, Shanghai
TMC_3^5	SMN1 SMN62 SMN71 SMN11 SMN43 SMN84 SMN21 SMN60 SMN67 SMN25 SMN79 SMN64 SMN76 SMN83 SMN82 SMN91	Mass, Small, Turning, Abroad
TMC_4^5	SMN54 SMN5 SMN95 SMN55 SMN17 SMN51 SMN77 SMN28 SMN56 SMN69 SMN42 SMN50	Mass, Small, Casting
TMC_5^5	SMN10 SMN31 SMN58 SMN59 SMN81 SMN89	Small, Mass, Chain
TMC_6^5	SMN38 SMN53 SMN99 SMN96	Casting, Mass, Mold, Tyre, LOM, Taiyuan, Abroad
TMC_7^5	SMN11 SMN43 SMN84	Small, Mass, Chain
TMC_8^5	SMN16 SMN88 SMN86	Fan, Taiyuan

TABLE 10. TMCs and their common MI features at time step 6.

TMC	Member	Common MI feature
TMC_1^6	SMN1 SMN61 SMN62 SMN88 SMN2 SMN23 SMN74 SMN5 SMN95 SMN6 SMN8 SMN78 SMN72 SMN80 SMN58 SMN59 SMN89 SMN12 SMN24 SMN15 SMN91 SMN17 SMN29 SMN75 SMN36 SMN52 SMN73 SMN92 SMN34 SMN35 SMN37 SMN57 SMN86 SMN38 SMN53 SMN45 SMN48 SMN49 SMN90 SMN99 SMN100 SMN66 SMN96 SMN98	Mass, Small, Turning
TMC_2^6	SMN54 SMN10 SMN31 SMN97 SMN51 SMN18 SMN77 SMN28 SMN56 SMN69 SMN42 SMN41 SMN40 SMN50	Mass, Shanghai, Haikou
TMC_3^6	SMN4 SMN63 SMN84 SMN13 SMN22 SMN19 SMN44 SMN20 SMN65 SMN30 SMN87 SMN70 SMN47 SMN82	Small, Casting, Mass, Hangzhou
TMC_4^6	SMN46 SMN21 SMN60 SMN67 SMN25 SMN68 SMN79 SMN81	Smal, Mass, Shanghai, Jinan
TMC_5^6	SMN9 SMN93 SMN94 SMN14 SMN55	Small, Mass, Tianjin, Nanjing
TMC_6^6	SMN83 SMN71 SMN64 SMN76	Mass, Soldering, FFF

The matching result between TMCs and DMCs at six time steps is shown in FIGURE 5a. According to the matching result, fundamental events can be mapped to the evolution of DMCs. FIGURE 5b shows the evolution of 18 DMCs over six time steps, and some fundamental events are mapped to DMCs. This figure clearly presents the evolution of DMCs, for example, from the figure, we can see that TMC_1^1 splits into TMC_1^2 and TMC_9^2 at time step 2.

V. DICUSSION

A. PARAMETER VALUES COMPARING AND CHOOSING

In the case, the values of threshold θ and parameter r have an important influence in clustering SMNs into overlapping TMCs by SLPA algorithm. In order to get the best clustering results, we conduct a comparative experiment to determine the values of threshold θ and parameter r. Firstly, we set the $\theta = 0.24$ and compare the effects of different values of

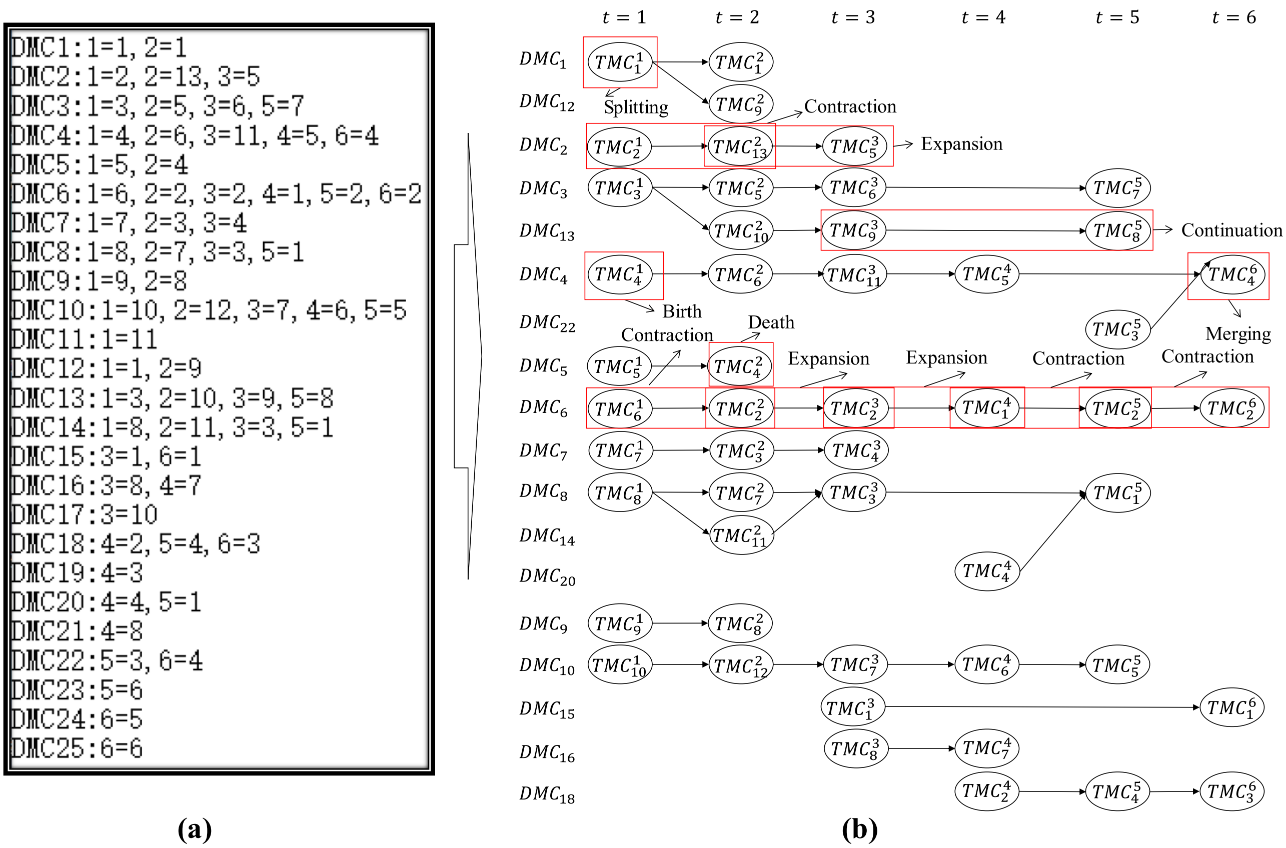

FIGURE 5. The result of matching TMCs and DMCs at six time steps and the evolution of DMCs at six time steps.

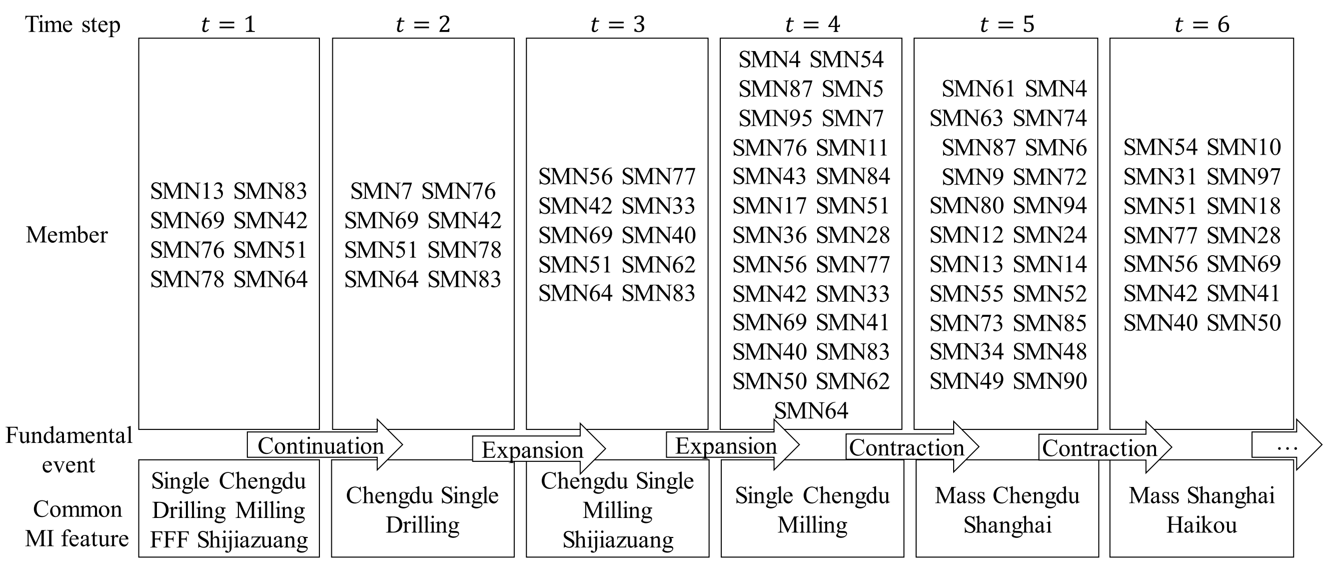

FIGURE 6. The evolution of DMC_6 at six time steps.

parameter r on the clustering results, as shown in Table 11. As seen from Table 11, with the value of parameter r increasing, the number of clustered TMCs increases and the member of the largest TMC decreases. The size of clustered TMCs is related to their fineness. The number of clustered TMCs is larger, their fineness is higher. On the other hand, in Table 11, the member number of the largest TMC is smaller, the size of clustered TMCs is smaller. However, when $r = 0.5$, there are no overlapping TMCs. Therefore, we consider $r = 0.45$ as the final parameter value in the SLPA algorithm. Afterward, we set 6 different values of threshold θ to obtain MIRs between SMNs, and the result is shown in Table 12. As seen

TABLE 11. The clustering results by setting different values of the parameter r when the threshold $\theta = 0.24$.

Number	Value of γ	The number of MC	The member of the largest MC	Is there any overlapping MC
1	0.01	8	34	yes
	0.05	7	33	yes
	0.1	9	29	yes
	0.15	9	25	yes
	0.2	9	25	yes
	0.25	9	25	yes
	0.3	9	24	yes
	0.35	9	24	yes
	0.4	9	23	yes
	0.45	9	23	yes
	0.5	9	22	no
2	0.01	9	34	yes
	0.05	9	30	yes
	0.1	9	27	yes
	0.15	10	24	yes
	0.2	10	23	yes
	0.25	10	22	yes
	0.3	11	12	yes
	0.35	11	11	yes
	0.4	11	11	yes
	0.45	11	11	yes
	0.5	11	9	no
3	0.01	10	29	yes
	0.05	9	25	yes
	0.1	9	22	yes
	0.15	10	20	yes
	0.2	10	20	yes
	0.25	9	20	yes
	0.3	9	20	yes
	0.35	9	20	yes
	0.4	9	20	yes
	0.45	9	20	no
	0.5	9	20	no
4	0.05	6	48	yes
	0.1	7	45	yes
	0.15	7	42	yes
	0.2	7	39	yes
	0.25	7	33	yes
	0.3	6	33	yes
	0.35	6	31	yes
	0.4	6	31	yes
	0.45	6	31	yes
	0.5	6	30	no
5	0.01	7	33	yes
	0.05	9	26	yes
	0.1	10	24	yes
	0.15	10	21	yes
	0.2	11	20	yes
	0.25	10	17	yes
	0.3	10	17	yes
	0.35	10	17	yes
	0.4	10	17	yes
	0.45	11	13	no
	0.5	11	13	no

from Table 12, if the value of threshold θ is too large, there are amount of single SMNs. On the contrary, if the value of threshold θ is too small, the number of clustered TMCs is small and their sizes are too large, which leads to too much overlap between TMCs. So we select $\theta = 0.24$ as the judging condition of whether there is a MIR between two SMNs.

B. EXPERIMENT RESULT ANALYSIS

1) EVOLUTION GRAPH

In order to further uncover the evolution of a DMC, we present its evolution as an evolution graph from its members, fundamental events and common MI features. Taking DMC_6 as an example, its evolution graph from time step 1

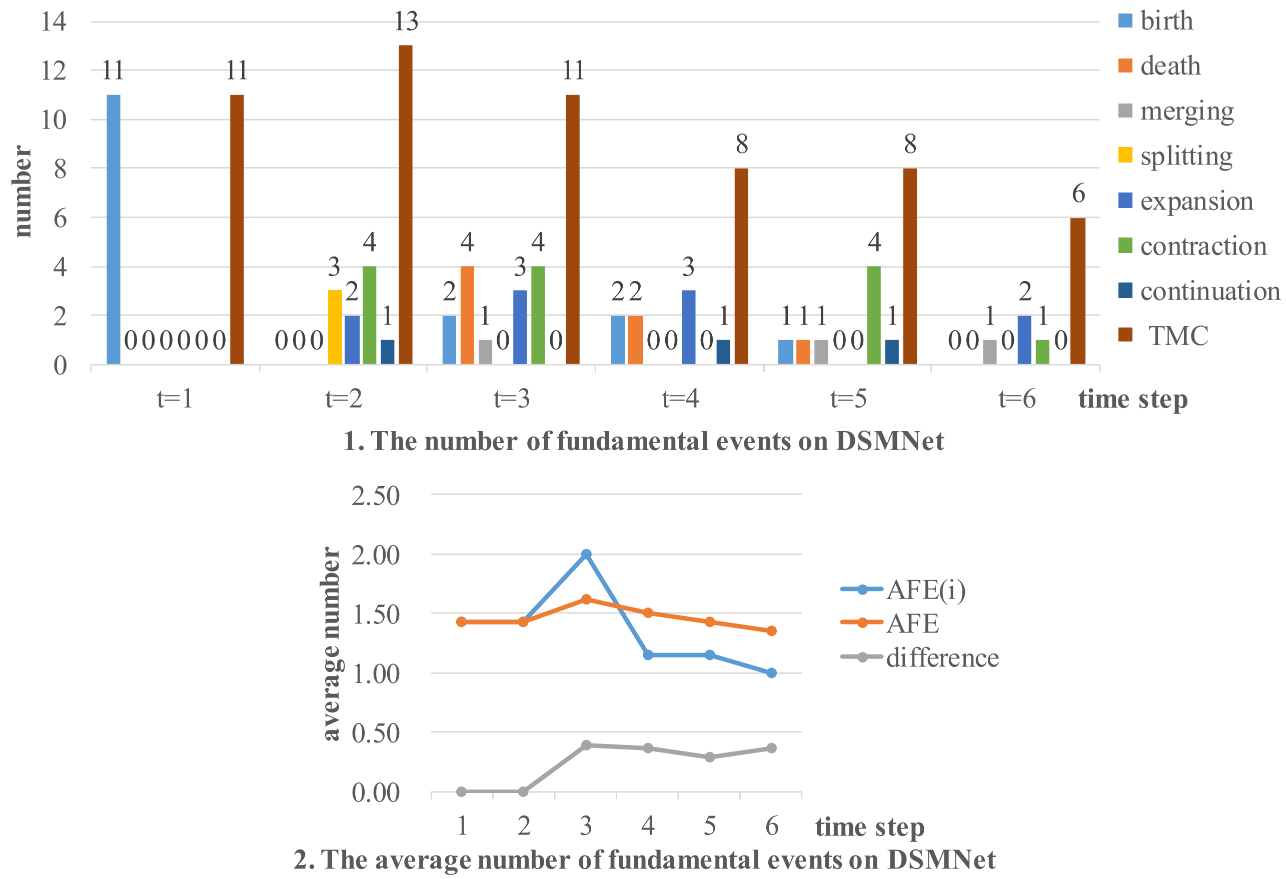

FIGURE 7. The number and average number of fundamental events on DSMNet.

to time step 6 is presented in FIGURE 6, from which we can see that DMC_6 experiences five fundamental events during its evolution, i.e., continuation, expansion, expansion, contraction, and contraction. The continuation of DMC_6 from time step 1 to time step 2 indicates that there are stable market demands for the manufacturing services that DMC_6 is interested in. From time step 2 to time step 4, increasing market demands for the manufacturing services has caused more and more SMNs to change their MIs to the common MIs of DMC_6, which leads to the expansion of DMC_6. At the same time, this also leads to increased competition in DMC_6 and the surplus of manufacturing resources in the manufacturing service domain. Afterwards, the contraction of DMC_6 happens from time step 4 to time step 6.

From the evolution graph of DMC_6, we can also see that it is operating well because it is identified in DSMNet from time step 1 to time step 6. This shows that DMC_6 has good development potential. Therefore, the manager of the network platform should devote more efforts (such as funds, technology, etc.) to protect such DMCs. At the same time, the manager of the network platform can analyze the reasons behind this to guide the operation of other DMCs. Similarly, other DMCs can also be analyzed.

2) FUNDAMENTAL EVENTS IN DSMNet

The number of fundamental events occurring in DSMNet can reflect the stability of DSMNet to a certain extent. Therefore, the average number of fundamental events on DSMNet is defined as (8), as shown at the bottom of the page, where n is the number of time steps DSMNet has experienced. AFE^i denotes the average number of fundamental events

$$
\begin{aligned}
AFE &= \frac{\sum_{i=1}^{n} AFE^i}{n} \\
AFE^i &= \frac{N^i_{birth} + N^i_{death} + N^i_{merging} + N^i_{splitting} + N^i_{expansion} + N^i_{contraction} + N^i_{continuation}}{7}
\end{aligned} \tag{8}
$$

TABLE 12. The clustering results by setting different values of threshold θ when parameter $r = 0.45$.

Value of threshold θ	The number of single SMNs	The number of clustered MC	The member of the largest MC	Is there any overlapping MC
0.21	0	4	50	yes
0.22	3	7	47	no
0.23	7	9	24	yes
0.24	12	11	11	yes
0.25	26	10	9	no
0.26	31	9	13	no

occuring in DSMNet at time step i, N^i_{birth}, N^i_{death}, $N^i_{merging}$, $N^i_{splitting}$, $N^i_{expansion}$, $N^i_{contraction}$, $N^i_{continuation}$ denote the number of times that "birth", "death", "merging", "splitting", "expansion", "contraction", "continuation" occur in DSMNet at time step i, respectively. By comparing the values of AFE^i and AFE, the stability of DSMNet at the time step i can be judged. The larger the difference of AFE^i and AFE, the worse the stability of DSMNet at the time step i. At this time, the manager of platform manager should be alert to exceptions that may occur on DSMNet.

In the case, the number and average number of fundamental events on DSMNet is presented in FIGURE 7. As can be seen from FIGURE 7, the difference of AFE^i and AFE on DSMNet is relatively small, not exceeding 0.5, which shows that the operation of DMCs on network platform is still quite stable so far.

VI. CONCLUSION

In this paper, a manufacturing network modeling and evolution characterizing approach is proposed. The purpose of this approach is to improve the efficiency of SMNs self-organization to form MCs, at the same time, to provide guidance for the operation of the network platform through characterizing the evolution of MCs. In this approach, firstly, a MI domain ontology is built to describe diverse MI of SMNs, and then the feature-based approach proposed is used to quantize MI similarity between SMNs. Furthermore, the speaker-listener label propagation algorithm is applied to cluster distributed SMNs into overlapping manufacturing communities based on their MIs. Finally, the evolution of MCs is characterized by defining seven fundamental events. The contributions of this study include following aspects: (1) the proposed approach can rapidly cluster distributed SMNs into overlapping MCs based on their MIRs, by which network platform can recommend potential partners for SMNs, thereby improving the efficiency of SMNs self-organizing into MCs. (2) the evolution of MCs can be characterized clearly by the proposed approach, by which the manager of the network platform in social manufacturing can get useful information on the operation of MCs to judge the stability of DSMNet.

Nevertheless, there also were limitations on our research. Firstly, in case study, only 100 SMNs produced by random are used to verify the feasibility of the proposed approach. More SMNs and real case also should be considered. In addition, the evolution patterns of MCs is also an attractive research, which is helpful to predict the future structure of DMCs. These limitations should be taken into consideration in future research.

REFERENCES

[1] F. Tao, Y. Cheng, L. Zhang, and A. Y. C. Nee, "Advanced manufacturing systems: Socialization characteristics and trends," *J. Intell. Manuf.*, vol. 28, no. 5, pp. 1079–1094, Jun. 2017.

[2] Y. Choi, D. Kang, H. Chae, and K. Kim, "An enterprise architecture framework for collaboration of virtual enterprise chains," *Int. J. Adv. Manuf. Technol.*, vol. 35, nos. 11–12, pp. 1065–1078, Feb. 2008.

[3] R. Sindhwani and V. Malhotra, "A framework to enhance agile manufacturing system," *Benchmarking: Int. J.*, vol. 24, no. 2, pp. 467–487, Mar. 2017.

[4] J. Gao, Y. Yao, V. C. Y. Zhu, L. Sun, and L. Lin, "Service-oriented manufacturing: A new product pattern and manufacturing paradigm," *J. Intell. Manuf.*, vol. 22, no. 3, pp. 435–446, Jun. 2011.

[5] B. Li, L. Zhang, S.-L. Wang, F. Tao, J. W. Cao, X. D. Jiang, X. Song, and X. D. Chai, "Cloud manufacturing: A new service-oriented networked manufacturing model," *Comput. Integr. Manuf. Syst.*, vol. 16, no. 1, pp. 1–7, 2010.

[6] Y. Lin, H. Xiong, M. Chen, D. Chen, Z. Zhou, Y. Ma, and D. T. Pham, "Design of agent-based supply chain-oriented virtual enterprise platform," *Kybernetes*, vol. 40, nos. 5–6, pp. 800–806, 2011.

[7] B. Huang, C. Li, C. Yin, and X. Zhao, "Cloud manufacturing service platform for small-and medium-sized enterprises," *Int. J. Adv. Manuf. Technol.*, vol. 65, nos. 9–12, pp. 1261–1272, 2013.

[8] K. Ding, P. Jiang, J. Leng, and W. Cao, "Modeling and analyzing of an enterprise relationship network in the context of social manufacturing," *Proc. Inst. Mech. Eng., B, J. Eng. Manuf.*, vol. 230, no. 4, pp. 752–769, Apr. 2016, doi: 10.1177/0954405414558730.

[9] X. Shang, B. Su, X. Liu, G. Xiong, and Z. You, "Social manufacture cloud mode in high-end apparel, footwear and hats," in *Proc. 11th World Congr. Intell. Control Autom.*, Shenyang, China, Jun. 2014, pp. 5264–5269.

[10] B. Mohajeri, "Paradigm shift from current manufacturing to social manufacturing, master degree," M.S. thesis, Dept. Ind. Eng. Manage., Aalto Univ. School Sci., Espoo, Finland, 2015.

[11] M. Hamalainen, B. Mohajeri, and T. Nyberg, "Removing barriers to sustainability research on personal fabrication and social manufacturing," *J. Cleaner Prod.*, vol. 180, pp. 666–681, Apr. 2018.

[12] P. Jiang and P. Li, "Shared factory: A new production node for social manufacturing in the context of sharing economy," *Proc. Inst. Mech. Eng., B, J. Eng. Manuf.*, vol. 234, nos. 1–2, pp. 285–294, Jan. 2020.

[13] P. Jiang, K. Ding, and J. Leng, "Towards a cyber-physical-social-connected and service-oriented manufacturing paradigm: Social manufacturing," *Manuf. Lett.*, vol. 7, pp. 15–21, Jan. 2016, doi: 10.1016/j.mfglet.2015.12.002.

[14] P. Jiang, J. Leng, K. Ding, P. Gu, and Y. Koren, "Social manufacturing as a sustainable paradigm for mass individualization," *Proc. Inst. Mech. Eng., B, J. Eng. Manuf.*, vol. 230, no. 10, pp. 1961–1968, Oct. 2016.

[15] P. Jiang, *Social Manufacturing: Fundamentals and Applications*, D. Pham, Ed. Cham, Switzerland: Springer, 2018.

[16] A.-L. Hirscher, K. Niinimäki, and C. M. Joyner Armstrong, "Social manufacturing in the fashion sector: New value creation through alternative design strategies?" *J. Cleaner Prod.*, vol. 172, pp. 4544–4554, Jan. 2018.

[17] P. Jiang, "Social manufacturing paradigm: Concepts, architecture and key enabled technologies," in *Social Manufacturing: Fundamentals and Applications*, D. Pham, Ed. Cham, Switzerland: Springer, 2018, pp. 13–50.
[18] M. Gan, X. Dou, and R. Jiang, "From ontology to semantic similarity: Calculation of ontology-based semantic similarity," *Sci. World J.*, vol. 2013, Feb. 2013, Art. no. 793091.
[19] D. Sánchez, M. Batet, D. Isern, and A. Valls, "Ontology-based semantic similarity: A new feature-based approach," *Expert Syst. Appl.*, vol. 39, no. 9, pp. 7718–7728, Jul. 2012.
[20] H. Al-Mubaid and H. A. Nguyen, "A cluster-based approach for semantic similarity in the biomedical domain," in *Proc. Int. Conf. IEEE Eng. Med. Biol. Soc.*, New York, NY, USA, Aug. 2006, pp. 2713–2717.
[21] G. Pirró, "A semantic similarity metric combining features and intrinsic information content," *Data Knowl. Eng.*, vol. 68, no. 11, pp. 1289–1308, Nov. 2009.
[22] J.-B. Gao, B.-W. Zhang, and X.-H. Chen, "A WordNet-based semantic similarity measurement combining edge-counting and information content theory," *Eng. Appl. Artif. Intell.*, vol. 39, pp. 80–88, Mar. 2015.
[23] M. A. Rodriguez and M. J. Egenhofer, "Determining semantic similarity among entity classes from different ontologies," *IEEE Trans. Knowl. Data Eng.*, vol. 15, no. 2, pp. 442–456, Mar. 2003.
[24] E. G. M. Petrakis, G. Varelas, A. Hliaoutakis, and R. Paraskevi, "X-similarity: Computing semantic similarity between concepts from different ontologies," *J. Digit. Inf. Manage.*, vol. 4, no. 4, pp. 233–237, 2006.
[25] M. Batet, D. Sánchez, A. Valls, and K. Gibert, "Semantic similarity estimation from multiple ontologies," *Int. J. Speech Technol.*, vol. 38, no. 1, pp. 29–44, Jan. 2013.
[26] A. Solé-Ribalta, D. Sánchez, M. Batet, and F. Serratosa, "Towards the estimation of feature-based semantic similarity using multiple ontologies," *Knowl.-Based Syst.*, vol. 55, pp. 101–113, Jan. 2014.
[27] G. Palla, I. Derényi, I. Farkas, and T. Vicsek, "Uncovering the overlapping community structure of complex networks in nature and society," *Nature*, vol. 435, p. 814, Jun. 2005.
[28] A. Lancichinetti, S. Fortunato, and J. Kertész, "Detecting the overlapping and hierarchical community structure in complex networks," *New J. Phys.*, vol. 11, no. 3, 2009, Art. no. 033015.
[29] J. Xie, B. K. Szymanski, and X. Liu, "SLPA: Uncovering overlapping communities in social networks via a speaker-listener interaction dynamic process," in *Proc. IEEE 11th Int. Conf. Data Mining Workshops*, Vancouver, BC, Canada, Dec. 2011, pp. 344–349.
[30] Z. Wu, Y. Lin, H. Wan, and S. Tian, "A fast and reasonable method for community detection with adjustable extent of overlapping," in *Proc. IEEE Int. Conf. Intell. Syst. Knowl. Eng.*, Hangzhou, China, Nov. 2010, pp. 15–16.
[31] A. Lancichinetti, F. Radicchi, J. J. Ramasco, and S. Fortunato, "Finding statistically significant communities in networks," *PLoS ONE*, vol. 6, no. 4, Apr. 2011, Art. no. e18961.
[32] W. Chen, Z. Liu, X. Sun, and Y. Wang, "A game-theoretic framework to identify overlapping communities in social networks," *Data Mining Knowl. Discovery*, vol. 21, no. 2, pp. 224–240, Sep. 2010.
[33] J. Xie, S. Kelley, and B. K. Szymanski, "Overlapping community detection in networks: The state-of-the-art and comparative study," *ACM Comput. Surv.*, vol. 45, no. 4, p. 43, 2013.
[34] U. N. Raghavan, R. Albert, and S. Kumara, "Near linear time algorithm to detect community structures in large-scale networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 76, no. 2, 2007, Art. no. 036106.
[35] S. Gregory, "Finding overlapping communities in networks by label propagation," *New J. Phys.*, vol. 12, no. 10, pp. 2011–2024, Oct. 2010.
[36] I. X. Y. Leung, P. Hui, P. Lio, and J. Crowcroft, "Towards real-time community detection in large networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 79, no. 6, Jun. 2009, Art. no. 066107.
[37] J. Xie and B. K. Szymanski, "Community detection using a neighborhood strength driven label propagation algorithm," in *Proc. IEEE Netw. Sci. Workshop*, Jun. 2011, pp. 188–195.
[38] J. Xie and B. K. Szymanski, "LabelRank: A stabilized label propagation algorithm for community detection in networks," in *Proc. IEEE 2nd Netw. Sci. Workshop (NSW)*, West Point, NY, USA, Apr. 2013, pp. 138–143.
[39] Q. Dai, M. Guo, Y. Liu, X. Liu, and L. Chen, "MLPA: Detecting overlapping communities by multi-label propagation approach," in *Proc. IEEE Congr. Evol. Comput.*, Cancun, Mexico, Jun. 2013, pp. 20–23.
[40] G. Palla, A. L. Barabási, and T. Vicsek, "Quantifying social group evolution," *Nature*, vol. 446, no. 7136, pp. 664–667, 2007.
[41] Y.-R. Lin, Y. Chi, S. Zhu, H. Sundaram, and B. L. Tseng, "Analyzing communities and their evolutions in dynamic social networks," *ACM Trans. Knowl. Discovery Data*, vol. 3, no. 2, p. 8, 2009.
[42] S. Asur, S. Parthasarathy, and D. Ucar, "An event-based framework for characterizing the evolutionary behavior of interaction graphs," *ACM Trans. Knowl. Discovery from Data*, vol. 3, no. 4, pp. 1–36, Nov. 2009.
[43] D. Greene, D. Doyle, and P. Cunningham, "Tracking the evolution of communities in dynamic social networks," in *Proc. Int. Conf. Adv. Social Netw. Anal. Mining*, Odense, Denmark, Aug. 2010, pp. 9–11.
[44] M. Takaffoli, F. Sangi, J. Fagnan, and O. R. Zäiane, "Community evolution mining in dynamic social networks," *Procedia - Social Behav. Sci.*, vol. 22, pp. 49–58, 2011.
[45] S. Saganowski, P. Bródka, and P. Kazienko, "Influence of the dynamic social network timeframe type and size on the group evolution discovery," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining*, Istanbul, Turkey, Aug. 2012, pp. 679–683.
[46] M. Oliveira, A. Guerreiro, and J. Gama, "Dynamic communities in evolving customer networks: An analysis using landmark and sliding windows," *Social Netw. Anal. Mining*, vol. 4, no. 1, p. 208, Dec. 2014.
[47] G. Diakidis, D. Karna, D. Fasarakis-Hilliard, D. Vogiatzis, and G. Paliouras, "Predicting the evolution of communities in social networks," in *Proc. 5th Int. Conf. Web Intell., Mining Semantics (WIMS)*, 2015, pp. 1–6.
[48] J. Xie and B. K. Szymanski, "Towards linear time overlapping community detection in social networks," in *Proc. Pacific–Asia Conf. Knowl. Discovery Data Mining*, May 2012, pp. 25–36.