ARTICLE IN PRESS

+ MODEL

Available online at www.sciencedirect.com

ScienceDirect

Karbala International Journal of Modern Science xx (2017) 1−10

http://www.journals.elsevier.com/karbala-international-journal-of-modern-science/

A reliable path between target users and clients in social networks using an inverted ant colony optimization algorithm

Saied Asghari [a,*], Kianoush Azadi [b]

[a] *Young Researchers and Elite Club, Tabriz Branch, Islamic Azad University, Tabriz, Iran*

[b] *Department of Industrial Engineering, Amir Kabir University of Technology, Tehran, Iran*

Received 10 November 2016; revised 29 April 2017; accepted 23 May 2017

Abstract

Internet has become an integral section of human life. Millions of people are joining online social networks every day, interacting with others whom they did not know already. Establishing trust among those indirectly connected users performs a crucial role in improving the quality of social network services and creating the security for them. Nowadays, there are many paths between clients (service requesters) and target users (service providers) in online social networks. Among existing paths finding a trust path for trustworthy services is a vital job. Also, in many previous methods, such as ant colony optimization algorithm (ACO) load balancing among target users is inefficient. Therefore, in this paper we propose an inverted ant colony optimization algorithm to find a reliable path along with improving load balancing among the target users. The inverted ant colony optimization algorithm is a diversity of the basic ant colony optimization algorithm in which, the updated pheromone has a reverse effect on the selected path by the ants. Finally, we simulate the proposed method by using the original experimental dataset and evaluate the proposed method in terms of load balancing, waiting time and execution time in comparison with the ant colony optimization algorithm. The obtained results are very promising.

Keywords: Social networks; Trust; Inverted ant colony optimization algorithm; Load balancing; Waiting time

1. Introduction

Internet is one of the main telecommunication facilities utilized by all people around the world [1−4]. Also, in all of the circumstances, it is impossible not to use the concept of big data [5,6]. Social networking is the practice of developing the number of one's business and/or social contacts by creating connections through individuals [7]. Online Social Networks (OSNs) are becoming more and more famous and have been utilized as the means in a variety of applications, like employment, CRM and e-Commerce [8]. Online social networks are increasingly being utilized as places where communities gather to change information, form opinions, and collaborate in response to a happening [9]. In the last few years, many online social networks, such as Facebook and Viber, have spread out around the world and the participants in such kinds of social networks can have a great number of claimed friends [10]. More and more people utilize online social

* Corresponding author.
E-mail address: stu.saied.asghari@iaut.ac.ir (S. Asghari).
Peer review under responsibility of University of Kerbala.

http://dx.doi.org/10.1016/j.kijoms.2017.05.004

networks to share their activities and make friends [11]. With the current popularity of online social networks, more and more users are joining and more and more information is distributed through social network services [12]. Social network services (SNS) focus on building online communities of people who share interests and/or activities, or who are interested in prospecting the interests and activities of others [13]. Online social network services, such as Instagram, Facebook, and Twitter, have experienced exponential growth in membership in recent years [14]. The important kinds of social network services include directories of some categories meant to connect friends and recommended systems linked to trust [15]. Social network services, like Epinions,[1] permit their users to represent their judgment about any type of product [16].

With the growth of social network services, the need for identifying trustworthy people has become an important concern in order to protect the users' great amounts of information from being misused by unreliable users [17]. How is it possible to identify reliable users within the chain of connections and rebuff unreliable users from accessing the network and misusing information? [18]. To address these questions, “trust” is an invaluable belief in social network services [17]. Trust and friendship among users is a type of social relationship resulted from common interests among users [19—22]. Trust helps reconnoiter users with whom we can communicate, share information, and form friendships. There have been some attempts to create mathematical definitions of trust [23,24] like, the degree of subjective notion about the behaviors of (information from) a specific entity [25]. The expectation that a service will be provided or a commitment will be fulfilled [26]. Trust in a person is a commitment to a task based on a belief that the next tasks of that person will lead to a good outcome [27]. Trust inference, which aims to infer a trustworthy score from the trustor to the trustee in the underlying social network is a primary task in many real world applications [28] containing e-commerce [29], peer-to-peer networks [30], and mobile ad hoc networks [31]. In addition, trust in online social networks has three essential characteristics: transitivity, asymmetry, and personalization [32]. Also “referral trust” and “functional trust”, which were first proposed by Jøsang are distinguished [33].

[1] www.Epinions.com.

Due to the increasing of popularity and usage of social networks in recent years, social network optimization (SNO) is introduced as an optimization algorithm to solve different challenges in recent years. This method is proposed as a population based algorithm inspired to the social network knowledge sharing and decision making process. This method first introduced in Ref. [34] in a simplistic way and further developed to enhance its performance, essentially built as a population based approach inspired to the social network knowledge sharing and emulating the decision making process recently introduced by these networks. In this algorithm, in fact, each individual indicates a social network member characterized by a proper social environment (a specific position in the solution space), a proper character, a personal reputation recognized by his group and a personal interest which can be compared to a sort of taste (or liking) shared among his relational network. The personal interest can be seen as preferred direction in the space domain due to both stronger and weaker characters and particular opinion leaders. The main characteristic of the SNO are status, memory, ranking groups and influencers.

In this paper, we address the trust challenge in social networks along with improving load balancing among the target users by proposing an inverted ant colony optimization algorithm. The proposed method has an important effect on improving load balancing with the help of an updated pheromone. After pheromone update in each iteration, ants in the next iteration try to satisfy client's requirements with by using new target users. The primary objectives of the proposed algorithm are as follows:

1. We try to establish load balancing among target users or service providers.
2. We try to reduce the waiting time of the clients or service requesters to receive requirements.

In the remainder of the paper, Section 2 surveys related works. The problem definition is offered in Section 3. We present the proposed method in Section 4 and the results of the proposed method are described in Section 5. Finally, Section 6 concludes this paper and suggests future works.

2. Related work

In this section, some of the primary methods for evaluating trust in social networks are presented and analyzed.

Please cite this article in press as: S. Asghari, K. Azadi, A reliable path between target users and clients in social networks using an inverted ant colony optimization algorithm, Karbala International Journal of Modern Science (2017), http://dx.doi.org/10.1016/j.kijoms.2017.05.004

Chen et al. [35] have proposed implicit social behavioral graph to evaluate trust in mobile social networks. In this paper, the trust value was computed along a relationship path as the product of the trust values of the links on that path. There may be more than one path between two indirect neighbor users, and each path has its own trust value. In other words, "if *i* trusts *j*, and *j* trusts *k*, then *i* should have a somewhat more positive view of *k* based on this knowledge. In this paper, the implicit social behavioral graph, *i.e.*, ego-*i* graph formed by users' contacts was described, and proposed an algorithm for initiating ego-*i* graph. These relationships were rated to form a dynamic contact rank, which enabled users to evaluate the trust values between users within the context of MSNs. Then, group-based trust values were calculated according to the level of contacts, interaction evolution, and users' features. Based on a group-based trust, a cluster trust was obtained by the aggregation of inter group-based trust values.

Jiang et al. [36] have presented a novel trust framework to address the issue of "Can Alice trust Bob on a service?" in large online social networks (OSNs). In this paper, the focus is on generating small trusted graphs for large OSNs, which can be used to make previous trust evaluation algorithms more efficient and practical. This paper illustrates how to preprocess a social network (PSN) by developing a simple and practical user-domain-based trusted acquaintance chain discovery algorithm through using the small-world network characteristics of online social networks and taking advantage of "weak ties". Then, proposed how to build a trust network (BTN) and generated a trusted graph (GTG) with the adjustable width breadth-first search algorithm.

Al-Oufi et al. [17] have presented a method to evaluate trust in online social networks. In this study, the extended Advogato trust metric that facilitates the identification of trustworthy users associated with each individual user is presented. By incorporating the strength of social relationships, the capacity of a target user is recursively diffused throughout his/her personal network. Based on the capacity propagation, this paper also proposes the capacity-first maximum flow method capable of finding the strongest path pertinent to discovering an ordered set of reliable users and preventing unreliable users from accessing personal networks. Experimental results showed that this method has advantages over the existing representative methods in terms of both the discovery of reliable users and the preventability of unreliable users.

Yao et al. [28] have proposed sub graph extraction to address the trust challenge. The core of the presented algorithm includes two stages: path selection and component induction. The outputs of both stages can be utilized as an intermediate step to speed up the diversity of existing trust inference approaches. The experimental evaluations on real graphs illustrate that the presented algorithm can accelerate existing trust inference algorithms, while maintaining high accuracy. In addition, the extracted sub graph provides an intuitive way to interpret the resulting trustworthiness score.

Li and Bonti [11] have proposed T-OSN approach to evaluate trust. In this article, two main factors, Degree and Contact Interval are taken into consideration, which provide a new trust evaluation model (T-OSN). T-OSN is aimed to solve how to evaluate the trust value of an OSN user, which is also more efficient, more reliable and easy to implement. Based on the research performed, this model can be utilized in wide range, such as online social network (OSN) trust evaluation, mobile network message forwarding, ad hoc wireless networking, routing messages on the Internet and P2P file sharing network. The T-OSN model has the following explicit advantages compared to other trust evaluating approaches. First of all, it is not based on the attributes of traditional social networks, such as, distance and the shortest path.

Mohamadi-Baghmolaei et al. [37] have presented a Trust based Latency aware Influence Maximization model, abbreviated as TLIM, which selected the most influential nodes in social networks with considered time and trust simultaneously. In this article, for the first time, trust is studied in classic IC model and also both time and trust factors are jointly considered to influence the maximization problem. The important advantages of this article are listed as follows: first, extended the classic IC model to contain time and trust simultaneously, which is more applicable in existing social networks. Second, the most influential nodes are located in social networks considering time and trust together. But, considered way for finding the most influential nodes is sample and there is a no proper execution time.

Nuñez-Gonzalez et al. [38] have proposed an approach that utilizes reputation features for trust prediction in social networks. Trust prediction can be achieved by the application of machine learning algorithms applied to reputation features, which are extracted from the available trust information presented by witness users. Validation results have been presented in two approaches, a naïve selection of

Please cite this article in press as: S. Asghari, K. Azadi, A reliable path between target users and clients in social networks using an inverted ant colony optimization algorithm, Karbala International Journal of Modern Science (2017), http://dx.doi.org/10.1016/j.kijoms.2017.05.004

reputation features, and a probabilistic model of these features. In this approach, it is attempted to improve results applying a SMOTE method do not improved the overall accuracy, but provided improvements on the prediction of the minority class. The presented method is not compared with any method that is a disadvantage.

Aghdam and Navimipour [39] have presented a new framework to choose the opinion leaders in online communities. The framework utilizes the trust relationship between the users and evaluates the total trust value (TTV) of essential opinion leaders. To select opinion leaders in the social network, at first, there must be access to the relationships between users and datasets of the social network. At first, incorrect data are removed and then trust relationships are obtained to choose opinion leaders. Incorrect data includes removing self-trust statements, removing duplicate comments and removing troll's comments. Then, opinion leaders are selected. According to the obtained results, the presented framework provides better results in the social network marketing (SNM) campaigning in comparison to top in-degree method, top out-degree method, top centrality method and hybrid IO-degree method.

Sanadhya and Singh [40] have presented an ant colony algorithm to evaluate trust in online social networks. Today's social networks, which have lots of service requesters and service providers, connect to each other with many different paths. Among 'N' number of paths finding a trust path for trustworthy services is a major task. In this article, with the help of the ant colony optimization (ACO) the trust path and the trust cycle are calculated. Hence an algorithm (Trust-ACO) to calculate trust in online social networks is presented. Trust calculation is based on the probabilistic trust rule, social intimacy pheromone. In this approach, the load balancing among the service providers is inefficient. Furthermore, the proposed method is not compared with any other method.

3. Problem definition

In this section, at first, the original database along with available structure for the proposed method is described and then the objectives of the proposed optimization are introduced.

3.1. The original database along with available structure for the proposed method

The Epinions site is a social web service where users prepare reviews of products of any type, ranging from music up to perfumes or construction hardware. These reviews are the basis for the establishment of trust relations between users. Trust is a binary variable taking values in the range $\{1, -1\}$: a truster user can select to trust (1) or distrust (-1) to the trustee. Negative trust values are not published in the web service, but the anonymized dataset provided for experimentation, which is available to the public,[2] contains negative trust values. This dataset has 841,372 data samples. Regarding class distributions, the database is unbalanced: 85.3% of instances are positive trust (717,667 triplets), while 14.7% of instances are negative trust (123,705 triplets). Each sample is a triplet (A, B, t_{AB}) composed of two user indexing numbers (no personal data of any form is included) and the binary trust value of the first user on the second user. Also, for the work in this paper, the trust triplet (A, B, t_{AB}) is built from three attributes: voting user, voted user and vote value. Therefore, trust relations define a directed graph, with weighted edges. If User A votes to User B, a directed edge of User A to User B will be drawn.

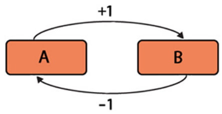

Fig. 1. The structure of the database in Epinions social network.

In the following, the structure of the database in Epinions social network is shown in Fig. 1 that is composed of triplet (A, B, t_{AB}). Fig. 1 depicts that this dataset contains two participants (A, B). User A has a positive trust to user B and user B has a negative trust to user A. In the proposed method, the negative trust instance is ignored and the focus is only on the positive trust instance. In other words, we consider only one out of two possible vote values: 1 (support) and -1 (oppose) and ignore the vote "-1". In fact, we only use trust relations like (A, B, +1). This database has previously been used in Refs. [41,42].

3.2. The objectives of the optimization

In the following, the objectives of the proposed algorithm are described. The trust challenge is evaluated among different users in social networks in this paper. One of the main problem in previous papers was imbalanced in workload among users that could satisfy

[2] http://www.trustlet.org/epinions.html.

client requirements. Therefore, in this paper, we decide to address this problem. The proposed solution is using the inverted ant colony optimization algorithm. In the inverted ant colony optimization algorithm, the smell of pheromone has reverse effect on the selected paths by the ants. Therefore, after pheromone updates in each iteration, ants in the next iteration try to satisfy client's requirements using new target users. Furthermore, when this work is done, the waiting time of the other clients to receive requirements can be reduced. Therefore, the primary objectives of the proposed algorithm are as follows:

- We try to establish load balancing among target users.
- We try to reduce the waiting time of the clients to receive requirements.

4. Proposed algorithm

In this section, we propose the inverted ant colony optimization algorithm for evaluating trust in social networks. Then, we present a case study and illustrate the performance of the proposed method.

4.1. Inverted ant colony optimization algorithm for evaluating trust

The inverted ant colony optimization algorithm is a variation of the basic ant colony algorithm that converts its logic by inverting the attraction of ants towards pheromones into a repulsion effect. In this method, the pheromone scent will create a repulsion effect for the other ants instead of an attraction effect [43]. The inverted ant colony optimization algorithm performs in two phases. In the first phase, the selection operation is conducted among different users, and the user that has a more probability is selected. In the second phase, the pheromone update operation is conducted and causes the next ants in the next iteration to traverse the new paths and visit the new target users.

In the presented work, there is a graph in the form of $G = (V, \ E)$ in which V represents the nodes and E represents the edges. Nodes refer to the users, and edges refer to the relations among users. There is a one client in the proposed method that refers to the start node. All ants are in the start node and begin the graph traversal of this node. In each iteration, all ants select the next user to traverse according to Eq. (1) and then pheromone update according to Eq. (2). After each iteration, ants examine that the client requirement is satisfied or not. If the requirement is satisfied, the ant exits from the system and calls a new ant. If the requirement is not satisfied, the ant continues to traverse the graph until the requirement of the client is satisfied. If the calculated probability for a user is zero, the user will not be selected by any ant. If two users exist in the same condition, one of them will be selected randomly.

In Eq. (1) t is the current iteration, $\tau_{kj}(t)$ is the pheromone on edge e_{kj} in which at first, the pheromone value for all edges is one and η_{kj} is the heuristic information on edge e_{kj}. α and β are values that show the amount of the impression of the pheromone and heuristic information respectively. The value of these parameters is considered one.

$$p^i_{kj}(t) = \frac{\left[\tau_{kj}(t)^{\alpha}\right]\left[\eta_{kj}(t)^{\beta}\right]}{\sum_{l=1}^{n} p^i_{kl}(t)\left[\tau_{kl}(t)^{\alpha}\right]\left[\eta_{kl}(t)^{\beta}\right]} \tag{1}$$

Eq. (2) indicates the updated pheromone that acts in each iteration and k is the number of the traversed user.

$$\tau(t+1) = \tau(t) \times \frac{1}{k} \tag{2}$$

Eq. (3) indicates the heuristic relation that refers to the adopted metric to measure the strength of trust relationships [44] based on the structural and social similarities between the two users. The number of users who are trusted by both users, i and j to the number of users that are trusted by either user i or user j, but not both.

$$\eta_{kj} = \frac{|out - degree(i) \cap out - degree(j)|}{|out - degree(i) \cup out - degree(j)|} \tag{3}$$

The waiting time refers to the time that each client demand to expend to obtain the requirement and it is assumed that 1 ms time is needed to use the resource of each target user that contains the requirement. It is assumed that if a common target user is needed to respond the client demands, this target user will be accessed to the client demands in order. Thus, other client demands will wait for 1, 2, 3 and … ms, respectively. Therefore, to calculate the waiting time, the number of common demands is important and crucial. It is assumed that each ant acts a client demand. Also, the waiting time for each client demand is obtained according to Eq. (4) in which N is the number of users, t is the time that each client demand to spend to use user resources and f is the waiting time for each client demand. Eq. (5) describes the total time that the client demands to spend to meet its needs and R is the number of demands, f is the obtained waiting time for each client demand and F is the waiting time for all client demands.

$$f = \sum_{i=1}^{N} t_{u_i} \tag{4}$$

$$F = \sum_{i=1}^{R} f \tag{5}$$

In the following, the presented algorithm for evaluating trust in online social network is provided.

The presented algorithm of the inverted ant colony optimization method is as follows:

Algorithm 1: Inverted ant colony optimization algorithm
Input:
$U = \{u_1, u_2, \ldots, u_n\}$: set of users, where $u_{i=} \{r_1, r_2, \ldots, r_k\}$ contains a set of services;
N: Number of users;
M: Number of ants;
Demand-set: client demands;
Output:
Optimal target user;
Begin
While (i = 1 **to** M **do)**
Begin graph traversal from the client;
Is selected a next user u_j according to the probability p^i_{kj} defined in (3-1);
Update the pheromone on edge e_{kj} according to (3-2);
Examines whether the client requirement is satisfied or not;
If satisfied, the ant exits from the system, otherwise the next user is selected;
End while;
The output of each ant is the target user that has been visited.
End

4.2. A solved example

In this section, an example about the effectiveness of the proposed approach in terms of load balancing among the target users and the waiting time is solved and evaluated. The experimental dataset for this section is extracted from the Epinions original database. Of course, some of the relationships are generated randomly. In this example, there are ten users that are shown in Table 1. Each of the users has unique ID. Table 1 has two columns that the first column refers to users ID that in each row there is only one user. The second column refers to users that are trusted to them. In other words, the trust value of related user in the same row is +1. In fact, the users that are trusted to them are neighbors of available users in the same row and the first column. For example, in the first row, the user 2086 has a positive trust to 15858 and 94 users and trust value of 2086 user to them is +1.

Table 1
The experimental dataset.

User ID	Users that are trusted to them
2086	15858, 94
94	15858, 44255, 13522, 580
15858	94, 30152, 974
44255	580
13522	44255, 580, 2256
30152	974, 18945
580	2256
2256	974
974	Empty
18945	974

According to the available information in Table 1, one graph that is shown in Fig. 2 is generated. To simplify the graph design, a characteristic is assigned to each user as shown in Table 2. In this example, it is assumed that the user u_1 is a client and the users u_4, u_5, and u_6 are target users. As is evident in Table 1, user u_9 (974) does not trust anybody. It is assumed that there are six ants to traverse the graph and satisfy the client requirements.

In Fig 2, all ants are in the start node and want to begin the graph traversal. The start node is the client or in other words is user u_1. The first ant begins the graph

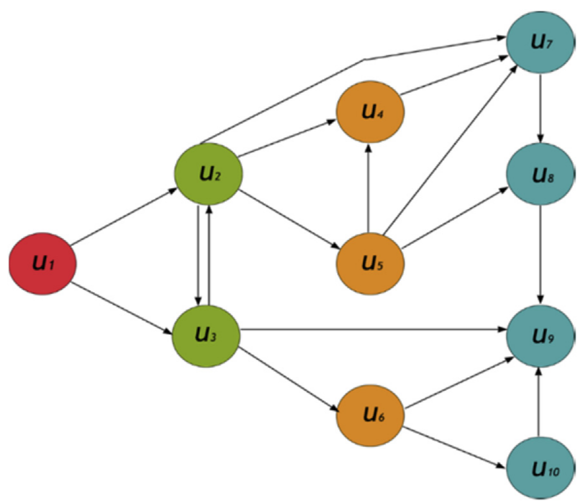

Fig. 2. The designed graph for the mentioned example.

Table 2
The symbols of the users.

User ID	Symbol
2086	u_1
94	u_2
15858	u_3
44255	u_4
13522	u_5
30152	u_6
580	u_7
2256	u_8
974	u_9
18945	u_{10}

traversal and must select from among u_2, u_3 users that the user u_3 has more probability than the user u_2. Therefore, u_3 will be selected and then the pheromone of the edge e_{13} is updated. Due to the fact that user u_3 is not the target user or in other words, the requirement of the client is not satisfied, the first ant continues the graph traversal. In the following, the first ant selects u_6 among u_2, u_6, and u_9 users because the user u_6 has more probability and the pheromone of edge e_{36} is updated. Because the user u_6 is a target user, or in other words the requirement of the client is satisfied, the first ant comes out of the system and calls the second ant.

The second ant begins the graph traversal and selects the user u_2 that has more probability than the user u_3 and then the pheromone of the edge e_{12} is updated. Because the user u_2 is not the provider or in the other words, the requirement of the client is not satisfied, the second ant continues the graph traversal and selects the user u_5 among u_3, u_4, u_5, and u_7 users. Then, the pheromone of the edge e_{25} is updated. Because the user u_5 is the target user, or in other words the requirement of the client is satisfied, the second ant comes out of the system and calls for the third ant.

The third ant starts the graph traversal and chooses user u_3 and in the following, choses user u_6 among u_2, u_6, and u_9 users because user u_6 has more probability and the pheromone of the edge e_{36} is updated. In the following, The fourth, fifth and sixth ants select u_4, u_6, and u_5 target users respectively.

5. Results

In this section, the experimental environment and the obtained results of the comparison of the inverted ant colony optimization to the ant colony optimization are described and presented. The experiment dataset is extracted of the Epinions original database available in http://www.trustlet.org/epinions.html. The experiments are evaluated using of java programming language in the Eclipse environment. The experiments on the different number of users are conducted and the load balancing, waiting time and execution time for any number of users are evaluated. The obtained results for each parameter are displayed separately in Sections 5.1, 5.2 and 5.3 respectively. The experiment variables and considered values for them are illustrated in Table 3. The experiment variables consist of the number of users, the number of neighbor users, the number of needed services, the number of target users and the number of ants. When the number of users is low, the considered values for variables are clear but when the number of users increases, the considered values for some of the variables are generated randomly.

Table 3
The experiment environment.

Experiment variables	Values
Number of users	10, 100, 1000, 17000
Number of neighbors	0, 1, 2, …
Number of needed services	1
Number of target users	1, 2, 3, …
Number of ants	6, 10, 100, 1000, 17000

5.1. Load balancing

The obtained results for the mentioned example in the previous section for the proposed method in comparison with the ant colony optimization algorithm in terms of load balancing are evaluated. It is assumed that there are six ants in the mentioned example. The obtained results in Fig. 3 show that the proposed method has a better load balancing in comparison to the ant colony optimization regarding the percentage of used by each target users. The proposed method uses three target users to satisfy the client requirement but the ant colony optimization algorithm uses only one

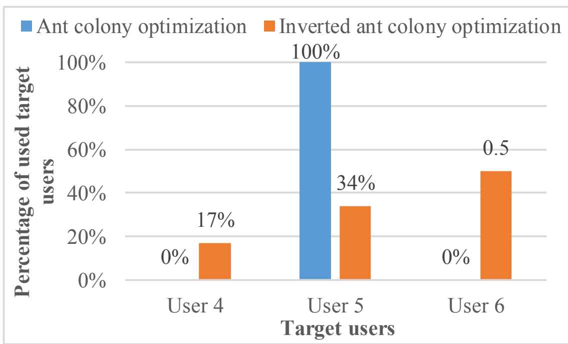

Fig. 3. The comparison of the algorithms in terms of percentage of used by each target user.

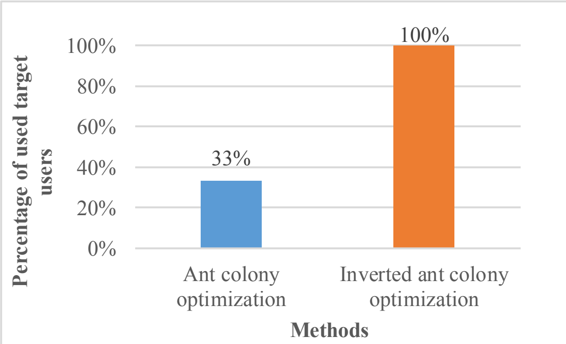

Fig. 4. The comparison of the algorithms in terms of the overall percentage of used target users.

target user. The obtained results in Fig. 4 show the overall percentage of used target users. It is clear that, the performance of the proposed method is better than that of the ant colony optimization algorithm in terms of load balancing.

In the following, the simulation is performed for the different numbers of the users and the results are evaluated. Because of the high number of users, information is generated randomly such as direct neighbors, therefore in each iteration different results might be obtained, but certainly the performance of the proposed algorithm will be better in comparison with the ant colony optimization algorithm. The number of target users is 4, 10, 100, and 500 respectively for 10, 100, 1000, and 17000 users. The number of ants is considered equal to the number of the target users. It is clear that, with the increase in the number of users, the performance of the proposed method also increases in terms of load balancing in comparison with the ant colony optimization algorithm. The obtained results are shown in Fig. 5.

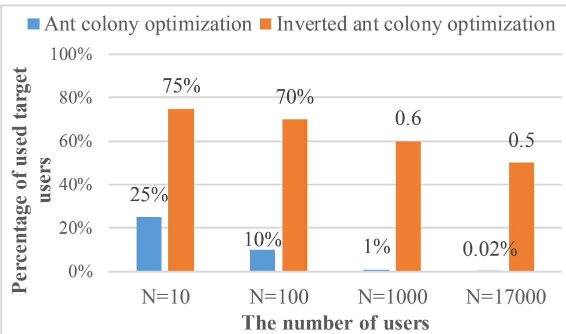

Fig. 5. The comparison of the algorithms in terms of the percentage of target users.

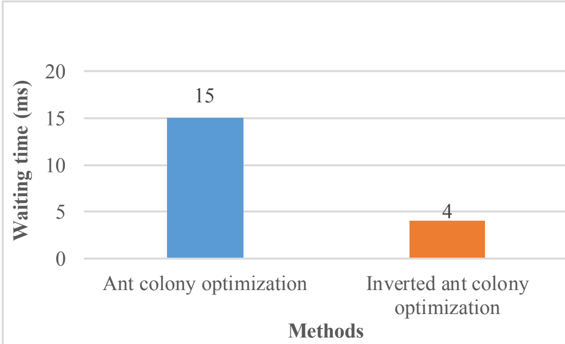

Fig. 6. The comparison of the algorithms in terms of the waiting time for the mentioned example.

5.2. Waiting time

The obtained results for the mentioned example in the previous section for the proposed method in terms of the waiting time in comparison with the ant colony optimization algorithm are evaluated. At first, the waiting time for the mentioned example in the previous section is evaluated. It is assumed that 6 common demands by the client are given to the system. The obtained results in Fig. 6 show that the client demands in the proposed method are rapidly processed. In other words, demands in the inverted ant colony optimization algorithm spend lower time to be processed. In the following, the waiting time is calculated for any number of users. It is assumed that 4, 10, 70, and 300 common demands in order for 10, 100, 1000 and 17000 users are given to the system. The Dijkstra algorithm is added for evaluating the performance of the proposed method in terms of the waiting time. For 17,000 users, due to the easiness of calculation for the ant colony optimization and Dijkstra algorithms, the waiting time is calculated for 100 common demand. By increasing the number of target users, the performance of the proposed method in terms of the waiting time increases in comparison with the ant colony optimization and Dijkstra algorithms. The performance of the ant colony optimization method and Dijkstra algorithm is equal. The obtained results are shown in Table 4.

5.3. Execution time

Finally, the inverted ant colony optimization algorithm, the ant colony optimization algorithm and the Dijkstra algorithm are compared together in terms of the execution time. The Dijkstra algorithm is added for evaluating the performance of the proposed method in

Please cite this article in press as: S. Asghari, K. Azadi, A reliable path between target users and clients in social networks using an inverted ant colony optimization algorithm, Karbala International Journal of Modern Science (2017), http://dx.doi.org/10.1016/j.kijoms.2017.05.004

Table 4
The comparison of the algorithms in terms of the waiting time.

Number of users	Algorithms		
	Ant colony optimization	Inverted ant colony optimization	Dijkstra algorithm
10	6	1	6
100	45	3	45
1000	2415	10	2415
17000	5050	50	5050

terms of the execution time. The obtained results in Table 5 indicate that the performance of the proposed algorithm is weaker than the ant colony optimization algorithm. But, the performance of the proposed method is better than the Dijkstra algorithm. The obtained results for the mentioned example in Section 4.2 show that the obtained execution time for the proposed algorithm is 4 ms more than the ant colony optimization algorithm. It is clear that, with the increase in the number of users, the performance of the proposed algorithm becomes weaker in comparison to the ant colony optimization method. This is a disadvantage of the presented method. However, the obtained execution time for the proposed method is lower than the Dijkstra algorithm.

6. Discussion

In this paper, we have developed a method called the inverted ant colony optimization algorithm to select the reliable target users in the social networks. The main objective of the proposed algorithm was to improve the load balancing among the target users. Furthermore, we could reduce the waiting time of the client demands to satisfy the requirements. The number of users, the number of neighbors and the number of ants had an important effect on the obtained results. The obtained results indicated that the proposed algorithm had a better performance in terms of the load balancing among the target users and the waiting time in comparison with the ant colony optimization. But, unfortunately the execution time of the proposed algorithm was more than that of the ant colony optimization algorithm. Furthermore, the performance of the proposed method is better than that of the Dijkstra algorithm in terms of the waiting time and execution time. For the future, it is recommended that trust evaluation by other meta-heuristic algorithms be conducted and the obtained results about them be compared with the obtained results of the proposed algorithm. Also, other original databases like Wikipedia and Slash Dot can also be used.

Table 5
The comparison of the algorithms in terms of the execution time.

Number of users	Algorithms		
	Ant colony optimization	Inverted ant colony optimization	Dijkstra algorithm
10	50	54	57
100	150	170	185
1000	14400	15500	16120
17000	193800	263500	294200

References

[1] S. Asghari, N.J. Navimipour, Service composition mechanisms in the multi-cloud environments: a survey, Int. J. New Comput. Archit. Appl. (IJNCAA) (2016) 40—48.
[2] N.J. Navimipour, F.S. Milani, A comprehensive study of the resource discovery techniques in Peer-to-Peer networks, Peer-to-Peer Netw. Appl. 8 (2015) 474—492.
[3] Y. Charband, N.J. Navimipour, Online knowledge sharing mechanisms: a systematic review of the state of the art literature and recommendations for future research, Inf. Syst. Front. (2016) 1—21.
[4] N.J. Navimipour, S. Asghari, Energy-aware service composition mechanism in grid computing using an ant colony optimization algorithm, 대한전자공학회 학술대회 (2017) 282—286.
[5] S. Asghari, N.J. Navimipour, Review and comparison of meta-heuristic algorithms for service composition in cloud computing, Majlesi J. Multimedia Process. 4 (2016).
[6] Z. Tavakoli, T.J. Gandomani, Big data and its challenges, Majlesi J. Multimedia Process. 4 (2015).
[7] A.J. Singhai, The Power of Social Networking, 2013.
[8] G. Liu, F. Zhu, K. Zheng, A. Liu, Z. Li, L. Zhao, et al., TOSI: a trust-oriented social influence evaluation method in contextual social networks, Neurocomputing 210 (2016) 130—140.
[9] G. Bachi, M. Coscia, A. Monreale, F. Giannotti, Classifying trust/distrust relationships in online social networks, in: Privacy, Security, Risk and Trust (PASSAT), 2012 International Conference on and 2012 International Confernece on Social Computing (SocialCom), 2012, pp. 552—557.
[10] Y. Ruan, A. Durresi, A survey of trust management systems for online social communities—Trust modeling, trust inference and attacks, Knowledge-Based Syst. 106 (2016) 150—163.
[11] M. Li, A. Bonti, T-OSN: a trust evaluation model in online social networks, in: Embedded and Ubiquitous Computing (EUC), 2011 IFIP 9th International Conference on, 2011, pp. 469—473.
[12] D. Boyd, N. Ellison, Social network sites: definition, history, and scholarship, IEEE Eng. Manag. Rev. 3 (2010) 16—31.
[13] D.-H. Shin, The effects of trust, security and privacy in social networking: a security-based approach to understand the pattern of adoption, Interact. Comput. 22 (2010) 428 vol. 25, pp. 270—270, 2013.
[14] V. Barker, Older adolescents' motivations for social network site use: the influence of gender, group identity, and collective self-esteem, CyberPsychology Behav. 12 (2009) 209—213.

Please cite this article in press as: S. Asghari, K. Azadi, A reliable path between target users and clients in social networks using an inverted ant colony optimization algorithm, Karbala International Journal of Modern Science (2017), http://dx.doi.org/10.1016/j.kijoms.2017.05.004

[15] J.v. Brocke, D. Richter, K. Riemer, Motives for using Social Network Sites (SNSs)-An analysis of SNS adoption among students, BLED 2009 Proc. (2009) 40.
[16] A. Jøsang, R. Ismail, C. Boyd, A survey of trust and reputation systems for online service provision, Decis. Support Syst. 43 (2007) 618—644.
[17] S. Al-Oufi, H.-N. Kim, A. El Saddik, A group trust metric for identifying people of trust in online social networks, Expert Syst. Appl. 39 (2012) 13173—13181.
[18] S. Aloufi, Trust-aware Link Prediction in Online Social Networks, Université d'Ottawa/University of Ottawa, 2012.
[19] V. Liptchinsky, R. Khazankin, S. Schulte, B. Satzger, H.-L. Truong, S. Dustdar, On modeling context-aware social collaboration processes, Inf. Syst. 43 (2014) 66—82.
[20] L. Zhang, Z. Cai, J. Lu, X. Wang, Spacial mobility prediction based routing scheme in delay/disruption-tolerant networks, in: Identification, Information and Knowledge in the Internet of Things (IIKI), 2014 International Conference on, 2014, pp. 274—279.
[21] J. Gong, J. Tang, A. Fong, ACTPred: activity prediction in mobile social networks, Tsinghua Sci. Technol. 19 (2014) 265—274.
[22] R. Bredereck, J. Chen, P. Faliszewski, J. Guo, R. Niedermeier, G.J. Woeginger, Parameterized algorithmics for computational social choice: nine research challenges, Tsinghua Sci. Technol. 19 (2014) 358—373.
[23] X. Gai, Y. Li, Y. Chen, C. Shen, Formal definitions for trust in trusted computing, in: Ubiquitous Intelligence & Computing and 7th International Conference on Autonomic & Trusted Computing (UIC/ATC), 2010 7th International Conference on, 2010, pp. 305—310.
[24] D. Xiu, Z. Liu, A formal definition for trust in distributed systems, in: International Conference on Information Security, 2005, pp. 482—489.
[25] K. Cook, Trust in Society, Russell Sage Foundation, 2001.
[26] L.J. Hoffman, K. Lawson-Jenkins, J. Blum, Trust beyond security: an expanded trust model, Commun. ACM 49 (2006) 94—101.
[27] Q. Fu, Trust, Social Capital, and Organizational Effectiveness, Virginia Polytechnic Institute and State University, 2004.
[28] Y. Yao, H. Tong, F. Xu, J. Lu, Subgraph extraction for trust inference in social networks, in: Encyclopedia of Social Network Analysis and Mining, Springer, 2014, pp. 2084—2098.
[29] L. Xiong, L. Liu, Peertrust: supporting reputation-based trust for peer-to-peer electronic communities, IEEE Trans. Knowl. Data Eng. 16 (2004) 843—857.
[30] S.D. Kamvar, M.T. Schlosser, H. Garcia-Molina, The eigentrust algorithm for reputation management in p2p networks, in: Proceedings of the 12th International Conference on World Wide Web, 2003, pp. 640—651.
[31] S. Buchegger, J.-Y. Le Boudec, A Robust Reputation System for Mobile Ad-hoc Networks, 2003.
[32] J.A. Golbeck, Computing and Applying Trust in Web-based Social Networks, 2005.
[33] A. Jøsang, R. Hayward, S. Pope, Trust network analysis with subjective logic, in: Proceedings of the 29th Australasian Computer Science Conference, Vol. 48, 2006, pp. 85—94.
[34] R. Albi, F. Grimaccia, M. Mussetta, A. Niccolai, R. Zich, A new algorithm for antenna optimization in aerospace applications: the sounding rocket test case, in: Electromagnetics in Advanced Applications (ICEAA), 2013 International Conference on, 2013, pp. 1537—1540.
[35] S. Chen, G. Wang, W. Jia, Cluster-group based trusted computing for mobile social networks using implicit social behavioral graph, Future Gener. Comput. Syst. 55 (2016) 391—400.
[36] W. Jiang, G. Wang, J. Wu, Generating trusted graphs for trust evaluation in online social networks, Future Gener. Comput. Syst. 31 (2014) 48—58.
[37] R. Mohamadi-Baghmolaei, N. Mozafari, A. Hamzeh, Trust based latency aware influence maximization in social networks, Eng. Appl. Artif. Intell. 41 (2015) 195—206.
[38] J.D. Nuñez-Gonzalez, M. Graña, B. Apolloni, Reputation features for trust prediction in social networks, Neurocomputing 166 (2015) 1—7.
[39] S.M. Aghdam, N.J. Navimipour, Opinion leaders selection in the social networks based on trust relationships propagation, Karbala Int. J. Mod. Sci. 2 (2016) 88—97.
[40] S. Sanadhya, S. Singh, Trust calculation with ant colony optimization in online social networks, Procedia Comput. Sci. 54 (2015) 186—195.
[41] P. Massa, P. Avesani, Controversial users demand local trust metrics: an experimental study on epinions. com community, in: Proceedings of the National Conference on Artificial Intelligence, 2005, p. 121.
[42] P. Massa, P. Avesani, Trust metrics on controversial users: balancing between tyranny of the majority, Int. J. Semantic Web Inf. Syst. (IJSWIS) 3 (2007) 39—64.
[43] J.C. Dias, P. Machado, D.C. Silva, P.H. Abreu, An inverted ant colony optimization approach to traffic, Eng. Appl. Artif. Intell. 36 (2014) 122—133.
[44] Y.S. Kim, V.L. Tran, Assessing the ripple effects of online opinion leaders with trust and distrust metrics, Expert Syst. Appl. 40 (2013) 3500—3511.

Please cite this article in press as: S. Asghari, K. Azadi, A reliable path between target users and clients in social networks using an inverted ant colony optimization algorithm, Karbala International Journal of Modern Science (2017), http://dx.doi.org/10.1016/j.kijoms.2017.05.004