

“Friend is Treasure”: Exploring and Exploiting Mobile Social Contacts for Efficient Task Offloading

Panlong Yang, *Member, IEEE*, Qingyu Li, *Student Member, IEEE*, Yubo Yan, *Student Member, IEEE*, Xiangyang Li *Fellow, IEEE*, Yan Xiong *Member, IEEE*, Baowei Wang, Xingming Sun

Abstract—In this study, we investigate the task offloading issue in mobile social networks. Although the ‘d-choice’ paradigm in ‘ball and bin’ theory [1] had shown the power of random choice in load balancing with random walk model, its performance could be fairly poor when real trace data sets are concerned. According to our preliminary evaluation results with ‘MobiClique’ [2], the ‘d-choice’ scheme could not achieve well balanced allocations in real trace data set. Nevertheless, it would bring fundamental challenges to task reassignment in the following aspects: First of all, some of the friendships are relatively stable, which would lead to a more imbalanced task assignment, even if the ‘d-choice’ scheme is applied for balancing. Secondly, some users would meet quite infrequently, which could inevitably lead to intolerable time delay and unfair task allocations.

In tackling with these difficulties, we revisit the real data sets [2] [3] [4] for exploring the contact property among users. We find that, the frequently met users could be leveraged for efficient task execution due to higher task priority. To this end, we propose the ‘iTop-K’ algorithm, leveraging the basic concept, *i.e.*, ‘your friends are more powerful than you’ [5], which encourages mobile users to assign tasks among intimate friends instead of pure random assignment. With careful selections of ‘Top-K’ friends, we achieve balanced load and guaranteed performance at the same time.

Experimental studies verify our scheme and show the effectiveness with three typical data trace sets, including ‘MobiClique’ [2], *et al.*. In these typical networking scenario [2] [3] [4], ours outperforms the conventional random choice scheme up to 15 \times , and the social relationship assignment without priority method up to 9 \times . Moreover, the ‘Top-K’ scheme could be adaptive even when no intimate friends are available. By scaling the ‘K’ factor to larger values, our scheme outperforms random assignment, and could be inspiringly close to the optimal solution. In summary, ours could effectively explore the social relationship and leverage it for efficient task assignment, which would further encourage more mobile users to work together under the rule of social contacts.

Index Terms—Task allocation, Traffic Balancing, Mobile Social Network

Panlong Yang and Yan Xiong are with the Department of Computer Science and Technology, University of Science and Technology of China (E-mail:panlongyang@gmail.com, yxiong@ustc.edu.cn)

Qingyu Li is with the China Satellite Maritime Tracking and Control Department, Jiangyin, China, (Email:liqingyu711@gmail.com)

Yubo Yan is with the PLA University of Science and Technology, Nanjing 21007, China (E-mail:yanyub@gmail.com).

Xiangyang Li is with the School of Software and TNLIST, Tsinghua University, China, and the Department of Computer Science, Illinois Institute of Technology, USA. (E-mail:xiangyang.li@gmail.com).

Panlong Yang, Baowei Wang and Xingming Sun are with the College of Computer and Software, and Jiangsu Energy Center of Network Monitoring, Nanjing University of Information Science and Technology, Nanjing, China (Email:wbw.first@163.com, sunnudt@163.com)

This research is partially supported by NSF China under Grants No. 61232018, 61272487, 61232016, U1405254, 61173136, NSF ECCS-1247944, and CMMI 1436786, PAPD fund, and CICAET.

I. INTRODUCTION

Nowadays’ smart phone is the main tool for us to access any kind of information from fixed or mobile ad hoc terminals. When mobile services are provided to the mass [6], the self-organized computing, communication, and storage services are main concerns for most of the mobile users. Specifically, organizing distributed participants’ sensing data as well as services in ad hoc way will need distributed coordination in sensing and computing, together with relatively low communication overhead. In particular, participatory sensing solutions enable more users to share their sensing data in non-invasive manner [7] [8] [9], which encourage collaborations among massive number of mobile participants [10]. Specifically, in data intensive mobile computing environment, more concrete collaborations are desperately needed. Mobile users with heavy tasks need to offload their tasks to the contacted friends for efficient execution. And users could offload their tasks to the contacted friends with higher energy and computing efficiency.

Previous studies fail to achieve perfect load balancing for two reasons: (1) For centralized task reassignment systems, the tasks are sent to the remote cloud instead of nearby users, which would inevitably cost too much bandwidth, and could not leverage the resources in cloud effectively [7] [11] [5]. (2) Even for pure distributed networks, the balanced task offloading is still hard to achieve [7] [12] because of the transitory inter-contact and highly dynamic queueing length [10] [13] [14].

In this work, we revisit the load balancing scheme according to ‘ball and bin’ theory [1], which has been proposed as a simple but effective approach for traffic balancing in distributed networks [11] [12]. Leveraging the real-trace data ‘MobiClique’ [2], we make some basic trace-driven simulations. We find that such scheme is not applicable, especially when real mobility traces are applied. The reason is, for real trace data, social relationship dominates users’ contact duration. For users with close relationship or similar interests, the contact durations could be extremely large. Due to the non-uniformly distributed contact duration, random assignment leads to imbalanced user load distribution. Also, for some users without such intimate relationship, task assignment in ‘ball and bin theory’ will inevitably lead to low efficiency. In summary, the root reason is, conventional ‘ball and bin’ scheme failed to explore and exploit the inherent social relationships among mobile users effectively.

We propose a new concept for balanced task offloading algorithm design, that is, ‘Your friends are more powerful

than you.’ Considering the social relationship among users, we propose an ‘*i*Top-K’ approach in building a robust and efficient task offloading scheme.

Such design is based on two fundamental observations in our preliminary experimental study.

- 1) Social contacts are relatively stable and could be used for task assignment [15] [16]. To this end, there are closer ‘social relationships’ [17] among users with shorter contact interval, which will lead to higher task execution priority as well as efficiency [5] [18].
- 2) The statistical results in social contacts can also be used for further optimization. For example, users are willing to assign their tasks to ones with higher social contacts frequency.

Leveraging these two basic but important findings, we propose a lightweight and distributed algorithm to enable task offloading according to social relationships. The contribution of our work is three-fold:

- First, we explore the social contacts, and exploit the ‘Top-K’ users with contacting frequency to improve the task offloading efficiency. In that, social contacts are used for the ‘Top-K’ mobile users, which are further leveraged for task execution priorities.
- Second, we validate ‘*i*Top-K’ with real traces, where the proposed algorithm could outperform ‘*d*-choice’ scheme [1] [19] significantly. To the best of our knowledge, this work is the first to achieve balanced task assignment in pure distributed mobile networks, especially when real trace data sets are considered for evaluation.
- Third, considering the case that, there is no intimate friends in contact window, we use a scalable ‘*K*’ scheme, where the ‘*K*’ value could be adaptively adjusted in selecting friends. The ‘*K*’ value means the number of candidate users¹ for task allocation. We find that, under the scalable ‘*K*’ scheme, ‘*i*Top-K’ significantly outperforms the basic scheme when ‘*K* = 2’, and shows improved balancing property even when ‘*K* = 16’.

The remainders of this paper are organized as follows: we review the state of art in Section II, and make a further understanding on social contacts with real trace data in Section III. Further, we present the problem formation in Section IV. After that, algorithm design is introduced in Section V. To validate our proposed scheme, extensive evaluations have been done and illustrated thoroughly in Section VI. Finally, we conclude our work in Section VII.

II. RELATED WORK

A. Random Walk

Our work relates to the efficient data transfer schemes over disruption-tolerant networks or opportunistic networks [10], which highly correlates with the random walk model. The random walk concept was proposed by Karl Pearson [20]. Note that, in mobile social network, the social relationship dominates the trajectories of the mobile users. Fortunately,

¹Here the candidate users denote the users in communication range to each other, and could successfully accomplish task offloading in contact windows.

random walks can be incorporated into mobile social network for exploring the character and opportunities in data transmissions. For instance, Newman [21] proposes the random-walk betweenness centrality. This interesting metric reasonably defines how often a node in graph is visited by a random walker between all possible mobile users. Similar to the betweenness evaluation, Noh and Rieger [22] introduce the random-walk closeness centrality metric, which measures how fast a node can effectively get a message from other mobile nodes, in the random deployed system, such as the distributed mobile social networks.

The intermittent contacts are useful for data sharing, which have been well explored and studied in variety of network settings, from military warfare [23] to disasters recovery [24]. These proposals believe that, the fixed infrastructure is unavailable or costly, even highly unreliable. With numerous cheap and distributed working nodes, more complex tasks could be accomplished successfully with proper and distributed cooperations.

B. Task allocation

Many working crowdsourcing systems are making efforts to realize this vision, in terms of designing actual platforms, providing cooperative task execution among users with similar interests and demands. In smart city sensing applications, crowdsourcing paradigms leverage the pervasive human behaviors, *e.g.*, walking, driving and shopping *etc.*, to provide a large scale urban sensing network with wider coverage in time and space domain. Even further, the social relationship, *e.g.*, the crowd gathering and migration are also important for some specific applications, *e.g.*, flu influence detection, air quality, and traffic monitoring *etc.*. The booming of smart phones speeds up the crowdsourcing based applications for urban sensing. Recently, the crowdsourcing based sensing applications are exploited to monitor the urban environment [28]–[31]. More specifically, Mun *et al.* [32] employ the customized and portable sensors on each participant to monitor the air quality of the city. And more efforts have been made for environmental monitoring with pervasive computing [30], [31] [33] [34]. For example, the constructions of noise map for the smart city are discussed in [35]. Leveraging the microphones of the participants, these works focus on implementation of the monitoring system. However, they failed to consider the task allocation efficiency, because unreliability and inaccuracy of the observations are inevitable in participatory sensing networks. To this end, more efforts are needed for efficient and cooperative task execution. One of the most fundamental requirement is to balance tasks among users, since overloaded tasks would lead to intolerable processing delay and single point of failure.

For traffic balancing schemes, few solutions have been proposed or formally addressed in dealing with the aforementioned challenges, especially in mobile networks for data delivery [36]. P. Key *et al.* [12] propose an efficient load balancing scheme for multi-path routing networks. They creatively leverage the ‘ball and bin’ theory, where traffics are assigned to paths with ‘*d*’ choices. Here ‘*d*’ is the number

of network paths. Recently, inspired by this idea, an efficient task offloading scheme has been proposed for mobile social networks [11], where tasks are assigned to the least loaded mobile users among ‘ d ’ choices, where ‘ d ’ is the number of candidate mobile users.

The main focus of this paper is load balancing in distributed crowdsourcing system. Different from previous crowdsourcing applications, we use a pure distributed computation and communication model, where users do not need to transmit any messages for centralized computation. Moreover, our major concern is how to apply the proposed task allocation scheme with real trace data, where social relationships could be effectively explored and exploited.

III. REVISITING THE ‘D-CHOICE’ SCHEME WITH REAL-TRACE DATA SET

First, we evaluate the impact of the real trace data on task offloading scheme. As it has been verified in previous studies, simple allocation schemes based on ‘ball and bin’ theory [12] [11], could effectively balance the network load without global information. We revisit this problem with real trace data [2], and find that, the conventional ‘ball and bin’ scheme could not provide balanced allocations, especially when real data traces are applied. After that, we make two basic observations on real trace data. First, the contact frequency is extremely high for some specific user pairs. Second, the social contact frequency is not uniformly distributed. These observations motivate us to explore the social relationship, and leverage it for efficient task offloading.

A. Impact of the social relationship in real trace data

The basic content of ‘ball and bin’ theory could be stated as follows. Given that, n nodes are to be thrown into n bins, where each ball is chosen to each bin uniformly and independently. The focus is the maximum loaded bin, *i.e.* the largest number of balls in all the bins, is approximately $\frac{\log n}{\log \log n}$. If the balls are thrown sequentially, and each ball is placed in the least loaded bins of $d \geq 2$, the maximum load is $\frac{\log \log n}{\log d} + \Theta(1)$ with high probability. We call this method ‘ d -choice paradigm’.

In ‘ d -choice’ paradigm, each user randomly selects ‘ d ’ users among the contacted neighbors, and assigns tasks to the least loaded one. In light of the real trace data [2], we make statistical analysis on the contact times of mobile users. The task load in each user could be available by basic communication model. When the users are in contact range, there is a probing/answering process among the contacted neighbors. To this end, the task load on each user could be shared among the contacted neighbors. Moreover, according to the basic scheme of ‘ d -choice’, only two neighbors need to be probed, and the traffic load could be relatively small. This property is favorable for bandwidth constrained crowdsourcing network. As shown in Fig. 1, the ‘ d -choice’ scheme could not balance the tasks among users effectively. Comparing with random assignment, it gets similar performance when task load is concerned. Specifically, it significantly differs from previous results evaluated under the random walk model [11].

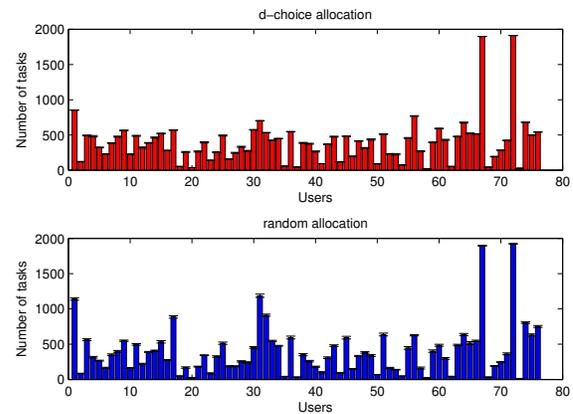


Fig. 1. Allocating tasks with d -choice paradigm

B. Basic observations

As shown in Fig. 2a, there are some extremely high contact ratio (over 1500 times during data collection period) between some specific user pairs. In mobile social networks, two persons with high social relationship would frequently meet each other. These extremely high contact frequencies are indeed existing, but not representative. As depicted in Fig. 2b, and Fig. 2c, we remove them by the following rules. First, the mitigated values should be at least 10 times higher than the average value. Second, the percentile of the mitigated values should be less than 1% of all the contacts. According to Fig. 2c, the distribution is clearly non-uniform. Such that, we could not assign each user with equal tasks. In fact, it is a ‘weighted bin’ problem in ‘ball and bin’ theory [1].

For mobile social networks, the contacts are dominated by the social relationship, instead of the stochastic features in random walk model. Taking mobile users equally will lose the valuable opportunity for improving task allocation efficiency, which is the root reason leading to poor load-balancing performance of the ‘ d -choice’ paradigm.

Inspired by these observations, we need to exploit the social relationship to enhance the task offloading schemes, where mobile users are classified according to social contact frequencies. Further more, task execution time is tightly coupled with the social relationships.

IV. SYSTEM MODEL AND PROBLEM FORMATION

A. System model

We consider a mobile social network, where n mobile users are included in a user set, denoted by $U = \{u_1, u_2, \dots, u_n\}$. Here we assume the application scenario needs cooperation to work out the given tasks in the system. The applications [25] [26] [27] could be user centric [37], providing cooperative task completion among interested users.

Each user $u_i \in U$ owns a task set $T_i = \{t_i^1, \dots, t_i^m\}$. For each task of user i , when it is reallocated to another user j , it will be processed at user j without being further forwarded to other users. We do not consider multi-hop forwarding schemes in this study. The reason is, task offloading is a

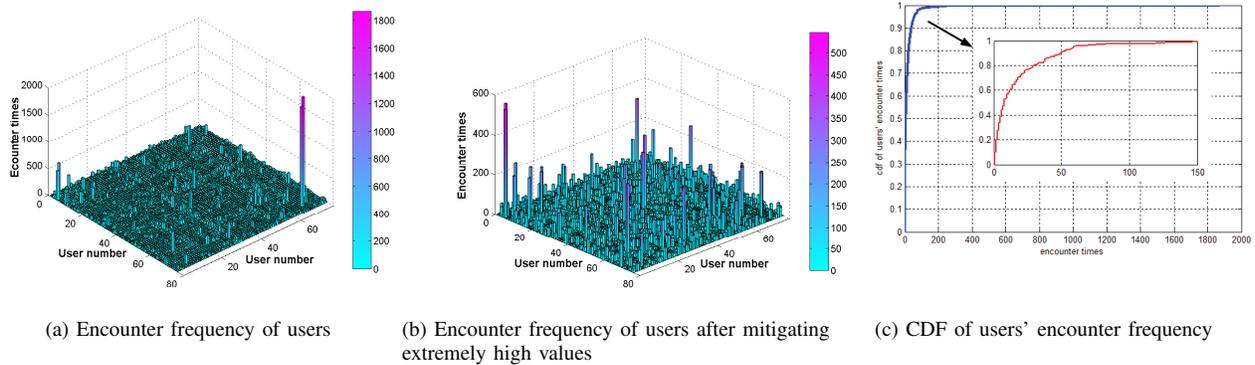


Fig. 2. Encounter frequency of users

relatively vital issue for each user. The users would like to believe their friends in direct contacts². Also, different from delay tolerant network routing, there is no destination for task forwarding. Thus, multiple forwarding will inevitably affect the task execution time seriously. We put it to future work, where routing with social relationship technology is incorporated for enhanced load balancing. The tasks in user i have not given priority here, and we do not consider the ‘dequeueing’ and ‘enqueueing’ technologies for task priority. Because in social task offloading network, there would be very few urgent tasks for reassignment. The tasks with extremely high priority should be processed on local devices.

For user i , the tasks getting from other users are listed in queue $Q_i = \{\tilde{t}_i^1, \tilde{t}_i^2, \dots, \tilde{t}_i^m\}$, and the queueing length is $\|Q_i\|$. Note that, the tasks assigned to user i , are different from the previous task definition, because these tasks are from different users. Thus, we denote these tasks with \tilde{t}_i^j , which means the j^{th} tasks assigned to user i .

B. Problem formation

The investigated problem could be formulated by minimizing the gap between the average value and the maximum queueing length achievable *w.h.p.*. The gap evaluation has two advantages. First, it is simple. Instead of computing the gap between average value and all the queues, the maximum queueing length is investigated, which saves large amount of communication overhead in distributed mobile network. Without loss of generality, such evaluation is also reasonable. Second, it is an important metric in evaluating between the worst and the average case. For example, we need to know the task finishing time in average and the worst cases. Also, in batched task offloading case, the task finishing time depends on the latest one, which corresponds to the longest queueing length. Thus, such evaluation is given by:

$$\min_{i \in U} (\max \|Q_i\| - E_{i \in U} [Q_i])$$

Two technical challenges need to be formally addressed

²Here contacts means the two users are in communication range and could exchange data successfully. Direct contacts are denoted to differ the case that needs forwarding.

before applying social relationship into task offloading algorithms. First, when considering the social relationship, the selection of mobile users becomes unstable and inefficient. Increasing the number of candidate users will not help much, because the number of contacted users is subjected to mobility patterns, which might not be large especially during specific ‘contact window’. Second, there are ‘exceptional effects’ in our model, *i.e.*, some extremely high and low meeting frequencies among users will affect the overall performance significantly. Once the tasks are assigned to an infrequently contacted user, the task completion time would be extremely long, especially when acknowledgment is required. Thus, the design requirement needs efficient and stable mobile user selection schemes, where balanced task assignment could be achieved, without being affected by intermittently connected mobile users seriously. We show this property with simulation and statistical results in Section V and VI.

V. ‘ i TOP-K’: SOCIAL RELATIONSHIP BASED TASK OFFLOADING ALGORITHM

In tackling with the aforementioned challenges, we propose a social relationship based task offloading algorithm. We leverage the concept of ‘Top-K’, where the first K users in the friend list are selected for task allocation. The ‘Top-K’ friends bear more regular contact pattern than others, which will lead to more stable task allocations. The only negative effect of ‘Top-K’ scheme is, when there are no users in contact window, the ‘Top-K’ scheme will lead to unbalanced allocation or longer delay. In this work, we use an adaptive scheme, where the number of ‘ K ’ is adaptive and scalable. Such that, we call our scheme ‘ i Top-K’, which means a customized and adaptive method. The ‘ i Top-K’ algorithm slightly modifies the original ‘ d -choice’ scheme, where the random ‘ d ’ candidates are replaced by the selected ‘friends’ in ‘Top-K’ discipline.

A. Finding the ‘Top-K’ Social Contacts

We leverage the trace data from MobiClique application at ACM Sigcomm conference 2009 [2]. The trace data recorded traces of Bluetooth encounters, opportunistic messaging, and social profiles of 76 users. We pick up the records of user

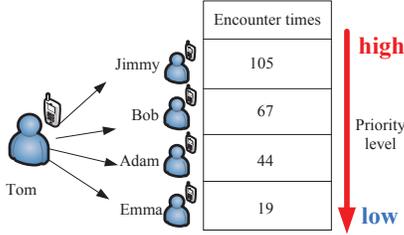


Fig. 3. Considering task priority with social contacts

TABLE I
ILLUSTRATION FOR SOCIAL RELATIONSHIP RANKING LIST

User	Top-K(K=3) contact user	Priority
1	5 (545), 10 (231), 69 (119)	1, 2, 3
2	75 (23), 63 (16), 66 (12)	1, 2, 3
3	32 (146), 33 (126), 4 (58)	1, 2, 3
4	3 (58), 32 (53), 29 (47)	1, 2, 3

$i(1 \leq i \leq 76)$ and sort them in descending order like Table I. As a result, we get the ‘Top-K’ friend list for each user i .

We build a relationship table according to the encountering frequency similar to Fig.3. Note that, for user Jimmy, who has the highest priority level, the number of contacts between Jimmy and Tom is 105 times during the data collection period. To decrease the task execution time, we need to assign the tasks to close friends or familiar people.

In mobile social network, the mobility dominates the task allocation opportunity. Different from conventional task allocation, the availability of each mobile user is considered in mobile network, as the availability differs from each other. Pure balancing among users would lead to unfairness among users. The reason for this claim is, when the availability of different users differs, say, some mobile users are frequently contacted while others are not. If we still allocate tasks to each user equally, the task allocation might be not efficient. First, the tasks assigned to inactive mobile users would lead to larger task execution time. In contrast, the active mobile users could be more efficient in task allocation. For efficient offloading consideration, the active mobile users could bear more tasks and make the network more efficient. Specifically, since the availability among users differs, we should not simply apply the load-balancing scheme to the mobile crowdsourcing network directly.

Thus, in our scheme, we pursuit another form of balancing, that is, users are allocating tasks according to the availability of mobile users, which is subjected to the mobility patterns. Moreover, there should be a balanced allocation among users with unequal availability. Thus, allocating more tasks to frequently contacted user could effectively improve the network efficiency, because tasks assigned to these users could be effectively executed, and the task execution results could be returned to the task sender in a relatively short

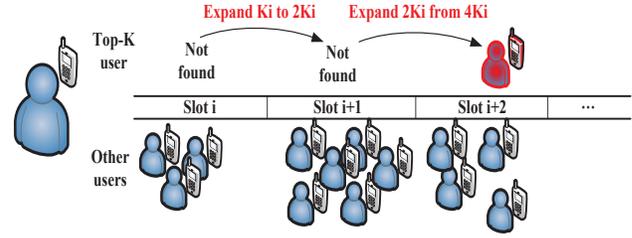


Fig. 4. Waiting for ‘Top-K’ users with scalability

time. In contrast, for the infrequently contacted users, when more tasks are assigned, the time delay could be very long. Thus, the balancing scheme should be tailored for mobility features. Also in the social domain, the mobility patterns could be leveraged for task execution. Because in social network discipline, intimate users would have a close relationship, and would like to execute the tasks with higher priority. This feature has been widely explored and exploited in many literatures [38] [24]. We make further exploration on our trace data for spatial and temporal correlation feature. Such feature could further support the close relationship between frequently contacted users. Thus, in our scheme, we fully consider the mobile and social features in load balancing scheme, where tasks are allocated according to the mobility patterns and social relationships.

Algorithm 1 iTop-K algorithm for task offloading in mobile social network

Input:

- trace data of Sigcomm2009 [2]
- users number, n

Output:

- queue length: w_1, w_2, \dots, w_n
- 1: **Initialize** $\{w_1, w_2, \dots, w_n\} = 0, count = 0$
- 2: Load trace data of SIGCOMM 2009
- 3: **Procedure** FindTopK(n, K)
- 4: Sort the encountering frequency list for each user
- 5: Record in file
- 6: **End Procedure**
- 7: **Procedure** Top-KAllocate(n)
- 8: **For** each slot **do**
- 9: search for encounters between users
- 10: **For** each user **do**
- 11: **If** Top-KContain(i)==true **then**
- 12: $S \leftarrow$ Top-K user first met in this slot
- 13: Flag=Rank(S)
- 14: **Else** Expand K value
- 15: **End**
- 16: $w_s = w_s + 1/g^{(K_{max}-Flag)}$
- 17: **Endfor**
- 18: **Endfor**
- 19: **End Procedure**
- 20: **return** w_1, w_2, \dots, w_n

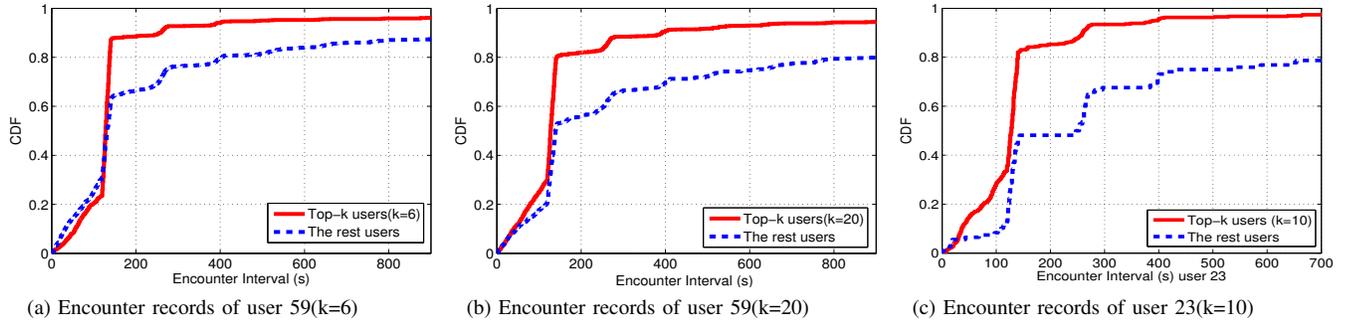


Fig. 5. Evaluations for Top-k users inter-contact interval

B. Waiting for K Friends with Scalability

We consider the task execution time, when ‘Top-K’ friends are identified. As shown in Fig. 4, if there is no user in the ‘Top-K’ list in one contact window, assigning tasks to ‘non Top-K’ users will not ensure the high contact frequency. Also, the task execution priority will not be guaranteed, which will be discussed in Sec. V-C. As shown in Fig. 4, at the given slot, say slot i , if in the next slot $i + 1$, there is no user in the ‘Top-K’ list. Thus, the selection scope K_i would be scaled to $2K_i$.

Further, if in the next slot $i + 2$, there is still no ‘Top-K’ friends, the selection scope K_{i+2} would be $2K_{i+1} = 4K_i$. Once there are users in the scalable ‘Top-K’ list, the selection scope returns to K_{min} , and in our scheme, we set it to 2, since it is the minimum value required for ‘d-choice’ scheme.

The scaling law of selection scope K is given by the following equations:

$$K_{i+1} = \begin{cases} 2K_i & \text{if there is no Top-K friend in slot } i \\ 2 & \text{if there is at least one Top-K friend in slot } i \end{cases}$$

C. Considering Task Priority with Social Contacts

In social network, friends often contact with each other frequently. According to previous studies, people meeting each other often would indicate higher social relationships among them [17]. And in these relationships, the task execution priority will be higher than others.

Previous studies have considered fully on the social relationship and the psychology factors. As the task execution priority is affected by the psychology reasons, the task execution time would follow the exponential feature, where users in different ranks may differ significantly in execution time [18]. In our study, the task execution time formula is empirical. And we set the task execution time for user ranked in the i^{th} place to:

$$T^i = T_\alpha \times 2^i$$

where T_α is the original task execution time, say, without priority. And K is the selection scope for ‘Top-K’ selection. Note that, for the scalable case, the factor K also scales.

Considering various psychology impacts and social relationships, the task execution time can be given by:

$$T^i = T_\alpha \times g^i$$

where $g > 1$ is the base factor.

The algorithm description is shown in Algorithm 1. The procedure $FindTopK(n, K)$ is called to get the ‘Top-K’ list of each user, and the procedure $Top - K Allocate(n)$ allocates tasks to users according to the ‘Top-K’ list. The function $Rank(S)$ could get the actual rank of user S . Also, $Top - K Contain(i)$ could check if user i has ‘Top-K’ friends in contacted ranges, where K_{max} is the maximum number of scalable K .

D. Discussions

The duration of a contact window could be formally defined in the following way: $T_w = t_e - t_s$, where the t_s denotes the starting time that the mobile users could transmit data over a contact session, and the t_e denotes the time that the contact session ends. Before the contact session, the neighbor discovery and handshake process for data transmission have been initiated. In our scheme, we assume that, the task information, *i.e.*, the task execution is delay tolerant. Thus, the major concern is the task load. We do not evaluate the finishing time for tasks. And the execution results for tasks need not to be fed back to the task sender within contact windows. In our model, we only consider the balancing property among mobile users. The application is for cooperative task execution, which is typical in most of the previous studies on crowdsourcing network. The main purposes for this cooperation are to save energy and make good use of computation resources with fair task allocations.

The contact duration is also an important factor to our scheme, especially when the task offloading overheads are concerned. In our system model, we assume that, the duration of contacts is sufficient for task sharing for two reasons. First, the mobile users would use short range, high bandwidth transmissions, such as WiFi and Bluetooth, where short transmission duration is required. Second, the contact durations, especially for the trace data set, are sufficiently large for simple data sharing. According to our statistical results, most of the contact duration is larger than several tens of seconds. Moreover, in most of the studies [38] [24], very short contact durations with unstable connections are generally not considered as contacts.

In this scenario, we also assume the inter contact interval is much greater than task execution time. Since in delay tolerant network, the inter contact interval could be several seconds to

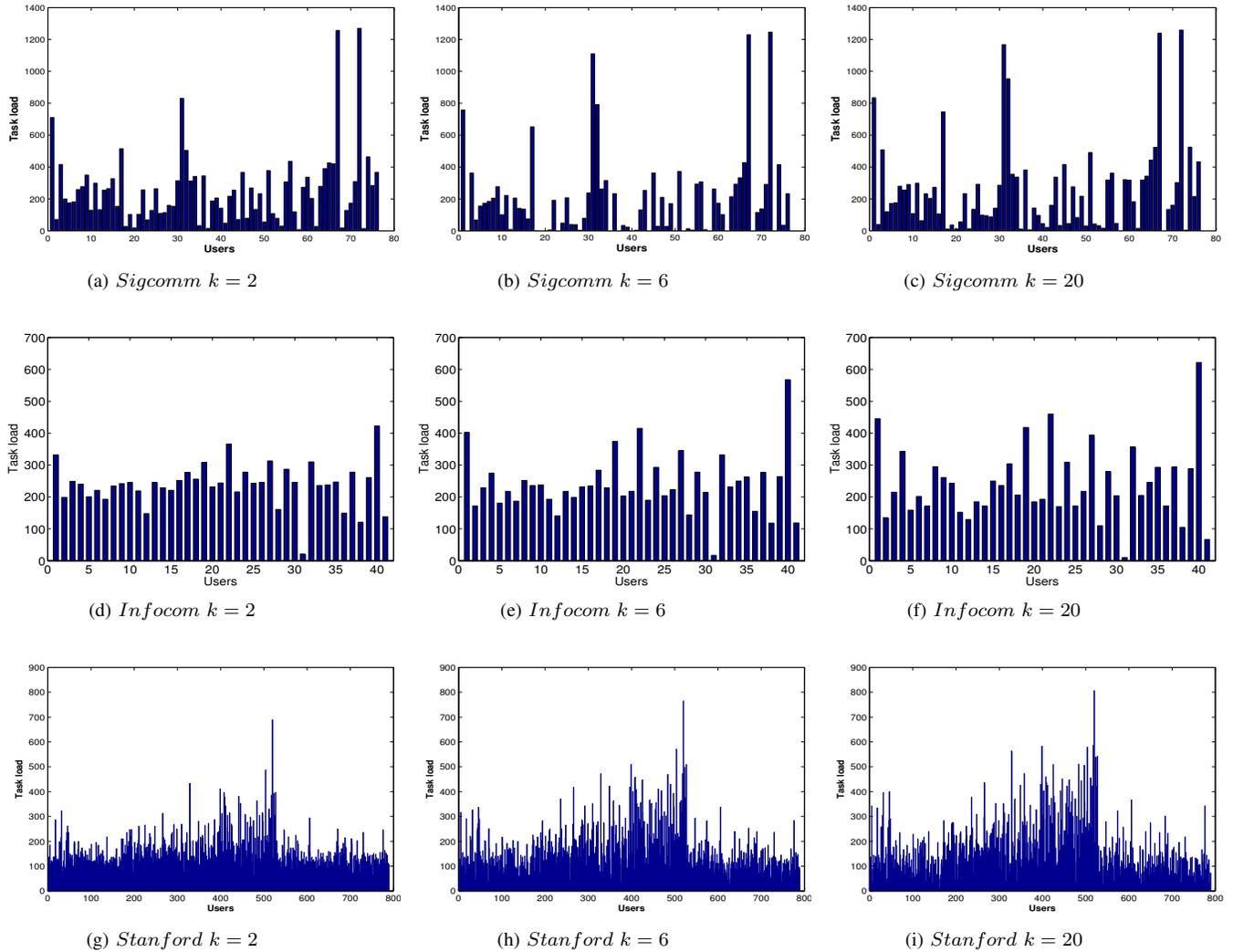


Fig. 6. Impact of K value when tasks are executed without priority

minutes long. Considering the bandwidth constrained mobile device-to-device scenario, the tasks could be generally simple with little traffic, means the relatively short time of the returning of results. We make further simulation study and show the aforementioned effects. As shown in the following figures, mobile users with shorter inter contact interval are selected as friends or Top-K users, where the tasks are executed more efficiently. Even for the case where the task execution time is longer than inter contact time, since the users with close relationship has higher priority for task execution, the task could also be processed with shorter time. We compare the encounter interval of top-k users and the rest users. As depicted in Fig 5, we find that, for highly contacted users, the inter-contact interval could be smaller than others, which have been depicted in aforementioned evaluation results. Without loss of generality, in Fig. 5a, user with No. 59 is selected for illustration. Users in No. 23 and 40 are further evaluated and comparison are made among the ‘Top-K’ users and the rest of common users, which are depicted in Fig. 5b and 5c. Such that, when tasks are assigned to top-k users, the execution

results could be returned to the task sender when next time contact happens.

VI. EXPERIMENTAL STUDY WITH REAL TRACE DATA SETS

A. Finding ‘Top-K’ users

In this paper, we use three experimental data sets for evaluations, which are: *Sigcomm-2009* [2], *Infocom05* [4], and *Stanford-2010* [3].

- *Sigcomm-2009* [2] recorded encounters between 76 people participated in Sigcomm’09.
- *Infocom-2005* [4] collected the same data between 41 people from Infocom’05 at Grand Hyatt Miami in March 2005.
- *Stanford-2010* [3] logged face-to-face contacts among 789 participants in a US high school between 7AM to 4PM.

The simulator is implemented in Matlab platform, and executed in a laptop of Lenovo X201, with Intel i5CPU (2.53GHz), and 4GB memory. This simulator has been tested and verified in our previous studies [11] [19].

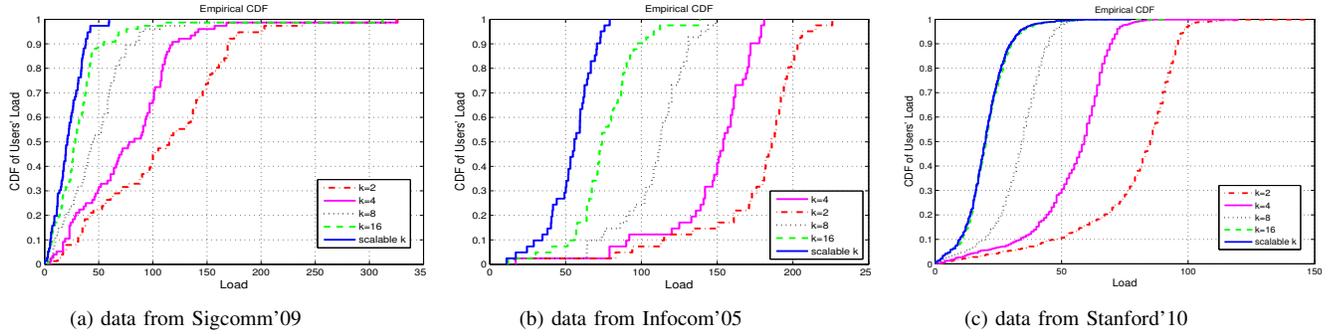
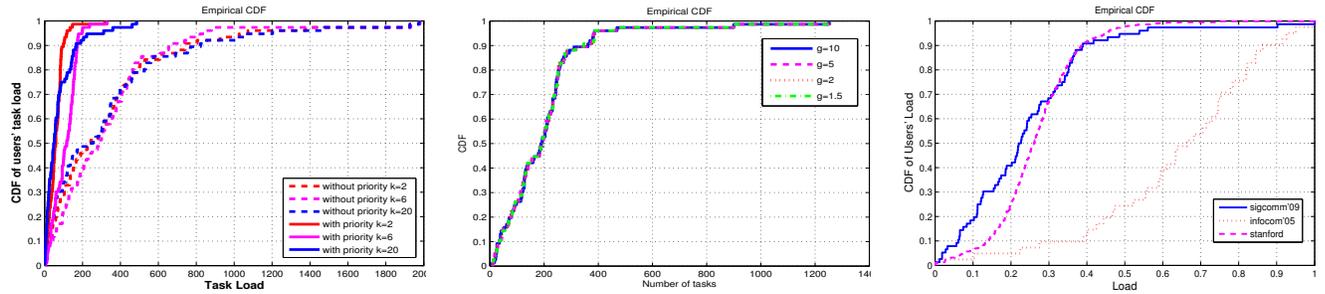


Fig. 7. Top-K scheme with task priority



(a) CDF of tasks for schemes with and without priority (b) Top-K scheme with task priority, when the base value changes (c) Efficiency compare among three datasets

Fig. 8. Evaluations for different factors impacting the task balancing performance

B. impact of Top-K without priority

Sigcomm-2009 [2]: We first evaluate the performance of the scalable ‘K’ with real traces from MobiClique. In order to execute task allocation in every time slot, we divide the original time data into series of time slots. As the earliest time stamp is 30s and the latest is 320684s, the slot interval is set into 200s according to previous analysis about contact intervals and distributions. It is a typical value for most of the mobile contacts, and also sufficient for data transfer in necessary. In each time slot, each user will share only one task. Thus, users allocate their tasks in each time slot respectively.

First, in order to show the effectiveness of task execution priority, we make an experimental study on the scheme ‘Top-K’ without priority. As shown in Fig. 6a, 6b, 6c, the ‘K-value’ is set into 2, 6 and 20 respectively. Note that, $K = 2$ is a basic setting, and $K = 6$ is a typical value when mobile traces are applied. Also, note that, the case $K = 20$ means very large selection scope for candidate users.

Infocom-2005 [4]: We then evaluate the performance of the scalable ‘K’ with real traces from ‘Haggle’. Considering the earliest time stamp is 20733s and the latest is 274883s, the slot interval is set into 200s, which is appropriate according to previous analysis about contact intervals and distributions. Fig. 6d, 6e, and 6f depict the task load, especially when tasks are executed without priority. Even when the K value increases, there are still large number of imbalanced task loads among users.

Stanford-2010 [3]: The third trace comes from [3]. Fig. 6g, 6h, and 6i show the performance of same scheme on ‘Top-K’

without priority. Although the Stanford trace has shown significantly good social relationship property, the task execution without priority still leads to poor performance. As depicted in these sub-figures, the task allocations have not shown effective load balancing among users. To this end, we can conclude that, increasing selection scope K only will not improve the balancing performance, especially when social relationships are dominating the user contacts.

C. Impact of Top-K with priority

Sigcomm-2009 [2]: We show the effectiveness of task execution priority in *Sigcomm-2009* data set. The task priority is set according to the method proposed in Sec. V-C. Fig. 7a shows that, the tasks are well balanced among users, even when K is set to 2. Notably, when the K value increases, the performance improves accordingly. The ‘scalable K’ scheme ($K_{min} = 2$) performs better comparing to the static case when ‘ $K=2$ ’, and performs even better than the case when ‘ $K=16$ ’.

Infocom-2005 [4]: As shown in Fig. 7b, we evaluate the scheme ‘Top-K’ with priority in data set ‘*Infocom-2005*’. The figure validates that the effective load balancing could be achieved when task priority is applied. Since *Infocom-2005* trace data has not shown strong social contact relationship, the value of K should be set larger to achieve balanced performance. We could see that, when K is greater than 8, especially when $K = 16$, the tasks could be well balanced, achieving similar performance as shown in Fig. 7a.

Stanford-2010 [3]: As shown in Fig. 7c, note that the ‘scalable k ’ curve is almost in same shape with the ‘ $k=16$ ’

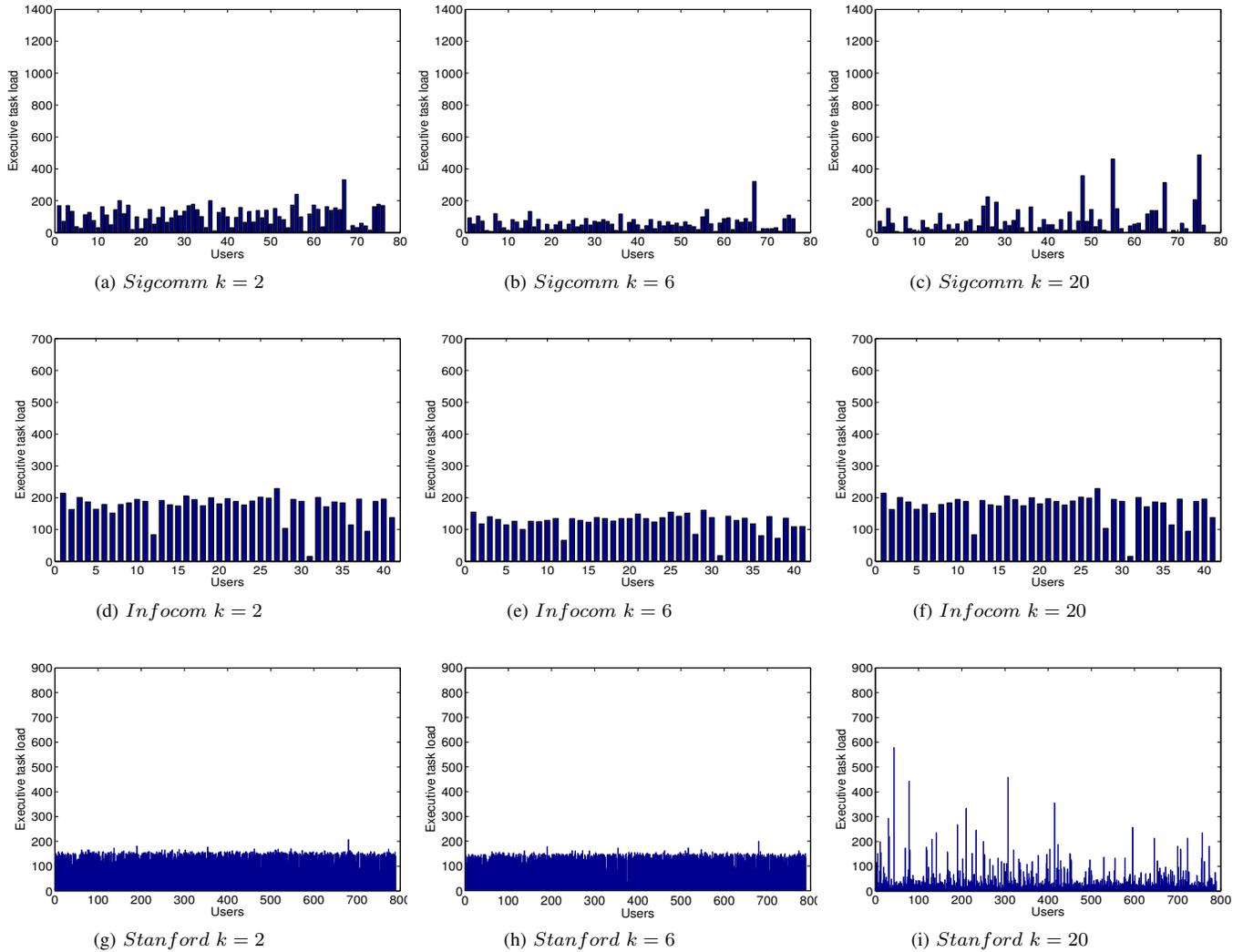


Fig. 9. Impact of K value when tasks are executed with priority

line. The reason is, the number of users in this trace is far more greater than the previous two data sets, where ‘marginal effects’ are more significant. Selecting ‘Top-K’ friends becomes more easy and convincible, which shows more potential when the algorithm is applied to the realistic social networks.

Remarks: The priority based execution effectively improves the task balancing performance. First, the social relationship, which is dominated by mobile contacts, could be fully leveraged for more than task assignment, and even further for execution. Second, the favorable feature is, the ‘Top-K’ users could be effectively used for efficient task offloading, where users could reach the relatively frequent user without big efforts. As depicted in Fig. 8a, the priority based schemes outperforms those methods without priority significantly.

D. Scaling the task execution factor

We also consider the case when the base value g is set to other values for psychology and social relationship reasons. That is, we need to scale this factor to see the impact of task execution. Fig. 8b tells that, when the base value g changes,

the task assignments provide similar performance, and we conclude that, this factor does not impact the performance significantly when task balancing is concerned.

E. Efficiency Comparison among different data sets

In order to compare the execution time of algorithm ‘ i Top-K’ with three data sets, we normalize the task weight into the range $[0, 1]$. As depicted in Fig. 8c, under trace data set *Infocom-2005*, ‘ i Top-K’ performs relatively poor comparing with the other two data sets. The reason is, records in ‘Infocom-2005’ are a little bit sparse, when the encountering frequencies are concerned. The relatively infrequent contacts would possibly lead to weak ‘Top-K’ friend list. In contrast, ‘ i Top-K’ performs well when *Stanford-2010* is applied. When more participants could be involved, especially for the real social environment, ‘ i Top-K’ would achieve better performance.

In summary, comparing with the most loaded users, our proposed scheme outperforms the conventional random choice up to $15\times$, and the social relationship assignment without priority method up to $9\times$.

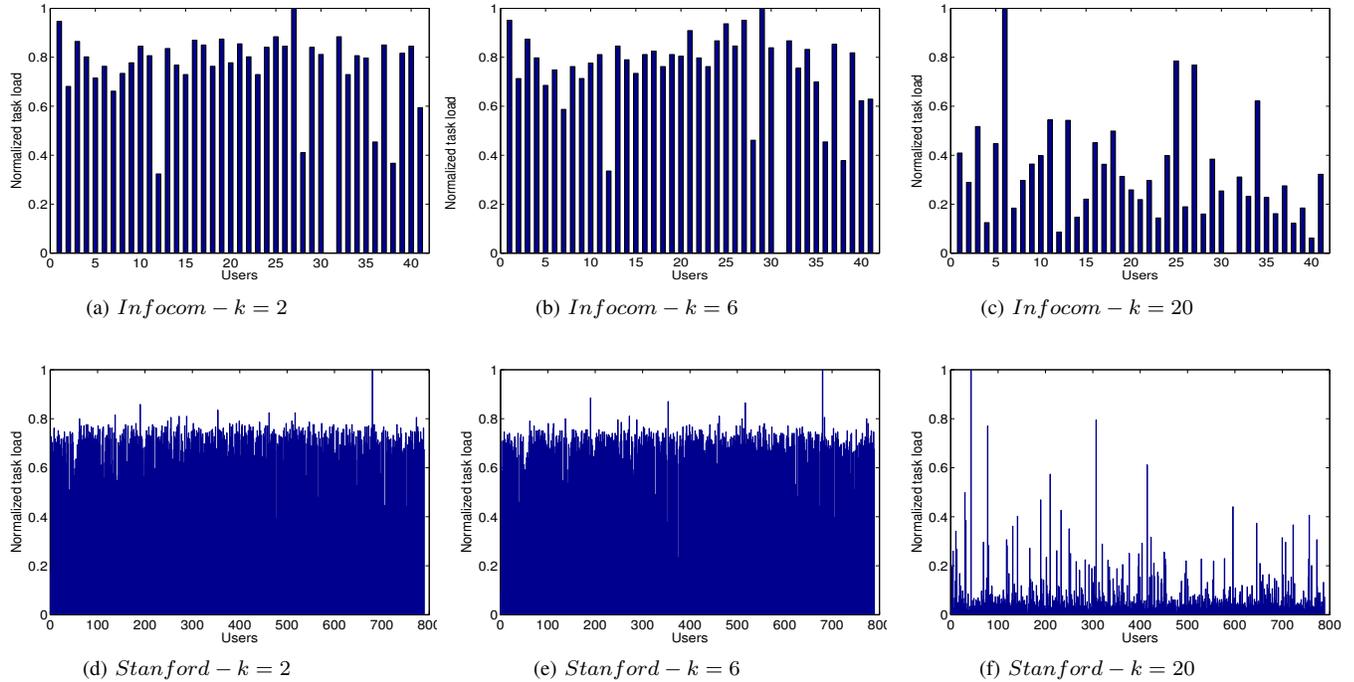


Fig. 10. Normalized task load performance of scheme with priority (Infocom and Stanford)

We show our results with the bar chart and CDF form, making comparisons for schemes with priority and without priority. As depicted in Fig. 6, the scheme without priority performs poor with imbalanced task load assignment. While for the priority based scheme, as depicted in Fig. 9, the balancing performance significantly improves comparing with the scheme without priority. While for trace Infocom and Stanford, the results are in Fig. 10. In general, scheme with priority works better than scheme without priority. However, when $k=20$, scheme with priority has not shown much superiority, where the friend list is too long that the lower-ranking users have little difference in task execution time.

VII. CONCLUSION

We propose a social relationship based algorithm for efficient task offloading, with trace data collected from real deployed mobile social networks [2] [4] [3]. We find that, the social relationship could be explored and exploited for balanced task offloading among mobile users. Task priority associates with the social relationships, and plays an important role for task execution time. Further, waiting for the forthcoming friends is applicable and effective for mobile users. In future work, we are going to further explore the social relationship and find more convincing models for task execution among social friends. Also, more effective methods for Top-K friends are called for, especially, when the task execution time differs significantly among friends and strangers. At last, we plan to apply our methods to realistic applications like processing pictures to translate words within a crowd of people.

REFERENCES

[1] M. Michael, A. W. Richa, and R. Sitaraman, "The power of two random choices: A survey of techniques and results," 2005.

[2] A.-K. Pietilainen, E. Oliver, J. LeBrun, G. Varghese, and C. Diot, "Mobiclique: Middleware for mobile social networking, proceedings of," in *Proc. the 2nd ACM SIGCOMM Workshop on Online Social Networks*, 2009.

[3] M. Salathé, M. Kazandjieva, J. W. Lee, P. Levis, M. W. Feldman, and J. H. Jones, "A high-resolution human contact network for infectious disease transmission," *Proceedings of the National Academy of Sciences*, vol. 107, no. 51, pp. 22 020–22 025, 2010.

[4] "Haggle project," www.cambridge.intel-research.net/haggle/.

[5] B. Han and A. Srinivasan, "Your friends have more friends than you do: Identifying influential mobile users through random walks," in *the 13th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, 2012.

[6] Z. Fu, X. Sun, Q. Liu, L. Zhou, and J. Shu, "Achieving efficient cloud search services: Multi-keyword ranked search over encrypted cloud data supporting parallel computing," *IEICE Transactions on Communications*, vol. E98-B, no. 1, pp. 190–200, 2015.

[7] K. K. Rachuri, C. Mascolo, M. Musolesi, and P. J. Rentfrow, "Sociable-sense: Exploring the trade-offs of adaptive sampling and computation offloading for social sensing," in *MobiCom*, 2011.

[8] K. K. Rachuri, M. Musolesi, and C. Mascolo, "Energy-accuracy trade-offs in querying sensor data for continuous sensing mobile systems," in *Mobile Context-Awareness Workshop*, 2010.

[9] D. Wang, L. Kaplan, and C. C. Aggarwal, "On quantifying the accuracy of maximum likelihood estimation of participant reliability in social sensing," in *International Workshop on Data Management for Sensor Networks(DMSN)*, 2011.

[10] C. Shi, V. Lakafosis, M. H. Ammar, and E. W. Zegura, "Serendipity: Enabling remote computing among intermittently connected mobile devices," *MobiHoc 2012*.

[11] Q. Li and P. Yang, "Store: Simple task offloading and reassignment for mobile social network," in *Proc. IEEE Chinacom*, 2013.

[12] P. Key, L. Massouli, and D. Towsley, "Multipath routing, congestion control and load balancing," in *Proc. IEEE ICASSP*, 2007.

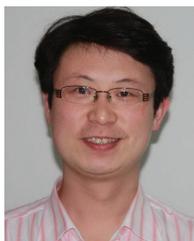
[13] L. Rudolph, M. Slivkin-Allalouf, and E. Upfal, "A simple load balancing scheme for task allocation in parallel machines." *The Third Annual ACM Symposium on Parallel Algorithms and Architectures*, 1991.

[14] V. Stemann, "Parallel balanced allocations." *The Eighth ACM Symposium on Parallel Algorithms and Architectures*, 1996.

[15] J. Fan, J. Chen, Y. Du, W. Gao, J. Wu, and Y. Sun, "Geo-community-based broadcasting for data dissemination in mobile social networks," in

IEEE Transactions on Parallel and Distributed Systems, vol. 24, no. 4, 2013, pp. 734–743.

- [16] J. Fan, J. Chen, Y. Du, P. Wang, and Y. Sun, “Delque: A socially-aware delegation query scheme in delay tolerant networks,” in *IEEE Transactions on Vehicular Technology*, vol. 60, no. 5, 2011, pp. 2181–2193.
- [17] T. Hossmann, T. Spyropoulos, and F. Legendre, “Putting contacts into context: Mobility modeling beyond inter-contact times,” in *ACM MobiHoc*, 2011.
- [18] B. Schwartz, “Queries, priorities, and social process,” in *Socila Psychology*, 1978.
- [19] Q. Li, P. Yang, and Y. Tao, “Your friends are more powerful than you: efficient task offloading in mobile social networks,” in *IEEE ICC*, 2014.
- [20] K. Pearson, “The problem of the random walk,” *Nature*, vol. 72, p. 1865, 1905.
- [21] M. E. Newman, “A measure of betweenness centrality based on random walks,” *Social Networks*, vol. 27, pp. 39–54, 2005.
- [22] J. D. Noh and H. Rieger, “Random walks on complex networks,” *Physical Review Letters*, vol. 92, 2004.
- [23] P. Marshall, “Darpa progress towards affordable, dense, and content focused tactical edge networks,” in *IEEE Milcom*, 2008.
- [24] K. Fall, G. Iannaccone, and J. Kannan, “A disruption-tolerant architecture for secure and efficient disaster response communications,” in *IEEE ISCRAM*, 2010.
- [25] “Casting words,” <http://castingwords.com>.
- [26] “Crowd flower,” <http://crowdflower.com>.
- [27] “Crowd spring,” <http://www.crowdspring.com>.
- [28] R. Rana, C. Chou, S. Kanhere, N. Bulusu, and W. Hu, “Ear-phone: an end-to-end participatory urban noise mapping system,” in *International Conference on Information Processing in Sensor Networks (IPSN)*. ACM, 2010, pp. 105–116.
- [29] M. Faulkner, M. Olson, R. Chandy, J. Krause, K. M. Chandy, and A. Krause, “The next big one: Detecting earthquakes and other rare events from community-based sensors,” in *IPSN*. IEEE, 2011, pp. 13–24.
- [30] W. Willett, P. Aoki, N. Kumar, S. Subramanian, and A. Woodruff, “Common sense community: Scaffolding mobile sensing and analysis for novice users,” *Pervasive Computing*, vol. 6030, pp. 301–318, 2010.
- [31] P. Volgyesi, A. Ndas, X. Koutsoukos, and J. Ldcezi, “Air quality monitoring with sensormap,” in *IPSN*. IEEE Computer Society, 2008, pp. 529–530.
- [32] M. Mun, S. Reddy, K. Shilton, N. Yau, J. Burke, D. Estrin, M. Hansen, E. Howard, R. West, and P. Boda, “Peir, the personal environmental impact report, as a platform for participatory sensing systems research,” in *Proceedings of the 7th international conference on Mobile systems, applications, and services*. ACM, 2009, pp. 55–68.
- [33] P. Guo, J. Wang, B. Li, and S. Lee, “A variable threshold-value authentication architecture for wireless mesh networks,” *Journal of Internet Technology*, vol. 15, no. 6, pp. 929–936, 2014.
- [34] J. Shen, H. Tan, J. Wang, J. Wang, and S. Lee, “A novel routing protocol providing good transmission reliability in underwater sensor networks,” *Journal of Internet Technology*, vol. 16, no. 1, pp. 171–178, 2015.
- [35] N. Maisonneuve, M. Stevens, and B. Ochab, “Participatory noise pollution monitoring using mobile phones,” *Information Polity*, vol. 15, no. 1, pp. 51–71, 2010.
- [36] S. Xie and Y. Wang, “Construction of tree network with limited delivery latency in homogeneous wireless sensor networks,” *Wireless Personal Communications*, vol. 78, no. 1, pp. 231–246, 2014.
- [37] D. Yang, G. Xue, X. Fang, and J. Tang, “Crowdsourcing to smartphones: incentive mechanism design for mobile phone sensing,” in *MobiCom*, 2012.
- [38] D. Zheng, W. Ge, and J. Zhang, “Distributed opportunistic scheduling for ad hoc networks with random access: an optimal stopping approach,” *IEEE Trans. Inf. Theor.*, vol. 55, pp. 205–222, January 2009.



Panlong Yang (M'02) received his B.S. degree, M.S. degree, and Ph.D. degree in communication and information system from Nanjing Institute of Communication Engineering, China, in 1999, 2002, and 2005 respectively. Dr. Yang is now a professor in the College of Computer Science and Technology, University of Science and Technology of China. His research interests include wireless mesh networks, wireless sensor networks and cognitive radio networks. He is a member of the IEEE Computer Society and ACM SIGMOBILE Society.



Qingyu Li (S'12) received the B.S. degree in Nanjing University of Posts and Telecommunications in 2012 and her Master degree at PLA University of Science and Technology in 2015. Her research interests include mobile crowdsensing networks and mobile social networks.



Yubo Yan (S'10) received the B.S. degree and M.S. degree in communication and information system from the College of Communications Engineering, PLA University of Science and Technology, China, in 2006 and 2011 respectively. He is currently working towards the Ph.D. degree at the PLA University of Science and Technology. His current research interests include software radio systems and wireless sensor networks. He is a student member of the IEEE and the IEEE Computer Society.



Xiang-Yang Li (M99-SM08-F15) is a professor of Computer Science at the Illinois Institute of Technology and EMC Visiting Chair Professor at Department of Computer Science, Tsinghua University. He is recipient of China NSF Outstanding Overseas Young Researcher (B) in 2008. He received MS (2000) and PhD (2001) degree at Computer Science Department from University of Illinois at Urbana-Champaign, B.Eng. at Computer Science and Bachelor degree at Business Management from Tsinghua University, P.R. China in 1995. His research interests

include wireless networks, big data privacy and security, cyber physical systems, and social networks. He and his students won four best paper awards (IEEE IPCCC 2014, ACM MobiCom 2014, COCOON 2001, IEEE HICSS 2001), one best demo award (ACM MobiCom 2002) and was selected as best paper candidates twice (ACM MobiCom 2008, ACM MobiCom 2005). He serves as an editor of several journals, including IEEE TMC, and was an editor of “IEEE TPDS”. He is an IEEE Fellow and an ACM Distinguished Scientist.



Yan Xiong was born in Anhui Province, in 1960. He is a professor in School of Computer Science and Technology, University of Science and Technology of China. His research interests include distributed processing, mobile computation, and information security.



Baowei Wang received his B.S. and Ph.D. degrees in Computer Science from Hunan University in 2005 and 2011, respectively. He is currently working as a lecturer in School of Computer and Software, Nanjing University of Information Science and Technology. His research interests include steganography, wireless networks and securing ad hoc networks..



Xingming Sun is a professor in the School of Computer and Software, Nanjing University of Information Science and Technology, China from 2011. He received the B.S. degree in Mathematical Science from Hunan Normal University and M.S. degree in Mathematical Science from Dalian University of Technology in 1984 and 1988, respectively. Then, he received the Ph.D. degree in Computer Engineering from Fudan University in 2001. His research interests include information security, network security, cryptography and ubiquitous computing security.