Article

Influence of Crowd Participation Features on Mobile Edge Computing

Peiyan Yuan * , **Xiaoxiao Pang and Xiaoyan Zhao**

School of Computer and Information Engineering, Henan Normal University, Xinxiang 453000, China; pangsmile@foxmail.com (X.P.); zhaoxiaoyan@htu.cn (X.Z.)

* Correspondence: peiyan@htu.cn; Tel.: +86-159-3734-8382

Abstract: Mobile edge computing is a new communication paradigm, which stores content close to the end users, so as to reduce the backhaul delay and alleviate the traffic load of the backbone networks. Crowd participation is one of the most striking features of this technology, and it enables numerous interesting applications. The dynamics of crowd participation offer unprecedented opportunities for both content caching and data forwarding. In this paper, we investigate the influence of the dynamics of crowd participation, from the perspective of opportunistic caching and forwarding, and discuss how we can exploit such opportunities to allocate content and select relays efficiently. Some existing issues in this emerging research area are also discussed.

1. Introduction

With the rapid progress in communication and sensor technologies and the continuous pervasion of portable devices, we are entering the era of the Internet of Things (IoT) [1]. Many interesting applications such as location-based recommendation and augmented reality are emerging [2]. All of these require a better quality of experience because there will be limited time to wait for responses from remote cloud servers. Mobile edge computing (MEC) [3], which is one of the key technologies of IoT, bridges the gap between the real and virtual spaces and caches content in local or even in the edges of networks, so as to efficiently reduce the backhaul delay and improve the network capacity [4]. As shown in Figure 1, the user terminals (human/vehicles) can connect directly to the network via access points (AP) and receive the desired content from their neighbors or the local edge servers [5].

Different from the content distribution networks, MEC places content close to end users, while the content distribution networks mainly focus on publishing and subscribing of content. On the other hand, unlike traditional cloud computing technology, where the capabilities of the cloud servers play a big role, crowd involvement is one of the key factors leading to success in MEC. The dynamics of crowd behavior (mobility, preferences, etc.) is a double-edged sword [6,7]. From a negative perspective, it introduces new challenges in network management. For example, it changes the network topology over time, and the journey of content experiences intermittent connected links [8,9]. From a positive perspective, it brings many unprecedented opportunities ranging from better scalability of networks to low cost of deployment [10]. There are some works that consider the mobility or the sociality of nodes to form clusters, so as to exchange content among nodes efficiently [11,12]. For example, the authors of [13,14] employed features such as interest, energy and the physical distance proximity to classify nodes and improve the quality of service.
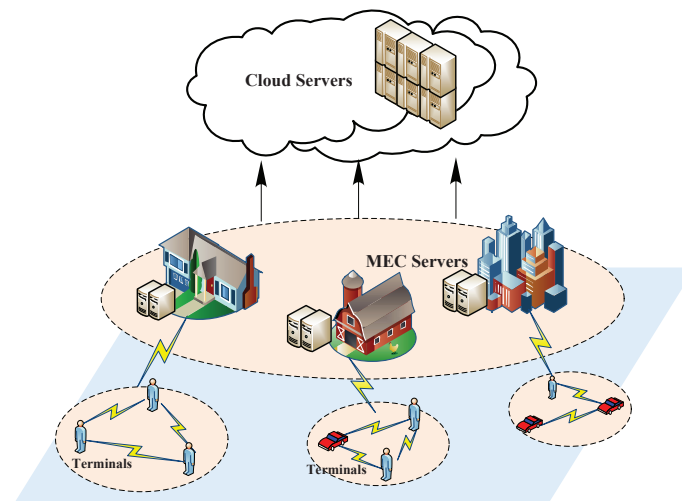
Figure 1. Illustration of mobile edge computing (MEC).

In this article, we specifically investigate how the crowd features can be exploited in the following two aspects:

- Opportunistic caching: The application can independently decide which content should be stored and shared, without active participation from end users.
- Opportunistic transmission: The routing path cannot be decided in advance, and the content is delivered in a store-carry-forward style.

In the following sections, we first discuss the opportunistic caching from the perspective of cache size, number of caching copies and content-allocation schemes. We then investigate the opportunistic transmission from the perspectives of the control and data planes, followed by an illustrated experiment. Finally, the conclusion and some open issues are presented in Section 5.

2. Opportunistic Caching

Many existing works have focused on the caching capabilities of small base stations (SBSs), neglecting the role of portable devices, which have an important influence on the edge caching system. Although an individual device has a relatively limited buffer space, putting several devices together (i.e., crowd participation) can provide enough content to satisfy various applications [15]. As an example, Figure 2 shows a system architecture in mobile edge computing, where some advertisement information, such as special bar recommendations and coupon distributions, can be pre-cached in the SBSs and diffused through user terminals (UT). In general, there are three basic questions that should be answered in such a system:

(1) How big a buffer space should each node reserve?
(2) How many copies should be stored for each content?
(3) Where should these copies be stored?

2.1. Cache Size: How Big Is the Buffer Space?

Since the connection between nodes in MEC cannot be guaranteed, the edge nodes must store part of the content in advance and cache some copies of the content temporarily in the buffer space. A fundamental question hence arises: how much buffer space should each node reserve? We think that an upper bound exists for the node buffer size, which is the smallest value that can guarantee the desired performance such as the caching hit ratio and transmission delay. In our study, the transmission process and the intermittently connected links are modeled by an edge-Markovian dynamic graph [16], and we analyze the buffer limitation from the percolation theory perspective. The inter-contact time

series reproduced by this model show a similar power law and exponential tail distribution as those observed in many real data traces. In the model, each edge has two states: active and inactive, which are distributed with an arbitrary initial probability. Based on a two-state Markovian process, one edge changes its state with an active probability p and an inactive one q between two sequent slots. Specifically, if an edge is active at slot t, it switches to the inactive state at slot $t + 1$ with a probability q. On the other hand, if it is inactive at slot t, it becomes active at the next slot with a probability p. Based on the percolation theory, there is a special state for the network, called the supercritical case, in which there exists a connected giant group at any moment when the density of nodes goes to infinity. The precondition for the supercritical case is $p > p_c$ (i.e., in an ultra-dense environment such as the cell networks we discussed), where $p_c = c \frac{\log n}{n}$, n is the number of nodes and c is a constant. We find that, in the supercritical case, there exits an upper bound on the node's buffer size, which is independent of the inactive probability q [17].
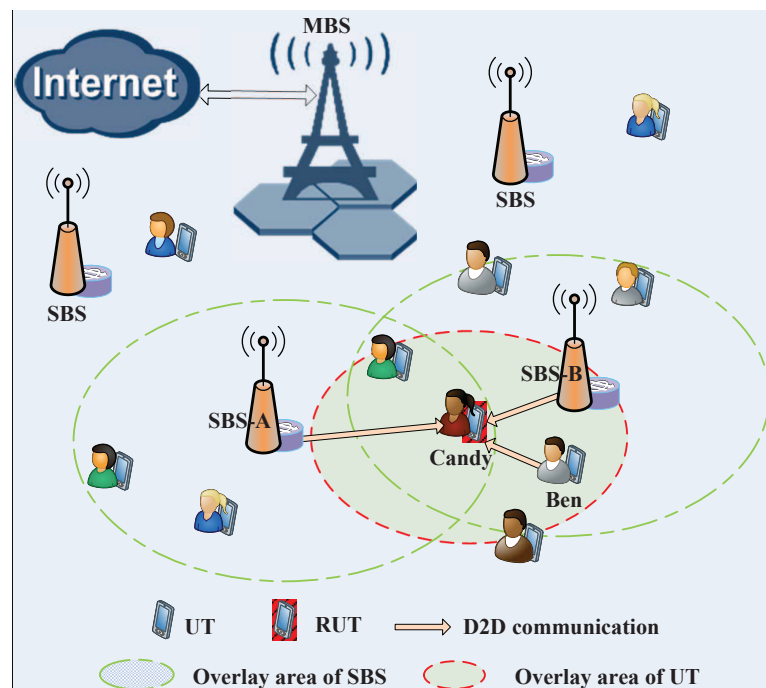

Figure 2. System architecture. MBS, macro base station; SBS, small base station; UT, user terminal; RUT, requesting user terminal; D2D, device-to-device.

Upper Bound Analysis

In the supercritical case, depending on whether the nodes belong to the giant cluster or not, they can be classified into two types: connected nodes and disconnected nodes. The former belongs to the giant cluster, while the latter does not. The connected nodes constitute a connected group, where packets are forwarded fast. On the contrary, buffer sizes of disconnected nodes are larger than those of the connected nodes, because packets in the disconnected nodes need to wait a longer time until the edges turn active. The worst case is that both the source node s and destination node d are separated by the giant cluster. This leads to the maximum time and buffer space being required to store the packets, as shown in Figure 3. There are three phases in the transmission process in this case: the source flooding phase, the shortest path phase and the destination flooding phase [18].

(1) Source flooding phase: In general, the flooding strategy has the lowest transmission delay and highest packet delivery ratio; at the same time, it results in the largest buffer space, which is necessary for this paper, since we focus on the fundamental bounds on node buffer size, especially in the worst case. In the flooding scheme, if both the source s and its neighbors belong to disconnected nodes, packets must be flooded to each other when the link between them is active. That is, if there is a packet

at node u at time slot $t - 1$, node u will send the packet to all its neighbors at time slot t. We use an infectious-disease-diffusion algorithm to characterize the flooding process [19] and classify nodes into a susceptible state and an infected state. A node is called infected if it carries a packet; otherwise, we call it susceptible.
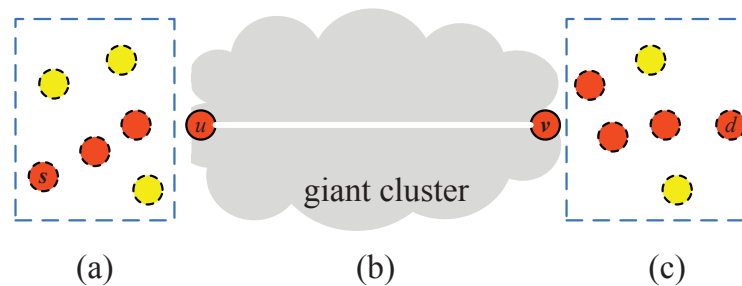

Figure 3. Three phases in the data forwarding strategy. (**a**) Source flooding phase; (**b**) shortest path phase; (**c**) destination flooding phase.

Based on the disease-diffusion algorithm, the source node s first transmits multiple copies of a packet m to neighbors. The newly infected nodes then repeat this process until one or more connected nodes belonging to the giant component receive m, as shown in Figure 3a. At the end of this phase, one connected node and several disconnected nodes consist of the source expanding tree, SET.

(2) Shortest path phase: In this case, a connected path for each source-destination pair exists with a high probability in the giant cluster; the packets therefore can be transmitted along this path. In other words, if there is a packet at time slot $t - 1$ at node u, node u will instantaneously send the packet to the next hop along the shortest path at the next time slot t, only if the edge between them is active. Thus, the packets will quickly reach the node v as shown in Figure 3b.

(3) Destination flooding phase: After the node v receives the packet m, it floods m among the disconnected nodes until one of the infected nodes encounters the destination node d. Finally, all infected nodes in this phase constitute the destination expanding tree, DET as shown in Figure 3c.

From the above discussion, we can infer that the buffer occupation in the disconnected nodes is low. The reason behind this is that there exists a giant cluster, leading to a finite size of SET and DET in each time slot. Hence, the disconnected nodes only need to cache packets coming from nearby sources or going to nearby destinations.

We can obtain the expected waiting time before the links become active through the property of the edge-Markov chain and derive the expectation of the buffer size for a random selected node. Based on the edge-Markov model, it can be inferred that the time when the packet is first sent from the source node into the connected nodes does not exceed $logn/pM$, and the expectation of the buffer occupation of node u will be the sum of the above two parts: $E(B_u(t)) = c_1 r T_d (logn/pM)$ where M is the size of the giant cluster and c_1 is determined by S_n and D_n, which are two parameters that denote the number of infected nodes in the source-flooding stage and destination-flooding stage, respectively.

2.2. Content Copies: How Many Copies Should Be Stored?

The gain of edge caching is actually larger than that of cellular communication. The performance improvement reaped via retrieving the content from the adjacent edge nodes should be derived theoretically. However, even if we consider the storage abilities of both user terminals and small base stations, a wide gap exists between the limited caching capacity and the continuously increasing data traffic from the wireless links. It is hence important to decide which content and how many content copies must be cached based on a holistic method for the maximization of profitability, i.e., we should prefetch content by integrating its potential popularity, transmission gains and locations of existing copies over the network topology. We can use the traditional knapsack algorithm to model the optimization caching problem and further convert it into a resource-allocation problem with a global

learning rate. Then, the tech-oriented solutions can be found so as to decide what to cache and how many copies to cache by analyzing the global user earning performance.

We introduce the system model shown in Figure 2, which includes a single macro cell, N_1 SBSs and N mobile UTs. The macro base station (MBS) can be regarded as a content server with the largest storage capacity, which means that it can access all files from a library. For the sake of simplicity, all files are limited to the same length. SBSs and UTs are modeled as mutually independent Poisson point distributions. The suitable content will be pre-stored to serve the requests through device-to-device (D2D) communication over short distances. The pre-store procedure will be controlled and implemented by the MBS during the off-peak hours. Each SBS is equipped with a cache of size up to m_1. Analogously, the UTs are equipped with a buffer size m.

To decide the number of copies of each file and the edge nodes that store these files before they are requested, the caching hit ratio $\Phi(C)$, i.e., the expected probability from the requesting user terminals (RUT), can be regarded as the gain function and optimal goal. From the feasible placement vector, $C = \{C_1, C_2, \cdots C_M\}$, which denotes the number of content copies in the communication area of the MBS, and p_i, which denotes the average request probability of the i-th file, the caching hit ratio can be modeled as $\Phi(C) = \sum\limits_{i=1}^{M} P_i(1 - e^{-\lambda\pi r^2 C_i})$. Note that the SBSs and UTs must have different spatial intensities with parameters λ_s and λ_u, i.e., the number of UTs distributed in the same space must be larger than that of SBSs for $\lambda_u > \lambda_s$. As the SBSs and UTs are equipped with different storage capacities, m_1 and m, respectively, the intensities of the edge nodes caching the i-th file in the communication radius of the RUT can be further described in terms of the constituent components as $\lambda = a(\lambda_s m_1 + \lambda_u m)$, where a is defined as an intensity factor of the caching probability. In fact, the mean number of SBSs and UTs in a macro-cell can be determined by their intensities λ_s and λ_u according to the theory of PPP, i.e., $N_1 = \lambda_s \pi R^2$ and $N = \lambda_u \pi R^2$.

Aiming at the maximization of the hit ratio in the local network, we transform the optimization problem from a many 0-1 knapsack problem into a complete-knapsack problem with a decomposition style. Constraints $0 \leq C_i \leq N_1 + N$ and $0 \leq \sum\limits_{i=1}^{M} C_i \leq (N_1 * m_1 + N * m)$ can be derived from the incidence matrix, since the total number of i-th file copies cannot exceed the total number of nodes, i.e., each node is permitted to store no more than one copy of the i-th file, so that the total number of file copies cached in the edge nodes should be smaller than the total storage capacities. Recall that in a complete-knapsack problem, the number of the items associated with a value function is infinite, instead of one, as in a 0-1 knapsack problem. In other words, it is possible to fetch an arbitrary number of i-th file copies cached in edge nodes within the domain. The optimal caching vector can be found by dynamic programming or various polynomial approximation algorithms. Since the optimization objective is a sum of convex functions and all the constraints are linear, the optimization problem is obviously convex. Thus, one approximation exists by solving a continuous relaxation problem with the method of Lagrangian multipliers, which is dependent on the fractional knapsack problem.

2.3. Content Allocation: Where to Store the Copies?

In this section, we introduce two solutions for caching the content in the edge nodes.

2.3.1. Cache Contents in the UTs and FAE

We propose a collaboration caching scheme, which places content in both the UTs and femtocaching auxiliary equipment (FAE) [20]. The FAE are deployed in the places where users request content frequently. It is obvious that the FAE can extend the network coverage area effectively and alleviate the pressure on the BS. In the scheme, the library files are all stored in the BS, and users can connect directly via a cellular communication link. In addition, each FAE has a buffer size of k and caches the top k most popular files without repetition. Considering the fact that UTs in the cellular networks are highly mobile, the caching strategy for UT is random caching. Each UT caches

just one file. To enhance the system performance, we use multiple D2D links to transmit content simultaneously. This is mainly because an RUT can detect a large number of suitable neighbor nodes in its coverage area. A many-to-one D2D relationship is thus built when all the suitable neighbors transmit data streams to the RUT simultaneously.

2.3.2. Cache Contents in the D2D Group

In the second scheme, we cache content in a D2D group where the D2D group consists of multiple mobile UTs that can communicate with each other through D2D links [21]. There will be many D2D groups in a cell when the nodes in the network are dense enough, to ensure that the nodes can communicate with their neighbors through D2D links with high probabilities. As mentioned before, we assume that all files have the same size and that one file is a basic storage unit. If there are i nodes in a D2D group and each node has the same buffer space to cache j files, we can cache the top $i \times j$ files with the greatest popularity in one D2D group, which means that the requests for these files can be served by D2D links. The D2D groups are refreshed after a certain interval because of the dynamics of the network topology.

3. Opportunistic Transmission

The store-carry-forward feature of opportunistic routing protocols (ORPs) is highly suitable for the mobile edge caching networks, where the D2D links change over time because of the mobility of the crowd. Though the research community has proposed many routing algorithms in the past few years, a comprehensive and extensive analysis on the current opportunistic routing protocols is lacking. We classify the current ORPs from a high-level view, as shown in Figure 4. There are three components [22]. The analysis component investigates the underlying principles of the ORPs. A quantitative analysis of zero-information ORPs, in terms of the packet delivery ratio, mean delivery delay, cost and number of hops is discussed; and a qualitative analysis of information-rich ORPs, especially the feature of the inter-contact-time is analyzed. After that, a complete performance evaluation on ORPs is implemented from the data plane and control plane perspective, respectively.
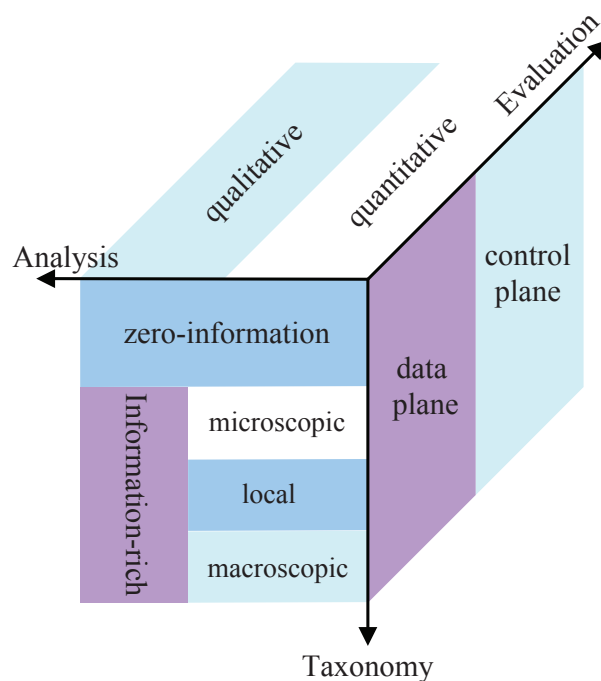

Figure 4. Taxonomy, analysis and evaluation of opportunistic routing protocols (ORPs).

3.1. Control Plane: How to Collect the Heuristic Information?

To optimize the performance of zero-information ORPs, many smart routing algorithms have emerged, including ProPHET [23], SimBet [24] and Bubble [25]. These protocols use various control information including the contact probability, social similarity and locations of nodes, so as to choose qualified relays. The method of collecting such heuristics obviously has a big influence on the routing performance. Note that while the existing works use elaborate methods to allocate packets, they collect the control information in an epidemic way. That is, any two nodes can swap the control information whenever they have a contact, resulting in a splurge on resource consumption and degrading the performance metric.

Considering this fact, we design ELECTION, a hierarchical information collection scheme, to improve the routing performance [26]. It has two important features. One is the effectiveness, which employs the packet delivery ratio to measure the degree to which the delivery is successful. The other is the efficiency, which refers to the degree of energy saving, measured by the exchange times of the control information. ELECTION first classifies nodes into core and edge nodes. The former is responsible for collecting and diffusing control information, and the latter receives control information from the former and sends their control information to the core nodes. Note that the information exchange happens only between two core nodes or between one core node and an edge node. That is, two edge nodes cannot be permitted to swap control information even if they encounter each other, so as to reduce the number of information exchanges and save network resources. ELECTION then transforms the information collection problem into the problem of minimum coverage, which is submodular. A greedy algorithm to construct the coverage set is designed and has been proven to converge the optimum solution with a probability of at least $1 - 1/e$.

3.2. Data Plane: How to Select the Desired Relay?

Another important issue in opportunistic transmission is the selection of the desired relays, so as to deliver the packets as quickly as possible. Considering the crowd feature in mobile edge systems, many social-based ORPs have been proposed, such as the PeopleRank [27] and Bubble [25]. They choose nodes with global centralities in the network to forward packets, neglecting the local importance of the nodes in different groups. By analyzing the real datasets KAISTand NCSU, we observed that people have different mean numbers of contacts in different social groups. The same person could be sociable in one clique and have many contacts, while being taciturn in another clique. Such crowd behaviors have also been found in social networks, where family members or friends contact frequently, while people accidentally encounter strangers. If one tries to characterize such dynamics by using the statistics of the average contact number of a person, we would be missing the important network features including the epidemics dynamics, leading to a biased understanding of human mobility behavior [28]. Figure 5 shows the roles of two different social features (i.e., global centrality and relative importance) in packet diffusion, by gradually removing the same number of the two kinds of nodes. Two important phenomena are observed. The first one is that relatively important nodes play a big role in the mean delivery delay. The transmission delay increases by almost 25% if removing the relatively important nodes. This finding contradicts the results in previous works [29]. The second one is that there exists a sudden sharp increase in the transmission delay if the source and destination belong to a community. When 25%–30% of the relatively important nodes are removed, the transmission delay increases suddenly. Meanwhile, removing the popular nodes alone causes a relatively stable increase in the transmission delay. As a result, the relative importance metric reveals fine-grained relations among nodes. Therefore, it is helpful to make smart forwarding decisions (e.g., if a node shows a high relative importance to the destination's community partners, it is a desired relay). Based on the results, we employ the graph spectrum theory to calculate the relative importance metric. We then design a packet forwarding algorithm by integrating both the relative importance and the community structure, so as to improve the routing efficiency.
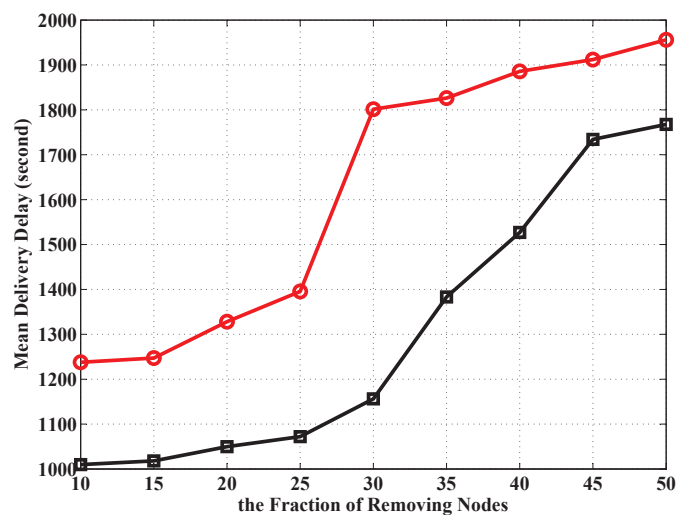
Figure 5. Nodes' roles in content delivery with NCSUdatasets. The figure corresponds to the case where nodes are removed based on their relative importance or centrality. The red curves with circles represent the results when removing first the high relative importance nodes, moving toward the lower nodes. The black curves with rectangles show the case where nodes with high global centrality are first removed.

4. Numerical Results

In this section, we evaluate the proposed framework [20] considering the user mobility under different numbers of nodes and storage sizes. Figure 6a,b show the hit rate of four caching algorithms. In addition to the caching algorithm in [10], we also contrast our collaborative caching scheme against two classical caching algorithms: random caching and the most popular caching based on the file popularity. The algorithm in [10] combines the random caching with the most popular aching, i.e., fixed nodes like SBSs use the most popular algorithm, while UTs use random storage due to the mobility. It can be observed that the proposed collaborative system brings greater caching gains owing to the dynamic and flexible characteristics, i.e., the edge nodes tend to have more great autonomy in collaborative caching. Comparing Figure 6a with Figure 6b, we can see that the hit rate with a different quantity level exactly coincides with that of the storage size for both the proposed algorithm and the random algorithm.
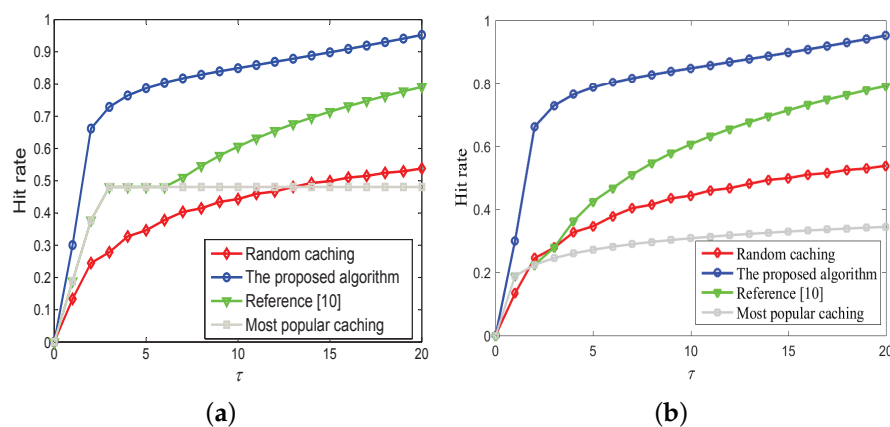

(**a**)　　　　　　　　　　　　　　　　　　　(**b**)

Figure 6. The hit rate of four caching algorithms under different settings. (**a**) Different quantity level; (**b**) different storage size.

5. Conclusions

This article discussed the opportunities and challenges in an MEC system brought about by crowd involvement. In particular, we investigated the opportunistic characteristics of crowd mobility and sociability. From the caching perspective, we provided a theoretical bound on the buffer size and analyzed the copies needed for caching the content. We also presented two solutions to address how the content was placed in edge nodes. From the transmission perspective, we provided a high-level view to classify the current ORPs and analyzed the influence of human sociability on the routing performance from both the control plane and data plane. There are many open issues in this emerging research area, including privacy and security in the case of human involvement. An effective incentive scheme will be advantageous to encourage more people/devices to participate in this system. In addition, the machine learning-based recommendation strategies are also helpful to predict users' requesting behavior.

Author Contributions: Conceptualization, P.Y. and X.Z.; Validation, X.P.; Writing-review & Editing, P.Y.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ma, H.; Liu, L.; Zhou, A.; Zhao, D. On networking of internet of things: Explorations and challenges. *IEEE Internet Things J.* **2016**, *3*, 441–452. [CrossRef]
2. Huang, J.; Xing, C.C.; Wang, C. Simultaneous Wireless Information and Power Transfer: Technologies, Applications, and Research Challenges. *IEEE Commun. Mag.* **2017**, *55*, 26–32. [CrossRef]
3. Wang, X.; Chen, M.; Taleb, T.; Ksentini, A.; Leung, V. Cache in the air: exploiting content caching and delivery techniques for 5G systems. *IEEE Commun. Mag.* **2014**, *52*, 131–139. [CrossRef]
4. Bilal, K.; Khalid, O.; Erbad, A.; Khan, S.U. Potentials, trends, and prospects in edge technologies: Fog, cloudlet, mobile edge, and micro data centers. *Comput. Netw.* **2018**, *130*, 94–120. [CrossRef]
5. Yang, C.; Yao, Y.; Chen, Z.; Xia, B. Analysis on Cache-Enabled Wireless Heterogeneous Networks. *IEEE Trans. Wirel. Commun.* **2016**, *15*, 131–145. [CrossRef]
6. Ma, H.; Zhao, D.; Yuan, P. Opportunities in mobile crowd sensing. *IEEE Commun. Mag.* **2014**, *52*, 29–35. [CrossRef]
7. Ma, H.; Liu, Y. On coverage problems of directional sensor networks. In Proceedings of the International Conference on Mobile Ad-Hoc and Sensor Networks, Wuhan, China, 13–15 December 2005; pp. 721–731.
8. Yuan, P.; Ma, H.; Fu, H. Hotspot-entropy based data forwarding in opportunistic social networks. *Pervasive Mob. Comput.* **2015**, *16*, 136–154. [CrossRef]
9. Luo, H.; Luo, J.; Liu, Y.; Das, S.K. Adaptive data fusion for energy efficient routing in wireless sensor networks. *IEEE Trans. Comput.* **2006**, *55*, 1286–1299.
10. Zhao, X.; Yuan, P.; Chen, Y.; Chen, P. Femtocaching assisted multi-source D2D content delivery in cellular networks. *EURASIP J. Wirel. Commun. Netw.* **2017**, *2017*, 125. [CrossRef]
11. Gerla, M.; Tsai, J.T.C. Multicluster, mobile, multimedia radio network. *Wirel. Netw.* **1995**, *1*, 255–265. [CrossRef]
12. Lin, C.R.; Gerla, M. Adaptive clustering for mobile wireless networks. *IEEE J. Sel. Areas Commun.* **1997**, *15*, 1265–1275. [CrossRef]
13. Tsiropoulou, E.E.; Paruchuri, S.T.; Baras, J.S. Interest, energy and physical-aware coalition formation and resource allocation in smart IoT applications. In Proceedings of the 2017 51st Annual Conference on Information Sciences and Systems (CISS), Baltimore, MD, USA, 22–24 March 2017; pp. 1–6.
14. Tsiropoulou, E.E.; Mitsis, G.; Papavassiliou, S. Interest-aware energy collection & resource management in machine to machine communications. *Ad Hoc Netw.* **2018**, *68*, 48–57.

15. Zhao, X.; Yuan, P.; Chen, Y.; Chen, P. Cooperative D2D for Content Delivery in Heterogeneous Networks. In Proceedings of the International Conference on Big Data Computing and Communications, Chengdu, China, 10–11 August 2017; pp. 265–271.

16. Becchetti, L.; Clementi, A.; Pasquale, F.; Resta, G.; Santi, P.; Silvestri, R. Flooding time in opportunistic networks under power law and exponential intercontact times. *IEEE Trans. Parallel Distrib. Syst.* **2014**, *25*, 2297–2306. [CrossRef]

17. Yuan, P.; Wu, H.; Zhao, X.; Dong, Z. Percolation-theoretic bounds on the cache size of nodes in mobile opportunistic networks. *Sci. Rep.* **2017**, *7*, 5662. [CrossRef] [PubMed]

18. Yuan, P.; Wu, H.; Zhao, X.; Dong, Z. Cache Potentiality of MONs: A Prime. In Proceedings of the IEEE International Conference on Distributed Computing Systems, Atlanta, GA, USA, 5–8 June 2017; pp. 2650–2653.

19. Zhu, H.; Huang, C.; Li, H. Information diffusion model based on privacy setting in online social networking services. *Comput. J.* **2014**, *58*, 536–548. [CrossRef]

20. Zhao, X.; Yuan, P.; Li, H.; Tang, S. Collaborative edge caching in context-aware D2D networks. *IEEE Trans. Veh. Technol.* **2018**. [CrossRef]

21. Fan, L.; Dong, Z.; Yuan, P. The capacity of Device-to-device communication underlaying cellular networks with relay links. *IEEE Access* **2017**, *5*, 16840–16846. [CrossRef]

22. Yuan, P.; Fan, L.; Liu, P.; Tang, S. Recent progress in routing protocols of mobile opportunistic networks. *J. Netw. Comput. Appl.* **2016**, *62*, 163–170. [CrossRef]

23. Lindgren, A.; Doria, A.; Schelen, O. Probabilistic routing in intermittently connected networks. In *Service Assurance with Partial and Intermittent Resources*; Springer: Berlin, Germany, 2004; pp. 239–254.

24. Daly, E.M.; Haahr, M. Social network analysis for information flow in disconnected delay-tolerant MANETs. *IEEE Trans. Mob. Comput.* **2009**, *8*, 606–621. [CrossRef]

25. Hui, P.; Crowcroft, J.; Yoneki, E. Bubble rap: Social-based forwarding in delay-tolerant networks. *IEEE Trans. Mob. Comput.* **2011**, *10*, 1576–1589. [CrossRef]

26. Yuan, P.; Song, M.; Zhao, X. Effective and efficient collection of control messages for opportunistic routing algorithms. *J. Netw. Comput. Appl.* **2017**, 125–130. [CrossRef]

27. Mtibaa, A.; May, M.; Diot, C.; Ammar, M. Peoplerank: Social opportunistic forwarding. In Proceedings of the Infocom, San Diego, CA, USA, 14–19 March 2010; pp. 1–5.

28. Yuan, P.; Liu, P.; Tang, S. RIM: Relative-importance based data forwarding in people-centric networks. *J. Netw. Comput. Appl.* **2016**, *62*, 100–111. [CrossRef]

29. Pietilänen, A.K.; Diot, C. Dissemination in opportunistic social networks: the role of temporal communities. In Proceedings of the Thirteenth ACM international Symposium on Mobile Ad Hoc Networking and Computing, Hilton Head, SC, USA, 11–14 June 2012; pp. 165–174.