Social Flocks: Simulating Crowds to Discover the Connection Between Spatial-Temporal Movements of People and Social Structure

Cheng-Te Li and Shou-De Lin

***Abstract*— Scientific studies from anthropologists, biologists, and sociologists hypothesized people who live in the geo neighborhood have more chances to contact with each other and construct the social relationships. This paper exploits a simulation-based approach to verify such hypothesis through unveiling the connection between the spatial-temporal movements of people and their social relationships. Based on the crowd simulation technique, we design an agent-based framework, social flocks, to model the geo spatial correlation of social elements. We simulate the movements of people to tackle two tasks, social network generation and network community detection. By mapping nodes in the network into agents in the simulation, we examine whether the social networks generated by our model can satisfy the network properties, such as high clustering coefficient, low average path length, and power-law degree distribution. Besides, given a social network, we simulate the social moving behaviors of agents/nodes to study the formation of communities. Experiments conducted for such two tasks verify the proposed hypotheses. Social flocks can also serve as a visualization platform for experts to explore the effects over the spatial, temporal, and social contexts. Through demonstrating how the simulation models are exploited to address social network problems, this paper encourages more studies on this direction.**

***Index Terms*— Community discovery, crowd simulation, network generation, social network, spatio-temporal modeling.**

I. Introduction

MANY anthropologists believed that spatial, temporal, and social contexts are the three components that have significant impact on the change of a society [5], [20]. Computer scientists had also found that the geographical activities of people can reveal the social connections between them [24], [29]. People who live within a certain geographical area have higher potentials to interact with each other, comparing to those who live far apart. Members of a tribe tend to construct social relationships through physical communications, and gradually form a tighter social group. Sooner or later, some adventurists would explore outside areas and develop new relationships with people from other cultures. Motivated by the above observations, this paper proposes a novel, intuitive, and anthropology-driven framework to study how people interact with one another and emerge the social structure.

Our main goal is to answer a scientific question: would the real-world spatial-temporal interactions between people lead to their social relationships and form the social structure? Moreover, what is the mechanism that governs the geographical movements of people to emerge the real-life social structure? While anthropologists [5], [20], biologists [6], [22], and sociologists [31] say yes to the first question, they can hardly verify it or their claim for the second question scientifically. This paper attempts to exploit the computational approach to validate this hypothesis. To do so, the direct manner is looking into the data. However, to really obtain the real data, we need to obtain not only the moving trajectory of ancient people (before a society is formed) but also "when" and "where" those people first met and become friends. It is extremely hard (if ever possible) to obtain such data. Therefore, we alternatively resort to the simulation-based approach to study and verify the mentioned hypothesis. We first design a simulation framework to model the spatial-temporal interaction among people through their movements, and based on such framework we examine whether social networks and social communities that satisfy real-world properties can be created. If the answer is yes, then we can confirm that there exists certain spatial-temporal interaction model that is capable of reproducing a real social network, which further provides a strong evidence to support the hypothesis.

In the literature, there are few studies that combine crowd simulation and social networks to unfold the interactions between people movements and social behaviors. Although *MobiCrowd* [23] uses social connections to guide the movements of agents, it cannot be used to explain how social communities are generated. Durupinar *et al.* [12] impose the theories of social psychology to simulate crowds, but the collective social behaviors in the spatio-temporal aspect do not be discussed. In addition, O'Connor *et al.* [35] introduce the social force to drive collective behaviors. However, one

Manuscript received August 26, 2017; accepted September 28, 2017. Date of publication November 17, 2017; date of current version February 23, 2018. This work was supported by the Ministry of Science and Technology of Taiwan under Grant 104-2221-E-001-027-MY2, Grant 106-2118-M-006-010-MY2, Grant 106-3114-E-006-002, and Grant 106-2628-E-006-005-MY3. *(Corresponding authors: Cheng-Te Li; Shou-De Lin.)*

C.-T. Li is with the Department of Statistics, National Cheng Kung University, Tainan 701, Taiwan (e-mail: chengte@mail.ncku.edu.tw).

S.-D. Lin is with the Department of Computer Science and Information Engineering, National Taiwan University, Taipei 106, Taiwan (e-mail: sdlin@csie.ntu.edu.tw).

Digital Object Identifier 10.1109/TCSS.2017.2763973

cannot acquire how social forces are obtained from the social network.

In particular, we consider two questions to answer. The first is whether, based on the spatial encountering of people, it is possible to form a social network that possesses some well-known phenomenon, such as high clustering coefficient (CC), low average path length (APL), and power-law degree distribution. The second is whether network communities can naturally emerge from the spatial-temporal interactions of people in the given social network. Eventually, we have found positive validation for the hypothesis: based on the simulation model we proposed, the spatial-temporal movements of people can play a significant role in forming social networks as well as establishing network communities.

To answer these two questions, we develop a novel simulation framework called social flocks. Social flocks aims to integrate the spatial, temporal, and social contexts of human beings to model the movement of people. Specifically, social flocks exploits the technique of *Crowd Simulation*, which belongs to computer animation, and aims at producing collective flocking behaviors of people by simulating the movement processes of individuals. The central idea of social flocks is to model each node as an agent that moves in the simulation space. There are two major tasks that we would like to establish through simulation in social flocks: 1) social network generation and 2) social network community detection. Since, social flocks can be regarded as a framework that exploits spatial-temporal correlation between the movements of people, being able to use such framework to produce real social networks or communities would imply the discovery of an underlying mechanism to form social networks and communities through physical interactions among people. The demonstration of social flocks framework can be accessed via http://mslab.csie.ntu.edu.tw/socialflocks/.

In the following we provide a brief overview of the two tasks we would like to focus on.

A. Social Network Generation

Social network generation models aim at producing artificial social networks satisfying some well-known properties that have been observed in real-world social networks [32]. Three of the most essential properties are: 1) high CC (nodes are densely connected to their neighborhood); 2) low APL (all pairs of nodes are connected via short paths on average); and 3) power-law degree distribution. As Watts and Strogatz [34] propose the random rewiring model to generate the small-world networks with high CC. Barabasi and Albert [7] propose the preferential attachment mechanism to generate the scale-free networks that satisfy both 2) and 3). More advanced generative methods [2], [8] have been proposed to model a series of sophisticated network properties as well. In reality, history shows that ancient people who lived in the same geographical region gradually interacted with each other to form societies. Therefore, in this paper, rather than resorting to the graph theory or other topological composition methods, we attempt to exploit the crowd simulation technique to simulate the people's movements in the spatial and temporal contexts for social network generation. The general idea is to create links between moving agents during the flocking simulation. We develop the *CrowdNetGen* component in social flocks to achieve this goal. In *CrowdNetGen*, we propose three agent-based network generation mechanisms, touch, neighborhood-density, and explorer models, where each of which possesses its own real-world physical meanings, to generate networks by gradually linking agents/nodes that are in contact with each other. We find that our approach is able to generate networks with high CC, low APL, and the power-law degree distribution. Some advanced properties mentioned by Akoglu and Faloutsos [2] and Leskovec *et al.* [27], including *Principal Eigenvalue Power*, *Densification Power Law (DPL)*, *Triangle Power Law* (TPL), and *Next-Large Connected Components*, can also be modeled by our method as well. Though some previous studies [4], [13], [18] have used the agent-based approach as spatial clues to generate social networks, the studies do not investigate or emphasize on whether the structural properties satisfy those of the real world. Since using the proposed agent-based simulation is able to generate the realistic social networks, it can be the first case to verify and strengthen the anthropologic hypothesis, when answering the scientific question.

B. Network Community Detection

Network community detection is a well-studied problem in the field of social network analysis and mining. Generally, it is tackled by first devising an objective function that captures the concept of the community structure (i.e., nodes within a cluster are tightly connected while nodes between communities are loosely connected) and then design a method to optimize such criterion. Many methods of community detection have been proposed and compared (see the review paper [28]). Most of them belong to the topology-based approach. We revisit the community detection problem by simulating the movements of agents/nodes. Our central idea is that people who live in a certain neighborhood tend to naturally form communities because they interact or contact with each other in the contexts of space and time [31] and have similar trajectories of daily movement. We first use real spatial-temporal check-in data to verify such hypothesis. Based on such hypothesis, given a social network, we simulate the flocking movements of agents/nodes. If communities emerge naturally, we can again verify the impact of spatial-temporal movements of people on their social relationship. Note that although it is generally believed that people belonging to the same community usually possess similar interests or attributes, here we concentrate on investigating the effects of spatial-temporal behaviors among agents on the formation of social communities. Based on the above insights and motivations, we present a novel crowd simulation-based approach, termed *Crowdstering*, to detect communities in a given social network. Specifically, we aim at producing social-based flocking behaviors among agents/nodes and exploiting the *trajectories* of agents to identify flocks as the network communities. In addition, *Crowdstering* is able to find not only groups but also outlier nodes in the given network. Finally, we compare our method with four conventional community detection algorithms, and the results show that our

agent-based method is competitive to the modern algorithms in terms of accuracy. As a result, the anthropologic answer to the correlation from the spatial-temporal to the social aspect can be verified again. Note that our *Crowdstering* is not devised to beat existing state-of-the-art community detection methods in terms of efficiency and scalability. The main argument of our model is that simulating people's movements with social guidance can lead to network communities. Hence our experiments are devised to verify this argument. If the performance of the proposed model is at least competitive to the state-of-the-art community detection methods, we say the hypothesis is hold and our model is indeed effective.

We summarize the contributions in the following.

1) We propose a computational approach to validate the anthropologic hypothesis: the spatial-temporal movements of people can lead to the construction of social relationships and form social groups. To tackle such a problem, we bring a marriage between *Crowd Simulation* and *Social Network Analysis*, and develop a simulation platform *Social Flocks* to study how the spatial, temporal, and social dimensions can affect one another to produce the dynamics of network structures and collective flocking behaviors. The hypothesis is eventually verified.
2) *CrowdNetGen*, proposed based on *Social Flocks*, can itself be regarded as a novel social network generation method. It contains three spatial-temporal simulation-based network generation models, *touch*, *neighborhood-density*, and *explorer*. Each of the models possesses its own physical meaning that corresponds to the real-life human behaviors. Experimental results show that our models can successfully produce the networks satisfying real-world properties including the small-world and power-law effects.
3) Another model, *Crowdstering*, proposed on *Social Flocks*, can be regarded as a network community formation. Different from existing traditional community detection methods based on graph clustering, our model possesses the physical meaning to model how ancient people form social groups through contacting each other in space. We also conduct a series of experiments on two real-world data sets to demonstrate the feasibility and accuracy of our model.
4) Comparing to the traditional graph-based and topology-driven approaches for social network analysis, *Social Flocks* provides a new angle to handle such problems through crowd simulation. One main advantage is that through *Social Flocks*, we are able to visualize the process of formulation of a society and community through physical interaction while performing tasks of social network generation and network community detection. Furthermore, *Social Flocks* can provide an experimental simulation platform for natural scientists to study the social and complex system.

II. Social Flocks Framework

We present the social flocks framework as shown in Fig. 1. Social flocks takes advantage of the Reynolds' flocking simulation model [39] as the backbone, in which we associate each node in the social network with a moving agent. The social flocks framework is developed to perform two tasks: social network generation and network community detection, for answering the proposed scientific question. In *CrowdNetGen*, we first propose the *Touch* model and the *Neighborhood-Density* model to generate social networks that satisfy the real-world properties such as high CC, low APL, and power-law degree distribution. To enhance the scalability, we further propose the *explorer* model that allows the generation of large-scaled networks in parallel. Second, given an underlying social network, we propose a spatial-temporal crowd simulation method for community detection. Our method, named *Crowdstering*, introduces the *acquaintance force* into the flocking simulation using the information obtained from the network structure. By simulating agents/nodes that constantly move together and form a community, we can exploit their trajectories for node clustering.

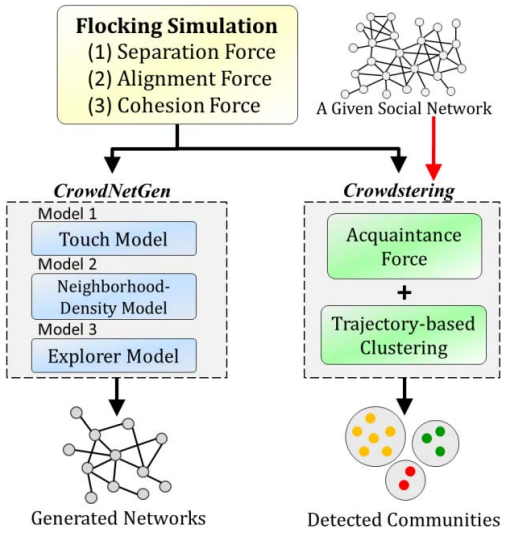

Fig. 1. Framework of the proposed social flocks.

We first briefly describe the Reynolds' flocking model [39], which aims to produce the flocking behaviors among artificial agents in a dynamic virtual environment. Reynolds' model consists of three steering rules. The first is the *separation force* f_s, which steers each agent to avoid crowding local flockmates. If agents move too close to each other in space, the separation force will drive them away from others. The direction of the separation force is computed by summing up the directions from the neighboring agents to that of the current agent. The second is the *alignment force* f_a, which steers each agent to move toward the average heading of local flockmates in its visible region. The direction of the alignment force is derived by subtracting the average direction of the neighboring agents from the direction of the current agent. The third is the *cohesion force* f_c, which steers each agent to move toward the average position of the local flockmates. Note that each agent is an independent actor and has his own local perception to navigate. Also note that the cohesion force keeps a flock of agents to move together while the separation force prohibits the agents to collide with each other when they move too close. Each agent is affected by only one cohesion force of another

agent while each agent can be affected by multiple separation forces given there are many other close agents. Furthermore, even there is just one separation force existing, it is unlikely this force would cancel the cohesion force since their direction are very different. For any given agent, the direction of its cohesion force points toward the "average position" of its perceptible agents. For any given agent, there can be multiple separation forces, each point to the opposite direction of the close-by agent. Setting parameters at 0.5 for f_s and f_c will not cancel their effects. More technical details about these forces can refer to Reynolds *et al.* [39].

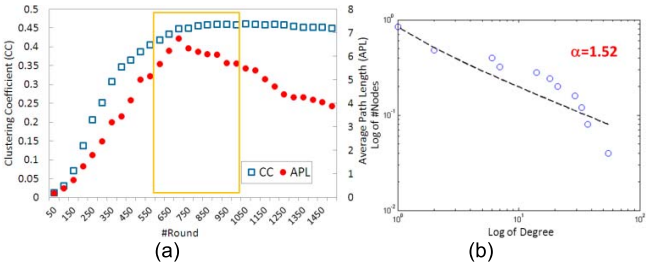

Fig. 2. (a) CC and APL and (b) log–log degree distribution ($\alpha = 1.52$) under the *Touch* model.

III. CrowdNetGen: Network Generation

Our first attempt to answer the scientific question is to generate real-world social networks by simulating the spatial-temporal moving agents. We rely on no topological clues for the network generation. The agent-based approach allows us to explore the relationships between the network structure and the movements of agents in the space. The fundamental idea of our model is to consider each node in the network as an agent in the virtual environment. As the flocking simulation proceeds, we gradually add edges to connect from one agent/node to another based on one of the following three models. Note that in this paper, we will use the term *agents* and *nodes* interchangeably, which refer to vertices in a social network. In addition, a *round* of simulation is finished when each agent performs one action to move itself to another position.

The intuition of using the three steering forces to generate social networks is threefold. First, the cohesion force creates the possibility for agents to move together and then make connections with each other. Such action can be mapped to that people live in a certain neighborhood tend to interact with one another. Second, once some small groups of agents gather together, the alignment force plays the role of keeping agents in each flocking group to move together. In the sense of real-world movements of people, those lived and acquainted with each other will keep their relationships and have higher potential to gather in the space. Third, in the real-life society, people in different social groups (e.g., family, association, and organization) could be acquainted with each other. The separation force allows agents belonging to a certain flock to have some chance for creating connections with individuals in other flocking groups. In other words, to some extent the separation force will help create the weak ties which link different communities in a network.

A. Touch Model

This *touch* model aims to produce a network that reflects the way people in the pretelecommunication era form relationships by physically meeting each other in space. In the *touch* model, an edge is added to connect agents (or equivalently, nodes) u and v only when u and v have a physical touch in the simulation during the simulating process. In the experiment of Fig. 2, 200 agents are allocated as the initial isolated nodes in the network. They are randomly scattered in the space at the beginning of the simulation. We assume that each person usually has the half probability to interact with people in different flocking groups (i.e., communities) and have the high loyalty to interact with the native group. And, thus we set the three steering forces (f_s, f_a, and f_c) to (0.5, 0.8, and 0.5), to have the realistic effect of flocking behaviors among agents. That says, agents tend to flock together and form some small groups, but still allow few agents to move from one flocking group to another. Note that the same setting is applied to the following two models in Sections III-B and III-C. Fig. 2 shows that as the number of rounds (#Round) increases from 0 to 1550, the network quickly gathers edges and both CC[1] and APL gradually increase. When the #round reaches 600 to 1000, as highlighted by the orange rectangle, the generated networks possess the small-world properties of high CC $\approx$ 0.45 and low APL $\approx$ 6.5. Unfortunately, the touch model does not quite produce the scale-free property. The power-law exponent α is about 1.5. Though it is slightly smaller than that of many real-world social networks ($\alpha \approx 2$), it still demonstrates the highly skew degree distribution.

B. Neighborhood-Density Model

To produce a network with higher power-law parameter α, we propose an alternative *neighborhood-density* model to generate the networks. The basic intuition is that an agent has higher likelihood to develop connections with others when there are more agents around, and furthermore it is more likely to develop relationship with the centralized persons in a group that the peripheral outliers. Therefore, for an agent/node v, we define its *neighborhood-density* k_v as the number of neighboring agents within its surrounding region (a circle area with a radius of ε pixels) in the space (set to be 40 pixels in the experiment for Figs 2 and 3). During the flocking simulation, for each agent/ node v, if k_v is larger than a predefined density threshold (set to be 5 for Fig. 3), the system adds an edge to connect v to a node u with the highest k_u value in v's local perception area because such node is more likely to be a centralized leader.

Fig. 3(a) presents the values of CC and APL in the simulation. We can see that as the rounds of simulation increases from 0 to 150, the network quickly gathers edges and both CC and APL increase drastically. The small-world properties emerge with even higher CC $\approx$ 0.75 and low APL $\approx$ 6

[1] $\mathrm{CC} = (1/n)\sum_{i=1}^{n} (2|\{e_{jk}\}|/d_i(d_i-1))$, where n is the number of nodes in the networks, d_i the degree of node i, $|\{e_{jk}\}|$ is the number of edges between any pair of node i's neighboring node j and k.

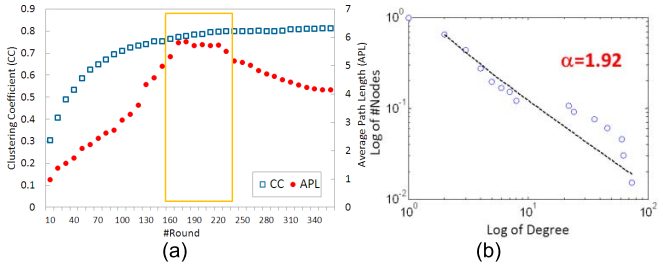

Fig. 3. (a) CC and APL and (b) log–log degree distribution ($\alpha = 1.92$) at #round = 170 under the *Neighborhood-Density* model.

between round 150 and 200, as highlighted by the orange rectangle. As #round further increases, more edges are introduced to connect nodes, which cause the APL to decrease gradually. The CC values remain stable because edges are added for all nodes in a neighborhood at the same time (i.e., forming triangles and tending to become cliques locally in the network). Note that comparing with the Touch model, the neighborhood-density model takes fewer rounds to produce the values of high CC and low APL. It is because the touch situations are relatively less likely to happen. In addition, the generated networks under the neighborhood-density model follow the power-law degree distribution. The log-scaled degree distribution at #round = 170 is shown in Fig. 3(b), where the power-law exponent α is 1.92. In brief, the neighborhood-density model can produce networks that satisfy the three properties of real-world networks, i.e., high CC, low APL, and power-law degree distribution ($\alpha \approx 2$ in average).

C. Explorer Model

Though the neighborhood-density model is able to produce networks that satisfy the three major properties of real-world networks, it suffers a drawback on scalability since the simulation will become slow when the number of agents increases. To enhance the scalability, we devise an advanced *explorer* model based on the neighborhood-density model that allows the parallel computation of simulation-based social network generation. The central idea is to divide the simulation space into several smaller areas, and perform parallel simulation on each individual area. That says, using a cloud or clustering machine with n cores, we can scale up the size of networks to n times. However, one major concern for such strategy is that after doing so, graphs belonging to different areas are not connected. And, thus the network could have no *giant component*, which is against the real-world observation that social networks generally have a giant component that connects most people [14], [32]. To address such concern, we propose the *explorer* model. The intuition is as follows. As the world is composed by several geographical regions, in general people lived in the same region contacted with each other more frequently. Though in the beginning different cultures were isolated without any connection, with time a few individuals (which are usually called the explorers) start to travel outside their own territory for exploration. Those explorers may meet explorers from other cultures and therefore forms connections between different groups of people.

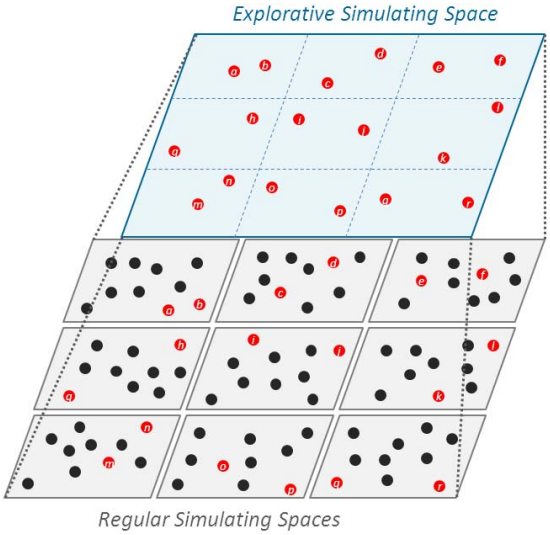

Fig. 4. Simulating spaces for the *Explorer* model: the larger space is divided into nine regions. Each region consists of normal agents (black ones) and explorer agents (red ones). Those explorer agents are allowed to move cross border to form cross-region connections. Technically, one can execute $9 + 1$ different neighborhood-density models in parallel threads to create a social network of larger size.

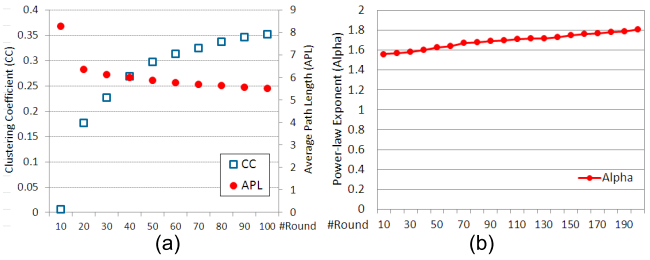

Fig. 5. Resulting property values as the simulating run increases. (a) Small-world effect gradually forms, especially after #Round = 40. (b) Long-tailed degree distributions whose power-law exponent α is about 1.6–1.8.

We attempt to model such exploration phenomenon in our simulation to connect different groups of people into a larger giant component.

As shown in Fig. 4, the explorer model starts from dividing spaces into different geographical regions and performing our neighborhood-density model independently in each region. That is, N agents are allocated randomly in each subarea and create links between agents following the neighborhood-density model. In addition, in each region we randomly pick a small amount of agents as the *explorer* agents (the red ones in Fig. 4) who are responsible for making interactions with explorers belonging to other societies. We perform one additional simulation that allows the explorers to move freely in the whole area to form connections, also based on the neighborhood-density model. Eventually, the explorers will play the role of mediators to form relatively few links across regions, and therefore closer regions have higher chance to be connected more tightly.

We demonstrate the effectiveness of the explorer model for generating the network containing 20 000 nodes. The results are shown in Figs 5 and 6. The first part is to investigate

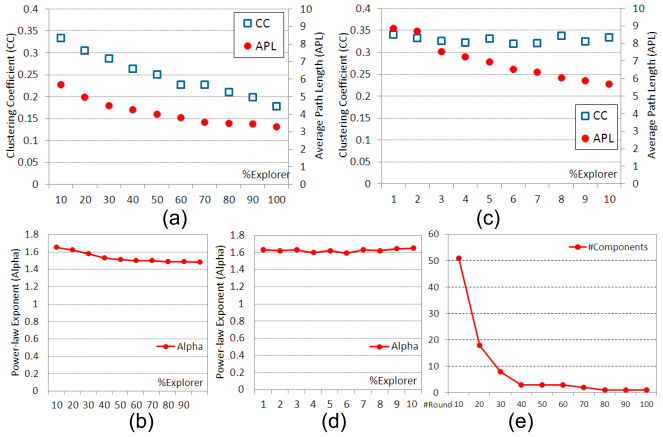

Fig. 6. Resulting property values by varying the percentage of explorers (%Explorer). (a) Values of CC and APL decreases as the %Explorer increases from 10% to 100%. (b) Power-law exponent α slowly decreases as well. (c) Values of CC remains stable about 3.5 while the APL values decreases from 9 to 6 as the %Explorer increases from 1% to 10%. (d) Power-law exponent α remains stable about 1.6. (e) number of connected components (#Components) as the simulation proceeds from 1 to 100.

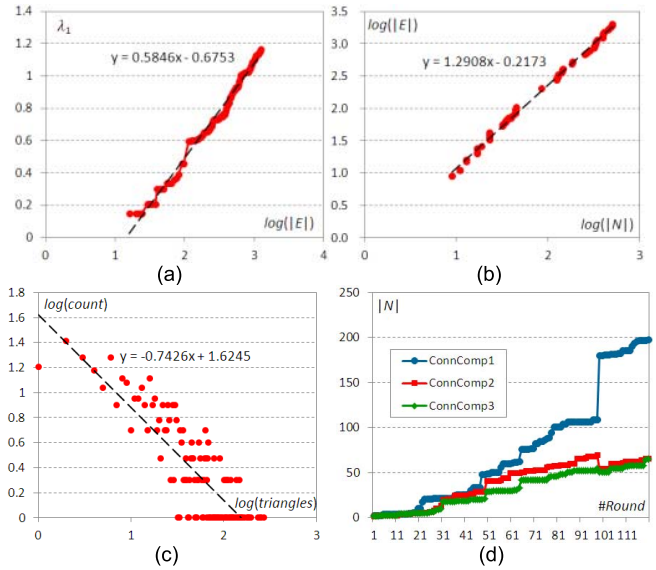

Fig. 7. Using the explorer model, our generated networks can satisfy advanced properties of (a) principal eigenvalue power law, (b) DPL, (c) TPL, and (d) constant size secondary and tertiary connected components.

the properties of the generated networks as the simulating runs increases, where the percentage of explorers in each space is set to be 10%. In Fig. 5(a), we can find that the values of CC quickly increase in the beginning and reaches 0.3 after $\#Round = 50$ while the values of APL quickly saturated to 6. Such phenomenon shows that the explorer agents are able to effectively construct connections to be local agents and play the role of bridges to shorten the distances between agents in different societies. On the other hand, in Fig. 5(b), we can find the power-law exponent α slowly increases from 1.6 toward 1.8. We think it is due to that as the simulation proceeds, few explorer agents/nodes not only be active to make connections to the flockmates in each of their own society, and further accumulate their links (degree vales) to other explorer agents from different societies. On the other hand, we investigate the effect of the percentage of explorers, denoted by %Explorer, on the network properties. Fig. 6(a) shows the values of CC and APL decreases toward 0.2 and 3, respectively, as the %Explorer increases from 10% to 100%. It is reasonable as that the more explorers are there, the more connections are established between groups which can significantly reduce APL. Furthermore, since the explorers are allowed to move more freely in a larger space, when more regular persons become explorers, it becomes harder to create triangle relationships (since people are moving more freely) and therefore reduce the CC. In Fig. 6(c) and (d), we further study the interval of %Explorer between 1% and 10% on the network properties. We can find that the APL is sensitive to the percentage of explorers while the CC is not. In Fig. 6(e), we show that the explorer model is able to emerge the *giant components* as the simulation proceeds. The results demonstrate the importance of explorers who are in charge of maintaining the weak links to agents in different societies. In short, to generate the networks satisfying real-world properties under the proposed explorer model, it needs only about 3%–10% explorers, which does match some real-world scenarios.

Here we response to our goal, answering the scientific question, mentioned in the beginning of this paper. Since the networks generated by the flocking simulation of spatial-temporal moving agents satisfy the common network properties, we obtain a positive echo to the anthropologic hypothesis: the spatial-temporal movements of people have effects on the formation of social structure. Those interact in a certain neighborhood, along with some explorers who travel around different geographic regions, tend to naturally compose the real-world social networks.

D. Advanced Network Properties

We further investigate whether the networks generated from the *Explorer* model can satisfy four additional advanced properties, described by Akoglu and Faloutsos [2] and Leskovec *et al.* [27]. The first is called *Principal Eigenvalue Power Law* ($\lambda_1 PL$), which describes that there is a power-law relationship between the largest eigenvalue λ_1 of the adjacency matrix of a network and the number of edges E, denoted by $\lambda_1(t) \propto E(t)^{\delta}$ with $\delta < 1$, over time. The second is the DPL, which points out that the number of nodes N and the number of edges E follow a power-law distribution, denoted by $E(t) \propto N(t)^{\alpha}$ with $2 > \alpha > 1$, over time. The third is the TPL, which says that the number of triangles Δ and the number of nodes that involve in these triangles follow a power-law distribution, denoted by $f(\Delta) \propto \Delta^{\sigma}$ with $\sigma < 0$. The fourth is the *constant size secondary and tertiary connected components*. It indicates that accompanied with the growth of the giant component, the secondary and tertiary connected components tend to remain constant in size or grow very slowly with small oscillations.

In Fig. 7, we show the quantitative results of the above four properties for the networks generated by the *Explorer* model. Note that the distributions of λ_1PL, DPL, and TPL are generated from the snapshot at $\#Round = 50$. We conclude

that the *Explorer* model is able to satisfy all the desired static and dynamic properties: for λ_1PL: $\delta = 0.58 < 1$, for DPL: $1 < \alpha = 1.29 < 2$, for TPL: $\sigma = -0.74 < 0$. All these observations help confirm our model can produce realistic social structures.

IV. Crowdstering: Crowd Simulation-Based Community Detection

Anthropology history tells us that the geographic information plays a significant role on the formation of different kinds of human societies [31], which indicates that people who have similar moving behaviors tend to have higher potential to interact with one another and then form community. In this section, we first investigate the real spatial-temporal data to verify such hypothesis. Then, we would like to show that the real network communities can indeed be formed under a spatial-temporal moving model by introducing the *Crowdstering* method, which models the spatial-temporal movements of individuals who belongs to a given social network. Specifically, given a social network, by mapping nodes into agents in the simulation space, we aim to use the moving trajectory of each agent generated from the flocking simulation to make communities emerge naturally. In other words, we model how people with tight connections flock together, and exploit such geographical moving patterns to discover the social communities.

Crowdstering consists of two parts. First, an *acquaintance force* based on the underlying social relationships is proposed to guide the flocking behaviors. Second, a *trajectory-based clustering* mechanism, which aims at grouping agents/nodes with similar flocking behaviors, is introduced to discover communities. We also conduct a series of experiments to compare the effectiveness of our method with conventional community detection algorithms. Note that *Crowdstering* is not devised to outperform the existing or state-of-the-art methods in terms of accuracy, efficiency, and scalability. Our idea is that if simulating the moving agents in a space is able to naturally emerge network communities (i.e., the performance of *Crwodstering* is competitive to some of existing methods in terms of effectiveness), we can response to the anthropologic hypothesis with a positive echo.

A. Hypothesis Verification From Real Data

We first use real spatial-temporal data to verify the underlining hypothesis: people who have close social relationship tend to have similar moving behaviors. We verify the hypothesis by testing whether pairs of individuals in the same communities have more similar daily trajectories than pairs in different communities. Gowalla location-based social check-in data [24] are employed for such purpose. We extract two sets of the check-in records from two urban districts, San Francisco Bay Area and New York City. We first extract users whose check-in frequency is higher than certain threshold (i.e., 10–30). The Gowalla data reveal the underlining social network for users in each district. Given the social network, we then use the methods, *Fast Modularity* [11], *WalkTrap* [37], *MapGen* [36], and *CFinder* [1], to find the communities of users from four different aspects. The details of these methods are described in Section V-C. The commonly used *Eucledean Time-uniform Distance* (ETD) [25] is adopted to measure the spatial-temporal distance of trajectories. The average ETD is computed over pairs of users belonging to different communities (BtwComm) to compare with pairs of users in the same communities (InComm). Fig. 8 shows the results that the average pairwise ETD distances of people in the same communities are much lower than those belonging to different communities. Such results confirm our hypothesis and provide the empirical justification of the models we propose.

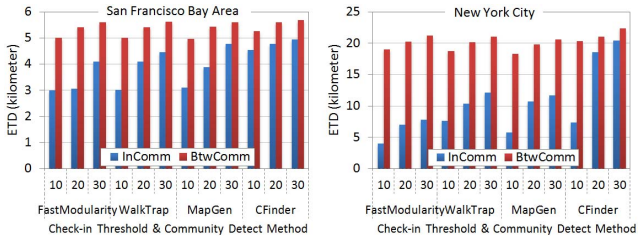

Fig. 8. Eucliean time-uniform distance (in kilometer) on BtwComm and InComm for two cities. The results verify the hypothesis that users in the same communities have more similar moving behaviors than those belong to different communities.

B. Acquaintance Force

In the Reynolds' flocking model, the concept of social interaction is not considered. Nevertheless, the spirit of any community detection algorithm is to use the targeted social network to detect communities. Therefore, our first step is to design a simulation framework that considers the given social network. In other words, we aim at exploiting the social network to direct the simulation of agents such that those acquainted with each other will flock together in the space. We introduce an additional force, the *acquaintance force* f_q, into the simulation. Different from the original three forces that are purely based on the spatial closeness or the agent's individual circumstance, the *acquaintance force* considers the social closeness between agents/nodes to bring attraction for agents that are close to each other in the network and repulsion to agents that are away from each other.

We devise the acquaintance force according to the distance between two agents in the given network. For an agent u, the acquaintance force $\overrightarrow{f_q(u)}$ is computed by

$$\overrightarrow{f_q(u)} = \sum_{v \in R_\delta(u)} \overrightarrow{d(u,v)} \cdot \left(\frac{\theta - \text{Length}(u,v) + 1}{\theta} \right)$$

where $R_\delta(u)$ is the set of surrounding agents under a certain neighboring threshold δ of an agent u, which is used to control the influence range of the acquaintance force. Length(u, v) is the length of shortest path between the node u and v in the network, and θ is a *boundary parameter* which determines the boundary between the attraction and repulsion forces. That says, if Length$(u, v) \leq \theta$, the agent v will exert an attraction

force to u. If Length$(u, v) = \theta + 1$, v will exert neither attraction nor repulsion force to u. And if Length$(u, v) > \theta + 1$, v exert a repulsion force to u. In our experiment, θ is set to be 2 and δ is set to be 200 pixels. In addition, the direction vector from the current agent u to the neighboring agent v, denoted by $\overrightarrow{d(u, v)}$, is the difference of their position vectors: $\overrightarrow{d(u, v)} = \overrightarrow{p_v} - \overrightarrow{p_u}$, where $\overrightarrow{p_u}$ is the position vector of the agent u in the simulating space. Note that the "plus 1" in the numerator of the parentheses is to ensure those nearest nodes (i.e., Length$(u, v) = 1$) do not get any discount and can acquire the highest force values among all kinds of graph length. In the end, by integrating our acquaintance force with original three steering forces, the eventual force $\overrightarrow{f(u)}$ to guide an agent u is

$$\overrightarrow{f(u)} = w_s \cdot \overrightarrow{f_s(u)} + w_a \cdot \overrightarrow{f_a(u)} + w_c \cdot \overrightarrow{f_c(u)} + w_q \cdot \overrightarrow{f_q(u)}$$

where w_s, w_a, w_c, and w_q are the weights of separation, alignment, cohesion, and acquaintance forces, respectively. In the later experiment, we set them to be (w_s, w_a, w_c, and w_q) = (0.3, 0.3, 0.3, and 0.1).

Note that the fundamental difference between Reynolds' three forces and our acquaintance force is that acquaintance force is created because of the social network (the length means the distance between nodes in the graph, not the real geo distance in a geographical area). Nearby neighbors in the network can attract each other while faraway ones repel each. Reynolds' three forces rely mainly on the geographical distance rather than social distance. In other words, if we have only the acquaintance force (i.e., no Reynolds' forces), agents acquainted with each other in the social network would gather and move locally at certain areas and cannot form a global collective flocking.

C. Trajectory-Based Community Detection

Normally people are involved in a variety of events every day. They sometimes act on their own, for example typing or reading, and sometimes interact with others, for example sports, carpooling, and hanging out with friends. Nevertheless, people tend to interact more frequently with friends, colleagues, or family members than strangers. From the geographical point of view, it is reasonable to assume that people in the same community have higher chance to move in the neighborhood for a longer period of time. Such fact is the main idea behind our simulation-based community detection.

Since the acquaintance force together with the Reynolds' separation, alignment, and cohesion forces are capable of producing the social-based flocking behaviors among agents/nodes, we move one step further to exploit them for the task of community detection. During the simulation, we have observed that agents which are closer to each other tend to walk together for longer period of time, and the outliers in the graph might not have too many accompanies along the way. Based on such observation, we have developed a hypothesis that the moving trajectories of nodes in the same community should resemble one another comparing with those that were not in the same community. Therefore, we consider the trajectories of agents during the simulation as a clue to find the communities in a given social network.

In the beginning of the simulation, the agents are scattered randomly in the space. Note that here we do not assign the agent's initial position based on the topology of the underlying social network, rather the topological information is used only for producing the acquaintance force. We attempt to avoid using too much topological information in order to distinguish our approach with other topology-driven community detection algorithms. During the early runs of simulation, agents wonder around from their initial random positions and gradually form groups. Once a group is formed, most of the members stick with each other for longer period of time. Therefore, we ignore the trajectory in the early rounds for clustering. Then based on the trajectories, the system groups those agents with similar trajectories into the same community.

We exploit the conventional data clustering techniques to group agents based on their trajectories. For each agent v, we consider the positions in its trajectory segment $T_v = \langle (x_{t-\lambda+1},\ y_{t-\lambda+1}),\ (x_{t-\lambda+2},\ y_{t-\lambda+2}), \ldots, (x_t, y_t) \rangle$, where λ is the considered trajectory length and t = *#Round*, as the attributes in clustering. The above essentially says that we choose a trajectory of length λ starting from time $t - \lambda$ to time t. We combine the density-based spatial clustering *DBSCAN* technique [15] with the k-means clustering method to achieve our final goal. The DBSCAN is used to determine the number of clusters k and the outliers. Then, we remove the outliers, and apply the determined value k in the k-means clustering method to group the trajectories into different clusters.

The complete algorithm is shown in Algorithm 1. Lines 1 and 2 initialize the setting of the social flocking simulation. Lines 3–11 perform the simulation and store the look-based trajectories as the simulation enters into the $(\lambda + 1)$th round (Lines 7–9), in which we use the previous one-round (i.e., t_1) positions of agents to compute the steering forces and determine the positions in current round t (Lined 5 and 6). We also store the trajectory and snapshot information for future clustering (Lines 7 and 8). Based on the last snapshot (t = *#Round*) of the space, Line 12 runs the DBSCAN algorithms to obtain the number of communities k and the set of outlier agents/nodes O_D. Line 13 removes the set of discovered outlier agents O_D from the trajectories of agents T. Line 14 executes the k-means clustering algorithm to find a set of flocking groups as the communities.

We use Fig. 9 as an example to demonstrate the effect of the proposed trajectory-based community detection. Given a social network shown in the right panel of Fig. 9, we can figure out there are two apparent communities in green and orange. With the trajectories of agents, our *Crowdstering* can find two social-based flocking groups, as shown in the left panel. In addition, our method is also able to recognize the outliers whose corresponding agents and trajectories are labeled by the gray color. Moreover, the trajectories also show that in the early rounds, the agents did scatter around in the space; but later the trajectories of the agents belonging to the

Algorithm 1 Trajectory-Based Community Detection

Input: (a) $G = (V, E)$: the given social network, where V is the set of nodes as well as the set of agents in the simulating space and E is the set of edges; (b) λ: the length of look-back trajectory used to detect communities; (c) $\#Round$: the number of total rounds which determines when to terminate the simulation and stop recording the trajectories; (d) S_t: the visual snapshot of the simulating space at round t. Note that the look-back trajectory of an agent u is a sequence of spatial positions is represented as $T_u = \langle (x_{t-\lambda+1}, y_{t-\lambda+1}), (x_{t-\lambda+2}, y_{t-\lambda+2}), \ldots, (x_t, y_t)\rangle$, and the set of trajectories of all agents/nodes is denoted by $T = \{T_1, T_2, \ldots, T_{|V|}\}$.

Output: $C_D = \{d_1, d_2, \ldots, d_k\}$: a set of detected communities, where d_i is a set of nodes and k is the number of detected communities. O_D: the set of detected outlier nodes.

1: Associate each node with an agent for the simulation.
2: Scatter the agents/nodes randomly in the space.
3: **for** $t = 1$ to $\#Round$ **do:**
4: **for** $u = 1$ to $|V|$ **do:**
5: Compute the steering force $\overrightarrow{f(u)}$.
6: $Coord_u = (x_t, y_t)$: update u's coordinate into S_t by $\overrightarrow{f(u)}$.
7: **if** $(t > \lambda)$ **do:**
8: Add (x_t, y_t) into the trajectory record T_u of agent u.
9: $(k, O_D) = DBSCAN(S_{\#Round}, epsilonRadius, minPts)$. //$epsilonRadius = 50$, $minPts = 2$ in this work.
10: Remove outlier agents from T.
11: $C_D = k\text{-}MEANS(T, k)$.
12: **Return:** (C_D, O_D).

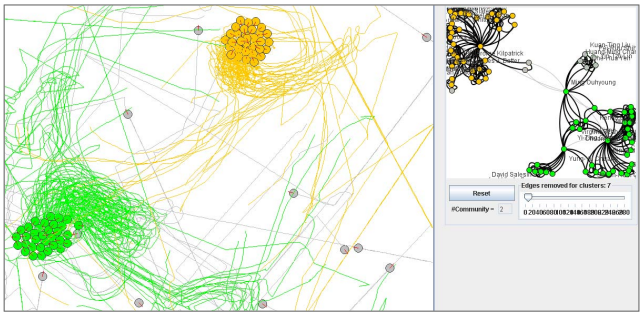

Fig. 9. Exampling trajectory-based clustering for a network with two communities. Social network (right). Trajectories of agents in two communities are colored (left).

same community gradually converge to the similar area, which justifies our assumption of ignoring the early trajectories while performing clustering.

V. Experiments

We conduct a series of experiments to demonstrate the effectiveness of the proposed crowd simulation-based approach for finding the communities in a social network. The evaluation plan consists of three parts: 1) we provide a visualization demo of the simulation process that allows the users to view how agents interact with each other and naturally form communities; 2) we compare the performance of *Crowdstering* with other graph-based community detection algorithms; and 3) we perform a series of sensitivity analysis on some parameters. The general goal is to show whether the spatial-temporal simulation of agents can make some groups of agents, which belongs to the actual network communities, naturally emerge during the flocking process. It is important to emphasize that our method is not designed to outperform the existing or state-of-the-art methods in terms of accuracy, efficiency, and scalability. The comparison to other community detection method aims to assist us to validate the anthropologic hypothesis and answer the scientific question. If the effectiveness of our detected communities is competitive to some conventional algorithms, we can justify the spatial-temporal movements of people actually have some effect on the formation of social structure.

A. Evaluation Settings

Two networks are used for the experiments. The first is extracted from DBLP[2] Computer Science Bibliography Database. It is a subgraph of the entire DBLP coauthorship networks. Such DBLP subgraph contains 83 nodes and 283 edges. We manually investigate the authors and identify two communities in the graph, as shown in the upper-right and lower-left panels. The second is the well-known friendship network of the Zachary's karate club [3] that is commonly used to evaluate the performance of community detection methods. The karate friendship network consists of 34 nodes and 78 links. This network contains two ground-truth communities: {1, 2, 3, 4, 5, 6, 7, 8, 11, 12, 13, 14, 17, 18, 20, 22} and {9, 10, 15, 16, 19, 21, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34}. It is apparent that the community structures in the DBLP subgraph are more obvious than that in the karate friendship network, which is nontrivial to be identified by human eyes. Hence, one can expect that the performance of community detection in the DBLP subgraph is better than that of the karate friendship network. Note that these two networks are used for the simulation and the evaluation. We need not to consider whether the nodes (e.g., authors in DBLP subgraph) come to have physical contacts in the real world.

There are two ways to evaluate a community detection algorithm. If the ground truth is missed, we can use the internal criteria to make sure the output communities are faithful. If the gold standard exists, one can simply compute the accuracy of an algorithm and compare it with the true answers. We exploit both strategies for the evaluation. First, we employ the measure of *conductance* [28], which evaluate whether the members of a detected community are tightly connected to each other and whether it is the opposite case for the nodes from different communities. Given a set of nodes C in a community, the conductance value can be generated using $\phi(C) = c_B/\min(\mathrm{Vol}(C), \mathrm{Vol}(V \backslash C))$, where c_B is the size of boundary edges of a community, $c_B = |\{(u, v) : u \in S, v \notin S|$, and $\mathrm{Vol}(S) = \sum_{u \in S} d(u)$, where $d(u)$ is the degree of node u. For a network G detected to have a set of communities S_{com}, its conductance $\phi(G) = \sum_{C_i \in S_{\mathrm{com}}} \phi(C)$. Lower conductance reflects implies better quality.

[2]http://dblp.uni-trier.de/

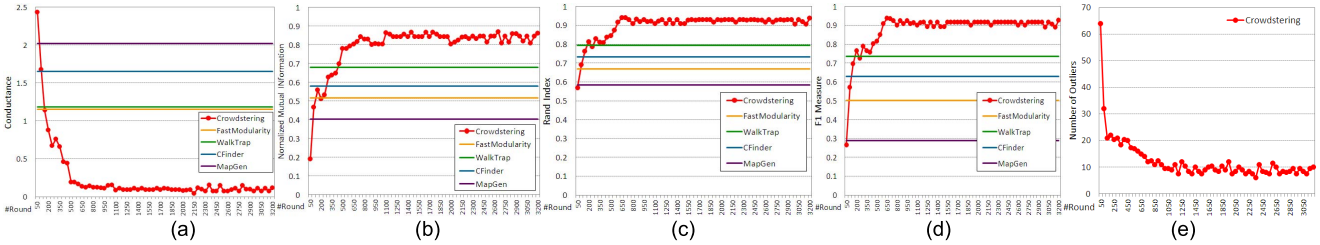

Fig. 10. Experimental results on the DBLP subgraph as the simulation proceeds for the (a)–(d) conductance, NMI, RI, and F_1 measure. (e) Number of outliers discovered by our *Crowdstering* is reduced to fewer than 10 when the simulation stabilized.

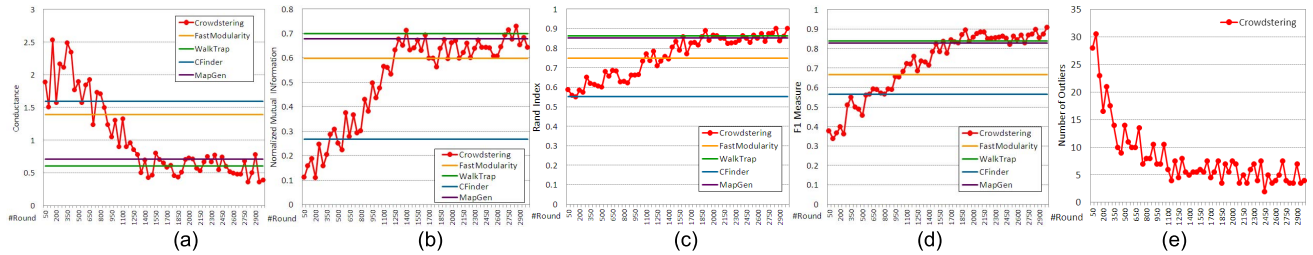

Fig. 11. Experimental results on the karate friendship network for the (a)–(d) conductance, NMI, RI, and F_1 measure. (e) Number of outliers discovered by our *Crowdstering* is reduced to be around 5 when the simulation stabilized.

Second, based on the ground truth, we employ three internal measures, *Normalized Mutual Information* (NMI), *Rand Index (RI)*, and F_1 *measure*, to examine the performance. The NMI score follows the idea in the information theory to measure the mutual dependence between the detected communities and the gold standard. Higher scores of these three measures indicate better performance. The NMI formula is as below

$$\mathrm{NMI}(C_D, C_G) = \frac{\sum_k \sum_j \frac{|d_k \cap c_j|}{N} \log \frac{N|d_k \cap c_j|}{|d_k||c_j|}}{(H(C_D) + H(C_G))/2} \tag{1}$$

where $C_D = \{d_1, d_2, \ldots, d_k\}$ is the set of detected communities, $C_G = \{c_1, c_2, \ldots, c_j\}$ is the set of ground-truth communities, and H is the entropy calculated by $H(C_D) = \sum_k (|d_k|/N) log(|d_k|/N)$.

RI can be computed via $\mathrm{RI} = (\mathrm{TP} + \mathrm{TN})/(\mathrm{TP} + \mathrm{FP} + \mathrm{FN} + \mathrm{TN})$, where TP is the true positive rate, TN is the true negative rate, FP is the false positive rate, and FN is the false negative rate. In addition, the F_1 measure is computed via $F_1 = 2\mathrm{P}^*\mathrm{R}/(\mathrm{P} + \mathrm{R})$, where P is the precision score and R is the recall score. For more details about the above measures, please refer to [30]. In addition to the above criteria, we will also report how many outliers are discovered in the experiments.

B. Comparison to Existing Methods

We compare our method with four well-known community detection algorithms. The first is *FastModularity* [11], which aims to maximize the objective function *modularity*. The second is *WalkTrap* [37] that assumes those nodes within the same community are much easier to reach from each other, and exploits the *Random Walk* technique to find communities. The third is *CFinder* [1], whose idea is to regard the community as a kind of relaxation of clique. CFinder uses *Clique Percolation* to find the communities. The fourth is *MapGen* [36] that decomposes the given network into communities by optimally compressing a description of information flows.

We use the measures conductance, NMI, RI, and F_1 measure (note that except the first, the rest three prefers larger values) for the evaluation. We also show the number of outliers discovered by our approach. Note that the other four methods do not have the capability to find nodes that might not belong to any group in a network. Figs. 10 and 11 demonstrate the experimental results as the simulation proceeds. Note that since the values of the four traditional methods are fixed as a certain value for those metrics, we draw horizontal lines to represent them in Figs 10 and 11. The experiments are conducted on the two data sets of the DBLP subgraph and the karate friendship network, respectively. Note that the resulting curves in our model are derived through calculating the average values of performing the simulation 10 times.

For the DBLP subgraph, as shown in Fig. 10, *Crowdstering* is competitive to the others for all the four measures after certain rounds of simulation. It can be commonly observed that the resulting curves of our method are fluctuant in the beginning rounds of the simulation. It is due to that the agents started from randomly assigned initial positions in the space, and are trying to find the right path in the beginning. When the simulation gets into the stable state, the fluctuation situation becomes mild. In Fig. 10(e), the number of outliers discovered by our approach turns out to become stable because the movements of agents gradually saturate.

The experimental results on the karate friendship network are shown in Fig. 11. We can apparently find that the extents of fluctuation for the curves are much bigger than those of the DBLP subgraph. Such effects are resulted from that the two communities of the karate friendship network are much closer and therefore harder to distinguish. The boundary of the two communities is quite indistinct even by human eyes.

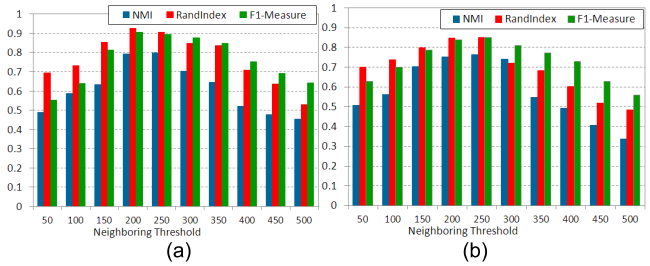

Fig. 12. Sensitivity tests about the neighboring threshold of the acquaintance force. We vary the neighboring threshold from 50 to 500 on (a) DBLP subgraph and (b) karate friendship network.

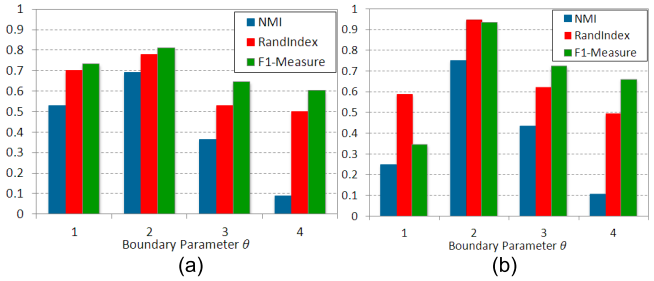

Fig. 13. Sensitivity tests about the boundary parameter of the acquaintance force. We set the boundary parameter as 1–4 on (a) DBLP subgraph and (b) karate friendship network.

Nevertheless, the results of our method are still competitive to WalkTrap and MapGen and outperform the other two methods when entering into the stable state (#Round $>$ 1200).

C. Sensitivity Tests for Parameters

1) Neighboring Threshold: We first examine how the acquaintance force affects the discovered communities. The neighboring threshold δ (i.e., local perception) controls the influence range of the acquaintance force. Note that all the results are based on the average values under the simulation rounds from 1000 to 2000, which are the stable states of the flocking behaviors. Fig. 12 shows the effects of different neighboring thresholds on the DBLP subgraph and karate friendship network. We can find that generally the best results locate from 200 to 250. We believe such result is due to that when the neighboring threshold is too small, the flocking behaviors will become very local and the size of each flock tends to be very small. Eventually, agents/nodes belonging to the same community are split into several smaller groups. On the contrary, if the neighboring threshold is too large, agents/nodes belonging to different communities will easily flock together. In this sense, the groups originally belong to different communities are prone to mix with one another and make the performance worse.

2) Boundary Parameter: Recall that in our Acquaintance fource, the *boundary parameter* θ determines the boundary between the attraction and repulsion forces. The experimental results of varying the boundary parameter are shown in Fig. 13. We can find that the boundary parameter 2 can have the best results. We believe it results from that looser boundaries allow more neighboring agents to exert the attractive force, which hurts the performance since agents/nodes from different communities are prone to form larger flocking groups. However, when the boundary parameter is set to 1, the negative acquaintance force dominates and leads to many small sized or even isolated groups, because fewer agents are considered to be acquainted with each other. Hence the performance becomes worse. In general we suggest that the most suitable value of the boundary parameter is 2.

VI. Related Work

A. Network Generation

We review topology-based network generation models. The earliest model is *Erdos–Renyi* (ER) *model* [14]. Though ER model does not fit real-world phenomena perfectly, it is the basis of many existing topology-based network generation models. The *Barabasi-Albert* (BA) *model* [7] introduces the idea of *preferential attachment* to produce networks with the *power-law* degree distribution. The *Watts–Strogatz* (WS) *model* [34] models the *small-world phenomenon*, and is able to create networks satisfying the small-world property. Leskovec *et al.* [27] propose the *forest fire* model to capture two observed properties, the *DPL* (i.e., the number of edges grows super linearly with the number of nodes) and the *shrinking diameter* properties, in evolving information networks. *Recursive Tensor Model* [2] is proposed to model a series of network properties. They exploit the idea of the *entropy plot* to discover the structure's fractal patterns during a graph's evolution. They propose to use a 3-D tensor to represent a graph by adding a time dimension, and combine *Zipf's law* and 2-D *Random Typing* to produce graphs that fit a list of observed properties. Moreover, AGM [10] generates and samples networks considering the attributes of nodes.

Though many methods are able to generate networks satisfying real-world properties, they consider the generative processes from the pure topological perspective and neglect the facts that the social networks are essentially the outcome under the spatial and temporal contexts. In this paper, we use the agent-based simulation approach that enables us to consider the spatial, temporal, and social factors together, comparing to existing works that utilize rule-based heuristics or topological-driven heuristics (e.g., ER model, WS model, BA model). On the other hand, though some existing studies [4], [13], [17], [18] have used the agent-based approach as spatial clues to generate graphs, they do not investigate or emphasize on whether the structural properties satisfy those of the real-world.

B. Network Community Detection

A number of methods were proposed to detect communities. The general approach to find dense subgraphs is by partitioning the graph recursively [16]. Recently, researchers utilize the *modularity*-based approach [32], [33] to detect communities. The idea behind modularity is to ensure the number of edges across groups is not only small but also smaller than expected. Leskovec *et al.* [28] exploit the conductance measure to define the *network community profile plot* for comparing the effectiveness of different community detection algorithms. To increase the time efficiency, Raghavan *et al.* [38] propose

a *label propagation* algorithm to find the communities in large networks. Yin *et al.* [21] detect seed-based communities based on network motifs. In addition, though a multiagent simulation is proposed to detect communities [26], the method cannot explain how communities are formed in the geographical space. In short, existing approaches to find community structures ignore the dynamic process of forming communities, which is the main goal of our approach. That is, we can not only produce groups but also demonstrate how the groups are generated. To the best of our knowledge, this paper is the first work to exploit the spatial-based simulation for finding communities in a social network.

VII. Conclusion

This paper aims to answer a scientific question: do the spatial-temporal movements of people affect the formation of social relationships and community structure, and how? By discovering a connection between crowd simulation (as the spatial and temporal factors) and social network analysis (as the social context), we are able to provide a positive assertion on this hypothesis. Note that the main goal of this paper is not about proposing a more efficient or accurate social network generation and community detection algorithms, rather we want to show that it is possible to tackle such problems from a different angle, which to some extent captures how a real society or community is formed. Through demonstrating how the simulation models can be exploited to address social network problems, we hope to point out and encourage more studies on this new direction of solutions for social network analysis.

References

[1] B. Adamcsek, G. Palla, I. J. Farkas, I. Derényi, and T. Vicsek, "CFinder: Locating cliques and overlapping modules in biological networks," *Bioinformatics*, vol. 22, no. 8, pp. 1021–1023, 2006.

[2] L. Akoglu and C. Faloutsos, "RTG: A recursive realistic graph generator using random typing," in *Proc. Eur. Conf. Principles Data Mining Knowl. Discovery (PKDD)*, 2009, pp. 13–28.

[3] W. W. Zachary, "An information flow model for conflict and fission in small groups," *J. Anthropol. Res.*, vol. 33, no. 4, pp. 452–473, 1977.

[4] M. Aoyagi and A. Namatame, "Synchronization in mobile agents and effects of network topology," *J. Agent-Based Soc. Syst.*, vol. 6, no. 3, pp. 31–42, 2009.

[5] M. E. F. Bloch, *How we Think They Think: Anthropological Approaches to Cognition, Memory, and Literacy*. Boulder. Co, USA: Westview, 1998.

[6] R. H. MacArthur, *Geographical Ecology: Patterns in the Distribution of Species*. Princeton, NJ, USA: Princeton Univ. Press, 1984.

[7] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, 1999.

[8] D. Chakrabarti and C. Faloutsos, "Graph mining: Laws, generators, and algorithms," *ACM Comput. Surv.*, vol. 38, no. 1, 2006, Art. no. 2.

[9] S. Chenney, "Flow tiles," in *Proc. ACM SIGGRAPH/Eurograph. Symp. Comput. Animation*, 2004, pp. 233–242.

[10] J. J. Pfeiffer, III, S. Moreno, T. La Fond, J. Neville, and B. Gallagher, "Attributed graph models: Modeling network structure with correlated attributes," in *Proc. Int. World Wide Web Conf. (WWW)*, 2014, pp. 831–842.

[11] A. Clauset, M. E. J. Newman, and C. Moore, "Finding community structure in very large networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 70, no. 6, p. 066111, 2004.

[12] F. Durupinar, U. Güdükbay, A. Aman, and N. I. Badler, "Psychological parameters for crowd simulation: From audiences to mobs," *IEEE Trans. Vis. Comput. Graphics*, vol. 22, no. 9, pp. 2145–2159, Sep. 2016.

[13] B. Edmonds, "How are physical and social spaces related?—Cognitive agents as the necessary 'glue,'" *J. Agent-Based Comput. Model.*, vol. 5, pp. 195–214, 2006. [Online]. Available: https://link.springer.com/chapter/10.1007/3-7908-1721-X_10

[14] P. Erdös and A. Rényi, "On random graphs I," *Publicationes Mathematicae Debrecen*, vol. 6, pp. 290–297, 1959.

[15] M. Ester, H. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters a density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, Aug. 1996, pp. 226–231.

[16] M. Fiedler, "Algebraic connectivity of graphs," *Czechoslovak Math. J.*, vol. 23, no. 2, pp. 298–305, 1973.

[17] Y. Gu, Y. Sun, and J. Gao, "The Co-evolution model for social network evolving and opinion migration," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2017, pp. 175–184.

[18] L. Hamill and N. Gilbert, "Social circles: A simple structure for agent-based social network models," *J. Artif. Soc. Soc. Simul.*, vol. 12, no. 2, 2009, Art. no. 3.

[19] D. Helbing, J. Farkas, and T. Vicsek, "Simulating dynamical features of escape panic," *Nature*, vol. 407, no. 6803, pp. 487–490, 2000.

[20] I. Hodder and C. Orton, *Spatial Analysis in Archaeology*. Cambridge, U.K.: Cambridge Univ. Press, 1976.

[21] H. Yin, A. R. Benson, J. Leskovec, and D. F. Gleich, "Local higher-order graph clustering," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2017, pp. 555–564.

[22] M. Slatkin, "Gene flow and the geographic structure of natural populations," *Science*, vol. 236, no. 4803, pp. 787–792, 1987.

[23] C.-T. Li and H.-P. Hsieh, "MobiCrowd: Simulating crowds with periodic and social mobility," in *Proc. Int. Conf. Auto. Agents Multi-Agent Syst. (AAMAS)*, 2014, pp. 1627–1628.

[24] E. Cho, S. A. Myers, and J. Leskovec, "Friendship and mobility: User movement in location-based social networks," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2011, pp. 1082–1090.

[25] H. Cao, O. Wolfson, and G. Trajcevski, "Spatio-temporal data reduction with deterministic error bounds," *VLDB J.*, vol. 15, no. 3, pp. 211–228, 2006.

[26] H. Zardi, L. B. Romdhane, and Z. Guessoum, "A multi-agent homophily-based approach for community detection in social networks," in *Proc. IEEE Int. Conf. Tools Artif. Intell. (ICTAI)*, Nov. 2014, pp. 501–505.

[27] J. Leskovec, J. Kleinberg, and C. Faloutsos, "Graph evolution: Densification and shrinking diameters," *ACM Trans. Knowl. Discovery Data*, vol. 1, no. 1, 2007, Art. no. 2.

[28] J. Leskovec, K. Lang, and M. Mahoney, "Empirical comparison of algorithms for network community detection," in *Proc. Int. World Wide Web Conf. (WWW)*, 2010, pp. 631–640.

[29] H.-P. Hsieh, R. Yan, and C.-T. Li, "Where you go reveals who you know: Analyzing social ties from millions of footprints," in *Proc. ACM Int. Conf. Inf. Knowl. Manage. (CIKM)*, 2015, pp. 1839–1842.

[30] C. D. Manning, P. Raghavan, and H. Schutze, *Introduction to Information Retrieval: Evaluation of Clustering*. Cambridge, U.K.: Cambridge Univ. Press, 2008.

[31] M. McPherson, L. Smith-Lovin, and J. M. Cook, "Birds of a feather: Homophily in social networks," *Annu. Rev. Sociol.*, vol. 27, no. 1, pp. 415–444, 2001.

[32] M. E. J. Newman, "The structure and function of complex networks," *SIAM Rev.*, vol. 45, no. 2, pp. 167–256, 2003.

[33] M. E. J. Newman, "Fast algorithm for detecting community structure in networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 69, no. 6, p. 066133, 2004.

[34] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," *Nature*, vol. 393, pp. 440–442, Jun. 1998.

[35] S. O'Connor, F. Liarokapis, and C. Jayne, "Perceived realism of crowd behaviour with social forces," in *Proc. Int. Conf. Inf. Vis.*, Jul. 2015, pp. 494–499.

[36] M. Rosvall and C. Bergstrom, "Maps of information flow reveal community structure in complex networks," in *Proc. Nat. Acad. Sci. (PNAS)*, 2008, pp. 1–6.

[37] P. Pons and M. Latapy, "Computing communities in large networks using random walks," *J. Graph Algorithms Appl.*, vol. 10, no. 2, pp. 191–218, 2006.

[38] 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. 3, p. 03610, 2007.

[39] C. W. Reynolds, "Flocks, herds and schools: A distributed behavioral model," in *Proc. ACM SIGGRAPH Int. Conf. Comput. Graph. Interact. Techn. (SIGGRAPH)*, 1987, pp. 25–34.

Cheng-Te Li received the M.S. and Ph.D. degrees from the Graduate Institute of Networking and Multimedia, National Taiwan University, Taipei, Taiwan, in 2009 and 2013, respectively.

He was an Assistant Research Fellow at the Research Center for Information Technology Innovation in Academia Sinica, Tainan, Taiwan. He is currently an Assistant Professor with the Department of Statistics, National Cheng Kung University, Tainan, Taiwan. His current research interests include social and information networks, data mining, and social media analytics.

Dr. Li was a recipient of the Facebook Fellowship 2012 Finalist Award, the ACM KDD Cup 2012 First Prize, the IEEE/ACM ASONAM 2011 Best Paper Award, and the Microsoft Research Asia Fellowship 2010.

Shou-De Lin received the B.S. degree from Electrical Engineering Department, National Taiwan University, Taipei, Taiwan, the M.S. degree in electrical engineering from the University of Michigan, Ann Arbor, MI, USA, the M.S. degree in computational linguistics and the Ph.D. degree in computer science from the University of Southern California, Los Angeles, CA, USA.

He is currently a Full Professor with the CSIE Department, National Taiwan University. His current research interests include machine learning and data mining, social network analysis, and natural language processing.

Dr. Lin was a recipient of the Best Paper Award at the IEEE WI conference 2003, the Google Research Award 2007, the Microsoft Research Award in 2008, 2015, and 2016, the Merit Paper Award in TAAI 2010, 2014, and 2016, the Best Paper Award in ASONAM 2011, and the U.S. Aerospace AFOSR/AOARD Research Award winner for five years. He is the all-time winners in ACM KDD Cup, leading or co-leading the NTU team to win five championships. He also leads a team to win WSDM Cup 2016.