IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received March 20, 2018, accepted April 15, 2018, date of publication May 10, 2018, date of current version June 20, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2835368

LOCASS: Local Optimal Caching Algorithm With Social Selfishness for Mixed Cooperative and Selfish Devices

YANG YANG[1,2,3], (Fellow, IEEE), YECHENG WU[1,3], NANXI CHEN[1], KUNLUN WANG[1], SHANZHI CHEN[4], (Senior Member, IEEE), AND SHA YAO[5]

[1]Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences, Shanghai 200050, China
[2]Shanghai Institute of Fog Computing Technology, SIST, ShanghaiTech University, Shanghai 201210, China
[3]University of Chinese Academy of Sciences, Beijing 100049, China
[4]State Key Laboratory of Wireless Mobile Communications, China Academy of Telecommunication Technology, Beijing 100083, China
[5]Enigma Investment, Shanghai 201210, China

Corresponding author: Yang Yang (yang.yang@wico.sh)

This work was supported in part by the Science and Technology Commission of Shanghai Municipality under Grant 16510711600 and in part by the National Natural Science Foundation of China under Grant 61571004.

ABSTRACT Device caching has emerged as a promising solution to alleviate backhaul overload in future wireless networks with mixed cooperative and selfish devices. The behaviors of these devices actually represent the inherent social-network characteristics of their users, i.e., people treat each other differently according to the closeness of their social relationships. In this paper, the concept of social selfishness is adopted to capture the social characteristics in mobile device caching. Devices can be cooperative or be selfish in dynamic content sharing environments according to their social tie strengths, just like their users tend to cooperate with their friends and show selfishness to strangers. Based on social selfishness, a novel device caching game model is proposed to analyze the limited resource and privacy issues in a typical mobile caching scenario. Then, a local optimal caching algorithm with social selfishness (LOCASS) is developed to address these challenging problems. Analytical results show that LOCASS can approach Nash equilibrium of the game, and therefore, achieving the best caching strategy for mixed cooperative and selfish devices. Further, extensive simulation results show that LOCASS offers much better performance, in terms of average offloading ratio and resource utilization, than traditional random, Most-popular-content and Greedy caching algorithms. Besides, under LOCASS, devices with low degrees are most likely to store popular contents, while devices with high degrees are more willing to store those comparatively unpopular contents and fetch popular ones from their cooperative-devices.

INDEX TERMS Device caching, social selfishness, offloading ratio, Nash equilibrium.

I. INTRODUCTION

Over the past decade, mobile data traffic has been increasing dramatically due to the explosive growth of the mobile devices and services [1]. According to the Cisco report [2], the global mobile traffic is expected to continue to grow at a compound growth rate of 47 percent from 2016 to 2021. Such explosive data has exerted a heavy burden on current network structure [3], and the capacity of cellular networks should be enhanced by adopting new technologies [4].

Recently, caching at devices with device-to-device (D2D) communication is considered to be a promising solution to cope with the explosive data [6]. With device caching, popular contents can be proactively stored in local devices' caches during off-peak time and shared directly with D2D links during the peak time, which can offload base stations' traffic and alleviate backhaul overload simultaneously [1], [3]. With a denser distribution of mobile devices [5], [8], larger cache space can be accessed nearby and a greater proportion of requests can be satisfied. By utilizing D2D communication technology, spatial reuse gain can be also achieved by constructing multiple direct links simultaneously [9]. Although significant performance can be achieved by device caching, there are also great challenges in terms of:

* Limited resource: due to the resource constraints, e.g., limited battery and cache capacity [7], devices are unlikely to sacrifice their own resources to serve others.

So that devices' caching strategy tends to maximize their own profit rather than that of the system. Hence, such strategy may deviate from network-wide optimal policy and significantly degrade the system performance.
* Privacy: cached contents in individual devices can be viewed as one kind of privacy, similar with individuals' interest and mobility information. The private information should be known only by others who have the right to access. This leads to incomplete information being utilized when each device makes its own caching strategy.

Traditional device caching technologies, such as fully cooperative caching and fully selfish caching [11], [14]–[19], cannot effectively cope with the above challenges. For example: in the fully cooperative caching mechanism, each device altruistically shares its cached contents with others, and complete information is utilized to make caching strategy, which consumes a large quantity of resources for content sharing and have a risk of privacy exposure; in the fully selfish caching mechanism, each device makes caching strategy just to maximize its own profit based on individual private information, which poses poor system performance.

Normally, devices of interest are carried and controlled by humans so that devices of interest are naturally equipped with inherent social characteristics, which open up a new avenue for device caching design and facilitating the solution of caching strategies. There have been lots of works introducing social characteristics to save the limited resource in device caching. In [20], compared with accessing contents from others with a close relationship, devices should pay higher bids for strangers. Wang *et al.* [21] divided the local cache space of a device into its own space and the space for friends where caching strategy for the latter is just related to the interest of its friends, while the strategy of preferentially storing and forwarding the content to friends was proposed in [18]. In [22], device can be chosen as a cache node to serve its physical and social neighbors. However, the aforementioned works considered less in protecting individuals' private information, including cached content, interest and mobility information, which are visible to all devices, especially for strangers. Although privacy is an important issue in device caching [12], limited works focus on privacy protection in device caching. Jung and Park [24] propose a privacy-preserving architecture to protect devices' location information. In [25], a privacy-preserving protocol is proposed to render such a caching system well protected against all kind of internal or external privacy breaches. Most works like [23]–[25] adopt virtual central servers to collect all devices' private information and make the caching strategy. Those virtual centralized servers don't have any social relationship and trust with devices, which also poses a risk of privacy exposure.

In order to address the challenges imposed by limited device capacities and privacy issue, this paper adopts the concept of "*social selfishness*" [13] and proposes a novel device caching game to achieve the optimal caching strategy for mixed cooperative and selfish devices. Social selfishness can be viewed as a mixture of cooperation and selfishness behaviors of a device, i.e., it provides different levels of trust and service to others according to the closeness of their social relationship. Similarly, in a human society, people tend to cooperate and show trust to their *friends*, but to act in an opposite way to *strangers* who have no or weak social relationships with them.

Specifically, our main contributions are as follows:
* A device caching game with social selfishness is proposed to address the limited resource and privacy issues in a typical scenario with mixed cooperative and selfish devices. Only cooperative-devices trust and cooperate with each other, i.e., they share their information, capacities and contents for saving time, energy and space in local services. On the other hand, selfish-devices do not share any resources or information for protecting their privacies.
* Based on the device caching game, a Local Optimal Caching Algorithm with Social Selfishness (LOCASS) is developed for achieving the best caching strategy for mixed cooperative and selfish-devices. By introducing social-related factors, LOCASS can effectively share resources and contents among cooperative-devices, while protecting privacies between selfish-devices.
* Theoretical analysis proves the existence of Nash Equilibrium in LOCASS. In addition, extensive simulation results show that LOCASS can offer much better performance, in terms of average offloading ratio and resource utilization, than traditional random, Most-Popular-Content (MPC) and Greedy caching algorithms.

The remainder of this paper is organized as follows. The device caching game is proposed in Section II. In Section III, the existence of pure Nash Equilibrium is proved in this game and a caching algorithm is proposed subsequently. The evaluation results are given in Section IV. Finally, Section V concludes the paper.

II. SYSTEM MODEL AND PROBLEM FORMULATION

A. NETWORK MODEL

We consider a wireless caching network consisting of a set of M cache-enabled devices $\mathcal{M} = \{1, 2, \ldots, M\}$. Those devices request contents from the library with F contents $\mathcal{F} = \{1, 2, \ldots, F\}$. The request probability of device m for the content f is $p_{m,f}$. Each device m is equipped with a certain cache space with size s_m. And each content $f \in \mathcal{F}$ has a size r_f, which can be divided into multiple segments to store in those cache-enabled devices. Each pair of devices (m, m') is equipped with a social tie strength $w_{m,m'}$ which range in interval [0,1]. Social tie strength is utilized to measure the closeness of social relationship for any pair of device. Each device cares more about those cooperative-devices with a larger tie strength and there is a larger probability to contribute its own cache space to store contents for them. The value of social tie strength is related to the relationship in the real life. For example, the social tie strength between family members is usually larger than that of colleagues. The social tie strength

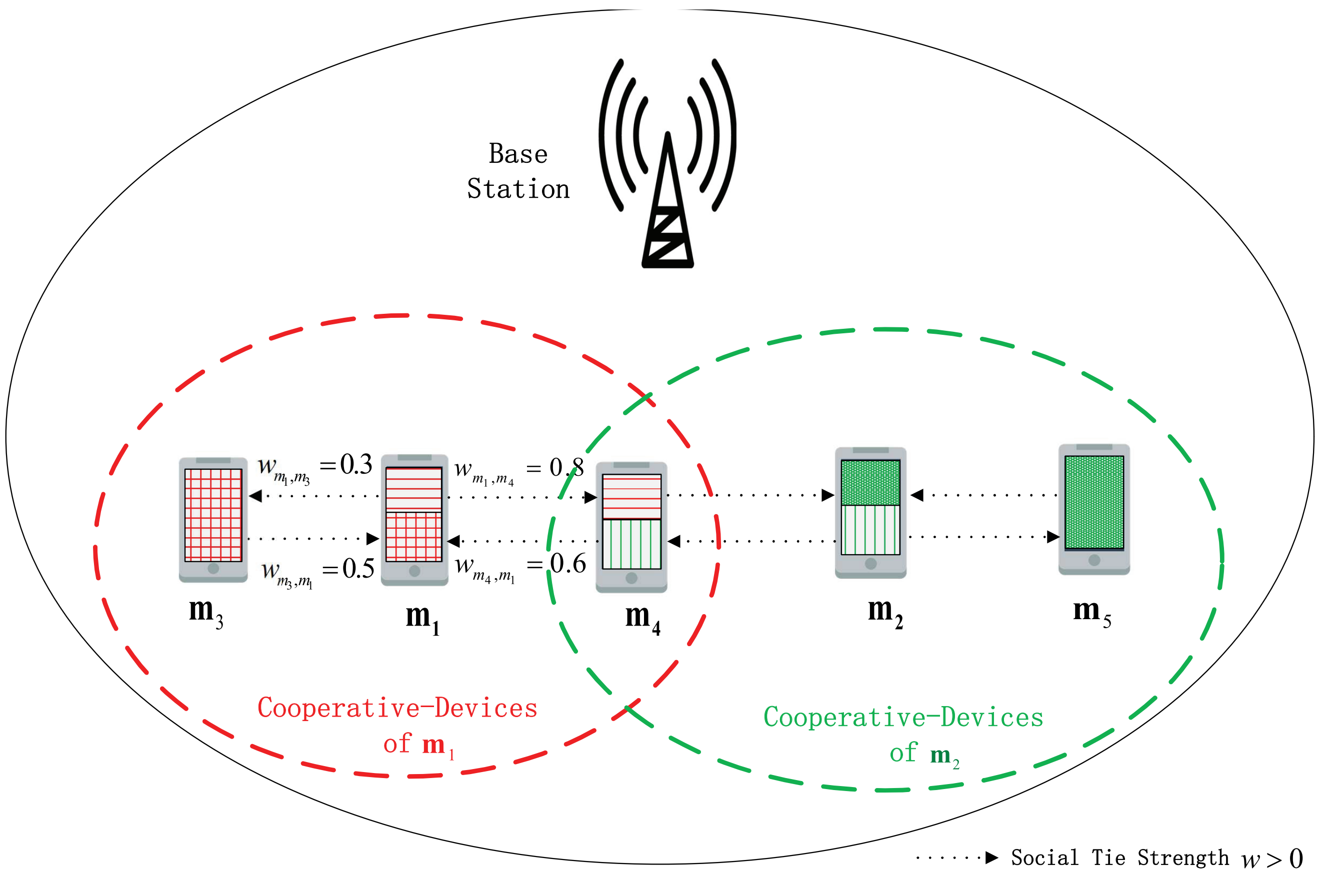


FIGURE 1. An illustration of system model, where each pair of cooperative-devices is equipped with a social tie strength larger than 0.

of M devices are specified by the social tie strength matrix $W_{M\times M}$. If $w_{m,m'} > 0$ and $w_{m',m} > 0$, they are considered as *cooperative-devices* with each other. Otherwise, they are considered as *selfish-devices*. The number of cooperative-devices for the device m is considered as its *degree* which is represented by N_m [10]. Here, we assume that it doesn't exist $w_{m,m'} = 0$ and $w_{m',m} > 0$ for any pair of devices (m, m'). Each pair of cooperative-devices is likely to cooperate and show trust with each other while each pair of selfish-devices is not. If each pair of cooperative-devices in proximity, D2D links can be established to share their cached contents before deadline time T_D. The system scenario is shown in Fig.1. In this section, we propose a device caching game with the concept of social selfishness for the mixed cooperative and selfish devices.

B. SOCIAL SELFISHNESS-BASED UTILITY

We firstly adopt an inter-contact mobility model [15] to capture devices' probabilistic mobility. Within a time unit, the number of contacts between the device m and the device m' obeys the Poisson process with parameter $\lambda_{m,m'}$. So, within the deadline time T_D, the expected number of contacts between the device m and the device m' is $T_D\lambda_{m,m'}$. For the device m and the device m', we assume that the expected duration is $\theta_{m,m'}$ for a single contact. Within deadline time T_D, the expected whole D2D communication duration $T_{m,m'}$ between the device m and the device m' can be expressed as follows:

$$T_{m,m'} = T_D\lambda_{m,m'}\theta_{m,m'}. \tag{1}$$

With the transmission rate $r_{m,m'}$ between the device m' and the device m, the device m can receive at most $T_{m,m'}r_{m,m'}$ segments from the device m' within deadline time T_D.

Content f can be recovered if the device m collects at least r_f segments within deadline time T_D. Otherwise, the device m will request the remaining segments from remote content server via the base station and the backhaul. Within T_D, for the device m, the expected amount of the segments of the content f that collected from the device m' is specified by the notation $o_{m,m',f}$ which can be written as:

$$o_{m,m',f} = \min\left\{T_{m,m'}r_{m,m'}, c_{m',f}\right\}, \tag{2}$$

where $c_{m',f}$ is the cached number of segments for the content f in the device m'. Obviously, the collected number of segments should not be greater than the number of segments cached in the device m'.

For the device m, the amount of segments collected through device caching can be expressed as:

$$o_{m,f} = \min\left\{\sum_{m' \in \mathcal{M}} o_{m,m',f}, r_f\right\}. \quad (3)$$

In this equation, $o_{m,f}$ should not be greater than r_f, which means that the device m only collect the segments that just recover the content f. Next, we introduce the concept of *offloading ratio* to reflect the percent of traffic which can be offloaded from the base station via device caching. The offloading ratio of the device m can be represented as:

$$\begin{aligned} O_m(\mathbf{c}) &= \sum_{f \in F} p_{m,f} o_{m,f} \\ &= \sum_{f \in F} p_{m,f} \frac{1}{r_f} \min\left\{\sum_{m' \in \mathcal{M}} \min\left\{T_{m,m'} r_{m,m'}, c_{m',f}\right\}, r_f\right\}. \end{aligned} \quad (4)$$

The offloading ratio O_m is also considered as the *individual utility* of the device m. From the system's perspective, the *average offloading ratio* is defined as:

$$O(\mathbf{c}) = \frac{1}{M} \sum_{m \in \mathcal{M}} O_m(\mathbf{c}). \quad (5)$$

Average offloading ratio represents the percent of requests which can be offloaded from the base station and served locally via device caching rather than base station from the system view.

Next, we introduce the *social selfishness-based utility* on mobile device caching. As a mixture of selfishness and cooperation, social selfishness-based utility involves two major aspects, the *access admission mechanism* and the *social group utility mechanism*.

Access admission mechanism denotes that only cooperative-devices can be admitted to access the cached contents of a device via D2D links, while selfish-devices cannot. With the access admission mechanism, no content sharing services are provided to selfish-devices, which can save the limited resources. Meanwhile, devices with social selfishness are more likely to trust with their cooperative-devices rather than selfish-devices. Each device only allows its cooperative-devices to access its cached contents, which can also prevent such private information of cached contents from the potential exposure to selfish-devices. For example, in Fig.1, device m_1 and device m_3 are cooperative-devices with each other, and they can share cached contents through D2D link. Device m_1 and device m_2 are selfish-devices with each other, and they cannot construct D2D links for sharing. So the offloading ratio of the device m with the access admission mechanism can be redefined as follows:

$$\begin{aligned} &O_m(\mathbf{c}) \\ &= \sum_{f \in \mathcal{F}} p_{m,f} \frac{1}{r_f} \min\left\{\sum_{m' \in \mathcal{N}_m \cup m} \min\left\{T_{m,m'} r_{m,m'}, c_{m',f}\right\}, r_f\right\}, \end{aligned} \quad (6)$$

where the device m can only access cached contents from cooperative-devices set $\mathcal{N}_m$. Here, N_m, the number of elements in $\mathcal{N}_m$, is also considered as the degree of the device m. For example, in Fig.1, N_{m_1}, the degree of device m_1 is 2. Commonly, devices with a high degrees have lots of cooperative-devices and a high social statue. The *social density* of cooperative-devices in the system can be shown as follows:

$$\eta = \frac{\sum_{m \in \mathcal{M}} N_m}{N(N-1)}. \quad (7)$$

Larger η means that there are more cooperative-devices in the system. When $\eta = 0$, there are no cooperative-devices in the system, and each device cannot access cached contents from any other devices. When $\eta = 1$, each pair of devices can access cached contents from any other devices. For example, in Fig.1, the social density η is 0.4.

To future motivate devices to store contents for cooperative-devices, we introduce *social group utility mechanism* [26]. Social group utility mechanism contains two components, i.e., individual utility and *cooperative utility*. The cooperative utility of the device m is the weighted sum of the individual utilities of the cooperative-devices in $\mathcal{N}_m$. As mentioned before, the individual utility of device m can be regarded as the offloading ratio $O_m(c)$.

Combining the social selfishness-based access admission mechanism and the social group utility mechanism, the *social selfishness-based utility* of the device m can be in Eq.(8), as shown at the bottom of this page. For the device m, taking

$$\begin{aligned} U_m(\mathbf{c}) &= \underbrace{O_m(\mathbf{c})}_{individual\ utility} + \underbrace{\sum_{m' \in \mathcal{N}_m} w_{m,m'} O_{m'}(\mathbf{c})}_{cooperative\ utility} \\ &= \sum_{m' \in \mathcal{N}_m \cup m} w_{m,m'} \sum_{f \in \mathcal{F}} p_{m',f} \frac{1}{r_f} \min\left\{\sum_{m'' \in \mathcal{N}_{m'} \cup m'} \min\left\{T_{m',m''} r_{m',m''}, c_{m'',f}\right\}, r_f\right\} \\ &= \sum_{m' \in \mathcal{N}_m \cup m} w_{m,m'} \sum_{f \in \mathcal{F}} p_{m',f} \frac{1}{r_f} \min\left\{T_{m',m} r_{m',m}, c_{m,f}, r_f - \sum_{m'' \in \mathcal{N}_{m'} \setminus m} \min\left\{T_{m',m''} r_{m',m''}, c_{m'',f}\right\}\right\}. \end{aligned} \quad (8)$$

the cooperative utility into consideration, social selfishness-based utility can motivate devices to store contents for cooperative-devices. Social tie strength w is utilized to measure the closeness of social relationship for any pair of device. Each device cares more about those cooperative-devices with a larger tie strength and there is a larger probability to contribute its own cache space to store contents for them. The value of social tie strength is related to the relationship in the real life. For example, the social tie strength between family members is usually larger than that of colleagues. Considering the cooperative-devices with different social tie strengths, device m cares more about those cooperative-devices with a larger tie strength and there is a larger probability to contribute its own cache space to store contents for them. For example, in Fig.1, device m_1 more cares the requests of device m_4 than that of device m_3 due to the reason that $w_{1,4} > w_{1,3}$. Meanwhile, without taking selfish-devices' utilities into consideration, the social group utility mechanism can also save the limited caching capacity just for cooperative-devices' requests. To avoid exposing private information to selfish-devices, when the device m makes caching strategy decision $\mathbf{c}_m$, only the mobility information $T_{m',m}$ and the request information $p_{m',f}$ for $\forall m' \in \mathcal{N}_m$, $f \in \mathcal{F}$ about cooperative-devices can be accessed by the device m. Additionally, in our proposed model, the private information about cooperative-device $m'' \in \mathcal{N}_{m'}$ will not be exposed to the device $m \in \mathcal{N}_{m'}$. A content requester $m' \in \mathcal{N}_m$ just need to send the quantity of required segments to the device m, i.e., $r_f - \sum_{m'' \in \mathcal{N}_{m'} \setminus m} \min\left\{T_{m',m''} r_{m',m''}, c_{m'',f}\right\}$, rather than the exact cached content information about the cooperative-device m'' for $\forall m'' \in \mathcal{N}_{m'} \setminus m$. For example, in Fig.1, the device m_4 won't leak the private information of the device m_2 to the device m_1.

Therefore, with the access admission mechanism and social group utility mechanism, the proposed social selfishness-based utility can effectively cope with the limited resource and privacy issues.

C. DEVICE CACHING GAME FORMULATION

With the social selfish-based utility, we propose a device caching game

$$\Psi = \left\{\mathcal{M}, \{\mathcal{C}_m\}_{m\in\mathcal{M}}, \{U_m\}_{m\in\mathcal{M}}\right\}. \tag{9}$$

The devices set $\mathcal{M}$ also represents the player set in the game. $\mathcal{C}_m$ depicts the caching strategy action space for the player m that complies with the cache space size s_m. And U_m is the social selfishness-based utility of the player m. In this game Ψ, each device m chooses its own caching strategy $\mathbf{c}_m \in \mathcal{C}_m$ aiming at maximizing its social selfishness-based utility U_m. The concept of Nash Equilibrium (NE) is introduced below.

Definition 1 (Nash Equilibrium): Let $\mathbf{c}^* \triangleq \left[\mathbf{c}_1^*, \mathbf{c}_2^*, \ldots, \mathbf{c}_M^*\right]$ be the solution for the game Ψ. Then the point $\mathbf{c}^*$ is a Nash Equilibrium for the proposed game Ψ if for any $\mathbf{c}_m \in \mathcal{C}_m$, the following condition is satisfied:

$$U_m(\mathbf{c}_m^*, \mathbf{c}_{-m}^*) \geq U_m(\mathbf{c}_m, \mathbf{c}_{-m}^*), \quad \forall m, \mathbf{c}_m \in \mathcal{C}_m. \tag{10}$$

The caching strategy of the device m can be specified by the vector $\mathbf{c}_m = [c_{m,1}, c_{m,2}, \ldots, c_{m,F}]$ where $c_{m,f}$ represents the amount of the segments cached in the device m for the content f. $\mathbf{c}_{-m}^*$ is the caching strategies of all devices except device m. Considering the constraint on the cache size s_m, the caching strategy space of the device m can be written as:

$$\mathcal{C}_m = \left\{\mathbf{c}_m \middle| \sum_{f\in\mathcal{F}} c_{m,f} \leq s_m\right\}. \tag{11}$$

In this game, each device m chooses its own caching strategy $\mathbf{c}_m^*$ to maximize its social selfishness based utility U_m:

$$c_m^* = \arg\max_{\mathbf{c}_m\in\mathcal{C}_m} U_m(\mathbf{c}_m, \mathbf{c}_{-m}^*) \tag{12}$$

III. LOCASS: LOCAL OPTIMAL CACHING ALGORITHM WITH SOCIAL SELFISHNESS

In this section, we propose a local optimal caching algorithm with social selfishness for the device caching game. To design the algorithm, we first prove the existence of Nash Equilibriums. We also prove that each device's social selfishness-based utility is a piecewise concave function. For optimizing the piecewise concave function, we employ the concept of right derivative. With the right derivative, we propose the local optimal caching algorithm with social selfishness. Device $m \in \mathcal{M}$ takes its own caching strategy $\mathbf{c}_m$ to maximize its own social selfishness based utility U_m.

Lemma 1: for $\forall m \in \mathcal{M}$, U_m is a concave function on $\mathbf{c}$.

Proof: : The detailed proof is appended in Appendix. ■

Proposition 1: There exists pure Nash Equilibrium for the device caching game.

Proof: Obviously, each caching strategy $\mathbf{c}_m$ is a closed bounded convex set. Meanwhile, according to the *Lemma 1*, for the device m, the utility function U_m is a concave function on $\mathbf{c}$. The utility function U_m is also continuous in $\mathbf{c}$. Therefore, the caching game is a concave game [28]. By the Schauder fixed-point theorem [29], the existence of Nash equilibrium in caching game is proved. ■

The social selfishness based utility of the device m can be rewritten as

$$U_m\{\mathbf{c}_m, \mathbf{c}_{-m}) = \sum_{f\in\mathcal{F}} \sum_{m'\in\mathcal{N}_m\cup m} \left\{ w_{m,m'} p_{m',f} \frac{1}{r_f} \right.$$
$$\left. \times \min\left\{B_{m',m,f} + T_{m',m} r_{m',m}, B_{m',m,f} + c_{m,f}, r_f\right\}\right\}, \tag{13}$$

where $B_{m',m,f} = \sum_{m"\in\mathcal{N}_{m'}\cup m'\setminus m} \min\left\{T_{m',m"} r_{m',m"}, c_{m",f}\right\}$ represents the quantity of the content f that device m' can access from the cooperative-devices set $\mathcal{N}_{m'}$ or from itself except the device m before reach the deadline T_D. When the device m takes its own caching strategy $\mathbf{c}_m$ to maximize its utility

$$
\begin{aligned}
&\left(u_m^f\right)'_+(c_{m,f}, c_{-m,f}) \\
&= \lim_{c^0_{m,f}\to c_{m,f}+} w_{m,m'}p_{m',f}\frac{1}{r_f}\frac{\sum\limits_{m'\in\mathcal{N}_m\cup m} L^m_{m'}\left(c^0_{m,f}, c_{-m,f}\right)\left(B_{m',m,f}+c^0_{m,f}\right)-\sum\limits_{m'\in\mathcal{N}_m\cup m} L^m_{m'}\left(c_{m,f}, c_{-m,f}\right)\left(B_{m',m,f}+c_{m,f}\right)}{c^0_{m,f}-c_{m,f}} \\
&= \sum_{m'\in\mathcal{N}_m\cup m \text{ and } L^m_{m'}(c_{m,f}, c_{-m,f})=1}\left(w_{m,m'}p_{m',f}\frac{1}{r_f}\right). \qquad (17)
\end{aligned}
$$

$$
u_m^f(c_{m,f}, c_{-m,f}) = \begin{cases} u_m^f(0, c_{-m,f}) + \left(u_m^f\right)'_+\left(0, c_{-m,f}\right)c_{m,f}, & 0 \le c_{m,f} < c^{rank_1}_{m,f} \\ \dots \\ u_m^f(c^{rank_{i-1}}_{m,f}, c_{-m,f}) + \left(u_m^f\right)'_+\left(c^{rank_{i-1}}_{m,f}, c_{-m,f}\right)\left(c_{m,f}-c^{rank_{i-1}}_{m,f}\right), & c^{rank_{i-1}}_{m,f} \le c_{m,f} < c^{rank_i}_{m,f} \\ \dots \\ u_m^f(c^{rank_{R_{m,f}}}_{m,f}, c_{-m,f}), & c^{rank_{R_{m,f}}}_{m,f} \le c_{m,f} \le r_{m,f} \end{cases} \qquad (20)
$$

function $U_m(c_m, c_{-m})$, and $B_{m',m,f}$ is a constant. The utility of the content f for the device m is defined as

$$
u_m^f(c_{m,f}, c_{-m,f}) = \sum_{m'\in\mathcal{N}_m\cup m}\left\{w_{m,m'}p_{m',f}\frac{1}{r_f} \times \min\left\{B_{m',m,f}+T_{m',m}r_{m',m}, B_{m',m,f}+c_{m,f}, r_f\right\}\right\}. \qquad (14)
$$

Thus, the *right derivative* of u_m^f on $c_{m,f}$ is denoted as

$$
u'^f_{m+}(c_{m,f}, c_{-m,f}) = \lim_{c^0_{m,f}\to c_{m,f}+}\frac{u_m^f(c^0_{m,f}, c_{-m,f}) - u_m^f(c_{m,f}, c_{-m,f})}{c^0_{m,f}-c_{m,f}}, \qquad (15)
$$

where $\left(u_m^f\right)'_+$ reflects the utility improvement if an additional unit amount of the content f is stored in the device m.

To optimize the social selfishness-based utility, we first introduce the function

$$
L^m_{m'}\left(c_{m,f}, c_{-m,f}\right) = \begin{cases} 1, & \text{if } c_{m,f} < T_{m',m}r_{m',m} \text{ or } c_{m,f} < r_f - B_{m',m,f} \\ 0, & \text{otherwise} \end{cases} \qquad (16)
$$

The value of this function equals to 1 when the amount of cached segments $c_{m,f}$ can be successfully transmitted between the device m and the device m' within the deadline or help the device m' to successfully retrieve the content f. Additionally, when $L^m_{m'}$ equals to 1, additional unit amount of the content f stored in the device m could bring utility improvement. We can also find $L^m_{m'}\left(c_{m,f}, c_{-m,f}\right)$ is non-decreasing and right continuous on $c_{m,f}$. So, given the definition of $L^m_{m'}$, $\left(u_m^f\right)'_+$ can be reformulated in Eq.(17), as shown at the top of this page.

Because $L^m_{m'}\left(c_{m,f}, c_{-m,f}\right)$ is a non-increasing function on $c_{m,f}$, $\left(u_m^f\right)'_+(c_{m,f}, c_{-m,f})$ is a non-increasing function on $c_{m,f}$ as well. We then denote the set of the turning point $Q_{m,f}(c_{-m,f})$ below:

$$
Q_{m,f}(c_{-m,f}) = \left\{c_{m,f} = \min\{B_{m',m,f}+T_{m',m}r_{m',m}, r_f\} - B_{m',m,f} \times |\min\{B_{m',m,f}+T_{m',m}r_{m',m}, r_f\} - B_{m',m,f} > 0, \forall m' \in \mathcal{N}_m\right\}. \qquad (18)
$$

We then rank the elements in $Q_{m,f}$ in a non-decreasing order

$$
Q_{m,f}(c_{-m,f}) = \left\{c^{rank_1}_{m,f}, c^{rank_2}_{m,f}, \dots, c^{rank_{R_{m,f}}}_{m,f}\right\}, \qquad (19)
$$

where $B_{m,f}$ represents the number of elements in set $Q_{m,f}(c_{-m,f})$ and $0 \le c^{rank_1}_{m,f} \le c^{rank_2}_{m,f} \le \dots \le c^{rank_{R_{m,f}}}_{m,f} \le r_f$.

Then, $u_m^f(c_{m,f}, c_{-m,f})$ can be divided into $(R_{m,f}+1)$ parts according to $Q_{m,f}(c_{-m,f})$. So that $u_m^f(c_{m,f}, c_{-m,f})$ can be reformulated by introducing the left deviate $\left(u_m^f\right)'_+$ in Eq.(20), as shown at the top of this page. From the equation, combined with Lemma 1, we can easily find that $u_m^f(c_{m,f}, c_{-m,f})$ is a continuous and non-decreasing concave piecewise linear function on $c_{m,f}$.

Based on the above theoretical analysis, a Local Optimal Caching Algorithm with Social Selfishness (LOCASS) is proposed to solve the device caching game. The main idea of our algorithm for the caching model is to choose devices one by one and maximize the social selfishness-based utility for each of them accordingly until the system reaches the Nash Equilibrium. In detail, for optimizing the social-selfishness based utility, we firstly choose and store the part with the largest right derivative. Then, the system updates the content's right derivative to the next part and repeat the previous step until the cache is full. The detail of the Local Optimal Caching Algorithm with Social Selfishness(LOCASS) is listed in the Algorithm 1.

Algorithm 1 LOCASS: Local Optimal Caching Algorithm with Social Selfishness

1:**Repeat**:
2: Randomly choose one device m
3: **For** $\forall f \in \mathcal{F}$
4: $c_{m,f} = 0$
5: compute $Q_{m,f}(c_{-m,f})$
6: $count_f = 1$
7: $A_{m,f} = \left(u_m^f\right)'_+(0, c_{-m,f})$
8: **End for**
9: $f = \arg\max_{f' \in \mathcal{F}} A_{m,f'}$
10: **If** $\sum_{f \in \mathcal{F} \setminus f} c_{m,f} + c_{m,f}^{rank_{count_f}} < s_m$
11: $c_{m,f} = c_{m,f}^{rank_{count_f}}$
12: $count_f = count_f + 1$
13: $A_{m,f} = \left(u_m^f\right)'_+(rank_{count_f}, c_{-m,f})$
14: repeat step 9
15: **Else**
16: $c_{m,f} = s_m - \sum_{f' \in \mathcal{F} \setminus f} c_{m,f'}$
17: **End If**
18:**Until** all devices won't change their own caching strategy
19:Obtain the Nash equilibrium of caching strategy.

IV. SIMULATION RESULTS

In this section, we investigated the performance of the our proposed algorithm LOCASS. In the LOCASS, we set social density $\eta = 0, 0.3, 0.6, 0.9, 1$ respectively. Specifically, LOCASS with social density $\eta = 1$ can be considered as the full cooperation case where each device altruistically devote its cache space to maximize the system performance, specifically the average offloading ratio. LOCASS with social density $\eta = 0$ can be considered as the full selfishness case where each device just stores contents only for itself and has no right to access others' cache space. We compared the performance of LOCASS with other three caching algorithms: the *Most Popular Content* (MPC) caching, the *Random* caching and *Greedy* caching. In the MPC caching, each device just stores the most popular contents according to its individual request probability. In the Random caching, each device randomly chooses contents to store until its own cache space is full. The greedy caching starts with an empty set; at each step, it adds one content with the highest marginal average offloading ratio to the set until cache is full. In this part, we only shown the performance of those three caching algorithms when social density $\eta = 0.3$.

We adopt two types of social graph model for depicting the social relationship among devices:

* Erdos-Renyi(**ER**) graph model [30]: Each pair of devices has a fixed probability of being cooperative-devices or not. All devices are likely to have similar degrees.
* Scale-free(**SC**) graph model [31]: The degree of devices follows power law distribution where only a few of devices are equipped with high degrees and the rest of them are equipped with low degrees.

Then, we generated the number of contacts within unit time λ for each pair of devices according to a Gamma distribution $\Gamma(4.43, 1/1088)$. Additionally, we assumed that the content request probability follows the Zipf distribution with parameter α, i.e., $p_{m,f} = \frac{(f)^{-\alpha}}{\sum_{f' \in \mathcal{F}} (f)^{-\alpha}}, \forall m, f$. Other baseline simulation parameters were shown in Table 1.

TABLE 1. Baseline simulation parameters.

the total number of devices M	200
the total number of contents F	200
the cache size of each device s_m	50M
the parameter of Zipf distribution α	0.6
the parameter of ER social graph model η	0.3
the transmitted amount of data during one contact	20M
the deadline time T_D	200s
the size of each content r_f	100M
the social tie strength w	1

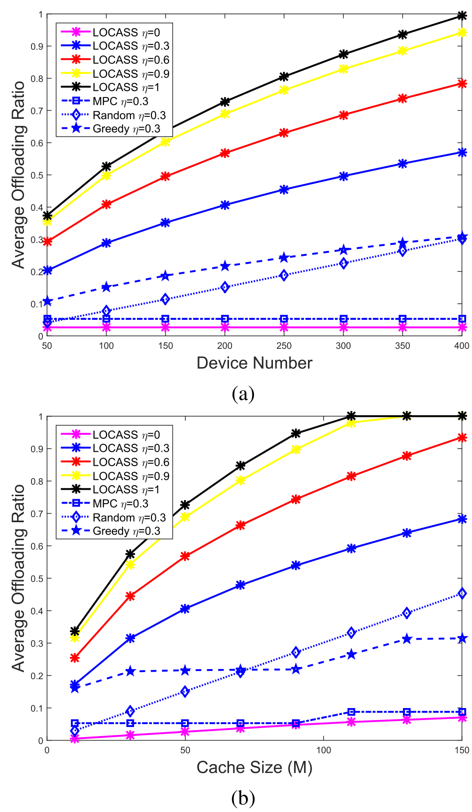


FIGURE 2. Average offloading ratio as the virtual cache size space changes in ER model. (a) Average offloading ratio v.s Device number. (b) Average offloading ratio v.s Cache size.

A. THE EVALUATION RESULTS IN THE ER SOCIAL GRAPH MODEL

1) THE IMPACT OF VIRTUAL CACHE SPACE

Virtual cache space is a set of individual cache space which is associated with the device number and the individual cache size. Fig. 2 shows the performance of the caching algorithms by increasing the individual cache size and the device number respectively. As the virtual cache space increases, the average offloading ratio of all the three algorithms increases except for the MPC with social density $\eta = 0.3$ and the LOCASS with social density $\eta = 0$. The increasing of average offloading ratio is because more content copies can be stored and shared via D2D links as the virtual cache space increases. In the MPC with social density $\eta = 0.3$, each device stores the same content so that no performance improvement can be achieved with the increasing device number. In the LOCASS with social density $\eta = 0$, each device is fully selfish and cannot access other devices' cache space, so the average offloading ratio doesn't change with the increasing device number. In addition, in terms of average offloading ratio, the LOCASS always outperform the MPC caching, the Random caching and the Greedy caching no matter how the device number or cache size changes. The greed caching outperforms the MPC and Random caching when the virtual cache space is small. And the Random caching outperforms the MPC caching when the virtual cache space is large.

With the increasing social density η, each device can access more cache space, and its requests can be taken into consideration by more cooperative-devices. So the average offloading ratio of LOCASS improves with larger social density η. From the Fig. 2, with the increase of social density η, LOCASS utilizes the virtual cache space in a less effective way in that the same size of accessible virtual cache space can achieve a lower average offloading ratio. For example, when the cache size is 50M and the device number is 200, each device of LOCASS with social density $\eta = 0.3$ can access 30% of virtual cache space and the average offloading ratio is 40.64%. Each device of LOCASS with $\eta = 1$ can access 100% of the virtual space, and the average offloading ratio is 72.73%. In Fig. 2, we can also find that the gap between different cases of LOCASS becomes larger with the increase of virtual cache space before the average offloading ratio reaches to 1.

2) THE IMPACT OF ZIPF PARAMETER

In the Zipf distribution $p_{m,f} = \frac{(f)^{-\alpha}}{\sum_{f' \in \mathcal{F}} (f)^{-\alpha}}, \forall m, f$, a larger α means a steeper request probability distribution where the majority of requests probably concentrate on the limited number of popular contents. Fig.3 depicts the simulation results of the average offloading ratio of the three algorithms with the change of the Zipf parameter α. With the increase of the Zipf parameter α, the average offloading ratio of three algorithms increases except for Random caching. In addition, the LOCASS outperforms the MPC caching, Random caching and Greedy caching no matter how α changes. Greedy caching outperforms the Random caching when α is large. And the Random caching outperforms the MPC caching when α is small. Considering the rising speed with the increase of the Zipf parameter α, the average

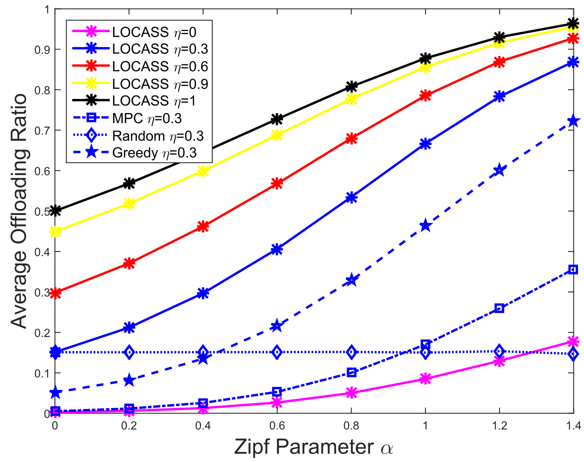


FIGURE 3. Average offloading ratio as the Zipf distribution parameter α changes.

offloading ratio of LOCASS rises slower with the increase of social density η. Because, devices can access limited virtual cache space, only a few number of popular contents can be stored (see Fig. 5b). With the increase of social density η, the number of the popular contents increases. The sum of the probability of those popular contents being requested increases slowly, thus smaller performance improvement can be achieved when α increases. Therefore, LOCASS is more sensitive to the change of the content request probability distribution with the decrease of social density η.

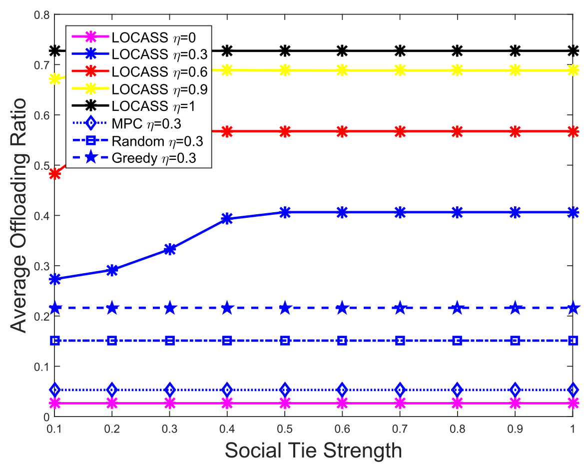


FIGURE 4. Average offloading ratio as the social tie strength w changes in ER model.

3) THE IMPACT OF SOCIAL TIE STRENGTH

Fig. 4 depicts the effect of social tie strength on the average offloading ratio. With the increase of social tie strength, the average offloading ratio of LOCASS improves except when social density $\eta = 0, 1$. The increasing of average offloading ratio is because the cooperation between devices is enhanced, and devices are more likely to store contents for cooperative-devices. As the social tie strength w increases, those contents with comparatively lower popularity have large chance to be cached and increase devices' social selfishness-based utilities. In the LOCASS with social density $\eta = 0$, each device cannot access cached contents from other devices so that social tie strength w has no impact on the average offloading ratio. Besides, in terms of average offloading ratio, LOCASS outperforms MPC caching, Random caching and Greedy caching no matter how social tie strength w changes when social density $\eta = 0.3$. We can also find that the average offloading ratio of LOCASS converges faster with the increase of social density η. In the Fig. 4, the average offloading ratio of LOCASS with social density $\eta = 0.3, 0.6, 0.9, 1$ converges when $w > 0.5$, 0.2, 0.2 and 0.1 respectively. When the average offloading ratio converges, it also means that the cached content distribution will not change. Therefore, when the social tie strength w is large enough, a large scale of rearrangement of content won't happen if the social tie strength changes a little. Larger η means more cooperative-devices and larger cooperative utility for each device. Smaller social tie strength w can pose the same effect on the average offloading with the increase of social density η. Therefore, the increase of social tie strength w can enhance the cooperation of devices and improve the average offloading ratio in LOCASS. With more cooperative-devices, i.e., larger social density η, social tie strength w poses less impact on the average offloading ratio.

4) THE IMPACT OF SOCIAL DENSITY η

Larger social density η means more cooperative-devices for each device. In another word, more virtual cache space can be accessed and more devices will take the requests of device m into consideration. We investigated the evolution result with the increasing of parameter social density η in Fig.5. In Fig. 5a, the average offloading ratio of LOCASS outperforms the MPC, Random caching and Greedy caching no matter how social density η changes. In addition, the average offloading ratio of four caching algorithms improves with the increase of social density η except MPC caching. For the LOCASS, the increasing speed reduces due to the fact that more cache space is utilized to store those comparatively unpopular contents, which is shown in Fig. 5b. Small performance improvement can be achieved if those comparatively unpopular contents are stored. Thus, the increasing speed of the average offloading ratio decreases as social density η grows.

Fig. 5b depicts the cumulative distribution function (CDF) of the virtual cache space volume for each content with different ranks in LOCASS. The content rank is sorted by content request probability in descending order. The content with a smaller rank is equipped with a larger request probability. If some new contents are requested, whether a large scale of rearrangement of content will happen depends on the rank of those new contents. If new contents are with small ranks, a large scale of virtual cache space will be utilized to store, and a large scale of rearrangement will happen. Respectively, if the ranks of new contents are large than 100, those contents have no chance to be stored so that the rearrangement of content won't happen. As shown in the Fig. 5b, the whole cache space is mainly occupied with the small-rank contents. Besides, with the increasing social density η, each individual cache space can be accessed by more devices, so fewer of copies can achieve similar performance. Devices with a larger η are likely to store the popular contents as well but tend to store those comparatively unpopular contents. This can further improve the system performance. Therefore, a larger social density η means more cache space of sharing via D2D links, and devices are encouraged to store diverse contents to avoid duplicate caching.

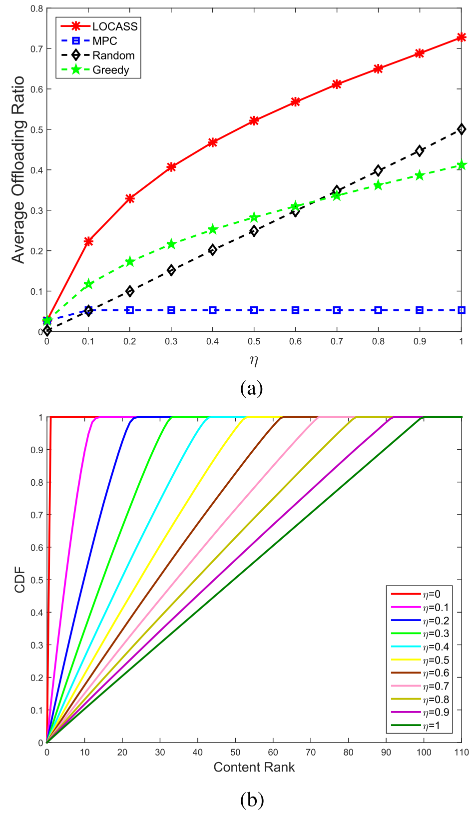


FIGURE 5. The performance as the social density η changes in ER model. (a) Average offloading ratio as the social density η changes in ER model. (b) Cached content distribution as the social density η changes for the LOCASS in ER model.

B. THE EVALUATION RESULTS IN THE REAL TRACE

To further validate the performance of LOCASS, we utilized the real trace of the Infocom06 dataset [32] to conduct a trace-driven simulation. The Infocom06 dataset contains the contact logs among 79 candidates in a period of 337417 seconds [32]. In the LOCASS, we applied two social graph model to construct social ties among these candidates: ER model and SC model. We assumed that the sum of social ties for the two models were equal. We set social density $\eta = 0.3$. For each tie, the social tie strength w was generated randomly in the interval [0,1]. Here, we just show

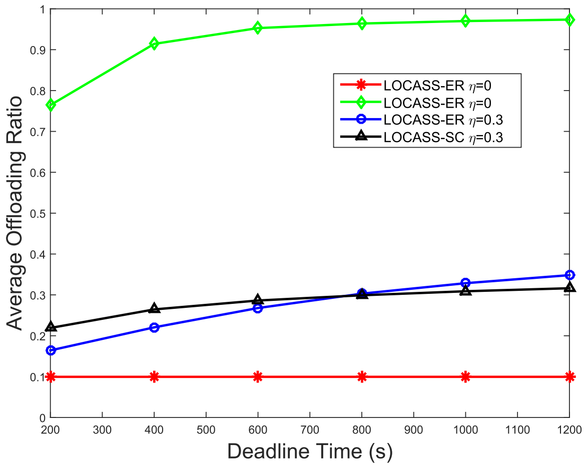


FIGURE 6. Average offloading ratio as the deadline time T_D changes in Infocom06 dataset.

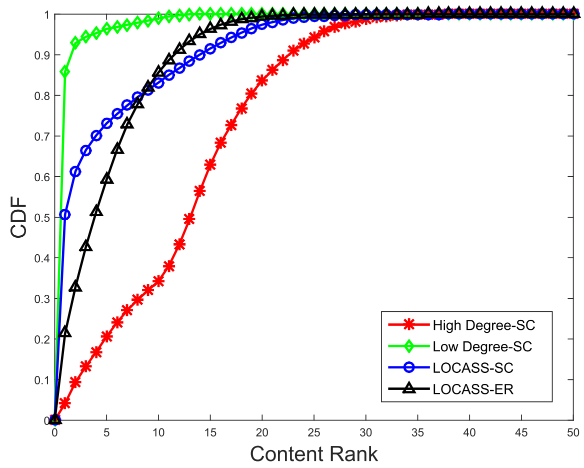


FIGURE 7. The cached content distribution in Infocom06 dataset.

the cached content distribution of devices with different degrees.

Fig. 6 shows the evolution result for the four cases (i.e., LOCASS $\eta = 0, 0.3, 1$ with ER model and LOCASS $\eta = 0.3$ with SC model) with the change of deadline time T_D. Except for the LOCASS $\eta = 0$ with ER model, the average offloading ratios of the other three cases grow along with the increase of deadline time T_D. That is because a larger T_D means more contacts between each pair of devices, and more cached segments can be transmitted via D2D links. In the case of LOCASS $\eta = 0$ with ER model, devices cannot access any other's cache space, so the average offloading ratio doesn't change with the increase of T_D. If there is same social density η, when T_D is small, the average offloading ratio of SC outperforms that of ER. Since, in the ER model, all devices' degrees are low, and the cooperative utilities of devices with small T_D are comparatively small. With a small cooperative utility, there is a lack of cooperation with their cooperative-devices and poor performance can be posed. However, in the SC, there are a few of devices equipped with high degrees, and their cooperative utility is comparatively large. With the large cooperative utility, those devices with high degrees have a strong motivation to cooperate with their cooperative-devices. Therefore, the average offloading ratio of SC outperforms that of ER when T_D is small. In the Fig. 6, when T_D exceeds 800s, for each pair of cooperative-devices, almost all of the cached contents can be transmitted before reaching the deadline time. In such situation, a large part of the virtual cache space in the LOCASS with SC is utilized to store unpopular contents for cooperative-devices with high degrees, which achieve only a limited performance improvement. However, in the ER, all devices have similar degrees and tend to store popular contents, which will achieve a great performance improvement. Therefore, in terms of average offloading ratio, the LOCASS with ER outperforms the LOCAS with SC when the deadline time T_D is large.

Fig. 7 shows how cache spaces are occupied by different contents, represented by a CDF graph. From the system perspective, the LOCASS-SC utilizes more cache space to store contents with a small rank than the LOCASS-ER. This is because of the devices in the SC equipped with low degrees, in that only a small part of virtual space can be accessed by each of them. Devices with a low degree can access few segments from their cooperative-devices and have to store popular contents for themselves. Therefore, compared to the LOCASS-ER, the LOCASS-SC utilizes a larger part of cache space to store popular contents. Considering the diversity of degrees in the SC, we also show the cache space occupied by different contents of our solution in the situation of devices with high degrees and low degrees which is called as *high degree-SC* and *low degree-SC* respectively. The former seletced five devices with highest degrees, and the latter evaluated five devices with lowest degrees in SC. As shown in the Fig. 7, devices with low degrees in SC tend to store popular contents, a.k.a., the contents with a small rank. In addition, devices with high degrees in LOCASS with SC tend to store comparatively unpopular contents and access popular contents from their cooperative-devices.

V. CONCLUSION

In this paper, with the social selfishness which can be viewed as a mixture of cooperation and selfishness, we formulate a device caching game. We also design a Local Optimal Caching Algorithm with Social Selfishness (LOCASS) to prove the existence of Nash equilibrium and solve the game. Simulation results are provided to demonstrate the validity of LOCASS. The results show that LOCASS can offer much better performance, in terms of average offloading ratio, than traditional Random, Most-Popular-Content (MPC) and Greedy caching algorithms. Besides, as the social tie strength and social density increases, the cooperation among cooperative-devices is enhanced, and the average offloading ratio of LOCASS also improves. Additionally, in LOCASS,

$$\begin{aligned} &\min\left\{T_{m',m"}r_{m',m"}, c'_{m",f}\right\} + \min\left\{T_{m',m"}r_{m',m"}, c''_{m",f}\right\} \\ &= \begin{cases} 2T_{m',m"}r_{m',m"}, & if\ c'_{m",f} \geq c''_{m",f} \geq T_{m',m"}r_{m',m"} \\ T_{m',m"}r_{m',m"} + c''_{m",f}, & if\ c'_{m",f} \geq T_{m',m"}r_{m',m"} \geq c''_{m",f}\ and\ \frac{c'_{m",f}+c''_{m",f}}{2} \geq T_{m',m"}r_{m',m"} \\ T_{m',m"}r_{m',m"} + c''_{m",f}, & if\ c'_{m",f} \geq T_{m',m"}r_{m',m"} \geq c''_{m",f}\ and\ \frac{c'_{m",f}+c''_{m",f}}{2} < T_{m',m"}r_{m',m"} \\ c'_{m",f} + c''_{m",f}, & if\ T_{m',m"}r_{m',m"} \geq c'_{m",f} \geq c''_{m",f} \end{cases} \end{aligned} \tag{21}$$

$$\min\left\{T_{m',m"}r_{m',m"}, \frac{c'_{m",f}+c''_{m",f}}{2}\right\} = \begin{cases} T_{m',m"}r_{m',m"}, & if\ c'_{m",f} \geq c''_{m'',f} \geq T_{m',m"}r_{m',m"} \\ T_{m',m"}r_{m',m"}, & if\ c'_{m",f} \geq T_{m',m"}r_{m',m"} \geq c''_{m",f}\ and\ \frac{c'_{m",f}+c''_{m",f}}{2} \geq T_{m',m"}r_{m',m"} \\ \frac{c'_{m",f}+c''_{m",f}}{2}, & if\ c'_{m",f} \geq T_{m',m"}r_{m',m"} \geq c''_{m",f}\ and\ \frac{c'_{m",f}+c''_{m",f}}{2} < T_{m',m"}r_{m',m"} \\ \frac{c'_{m",f}+c''_{m",f}}{2}, & if\ T_{m',m"}r_{m',m"} \geq c'_{m'',f} \geq c''_{m",f} \end{cases} \tag{22}$$

devices with low degrees are likely to store popular contents, and devices with high degrees tend to store those comparatively unpopular contents and fetch popular content from their cooperative-devices. Nowadays, devices' mobility and request forecast are becoming more previously by adopting new technologies, such as big data. Each device transmits and receives the information on the mobility and request forecast of its cooperative-devices and makes its own caching strategy decision according to our proposed caching algorithm LOCASS. During off-peak time, contents are proactively distributed to the devices cache space and satisfy the requests via D2D links at peak time. In our future work, we will focus on the dynamic social relationship in the system, where social tie strength can be strengthened if they share cached contents to each other.

APPENDIX

Proof: Firstly, $\min\left\{T_{m',m"}r_{m',m"}, c_{m",f}\right\}$ is concave [27] on **c**. Because, for $\forall c'_{m",f}$ and $c''_{m",f}$, where $c'_{m",f} \geq c''_{m",f}$, there exists (21), as shown at the top of this page. And, (22), as shown at the top of this page.

Then, the following condition is satisfied:

$$\begin{aligned} &\min\left\{T_{m',m"}r_{m',m"}, c'_{m",f}\right\} + \min\left\{T_{m',m''}r_{m',m"}, c''_{m",f}\right\} \\ &\quad \leq 2\min\left\{T_{m',m"}r_{m',m"}, \frac{c'_{m",f}+c''_{m",f}}{2}\right\} \end{aligned} \tag{23}$$

And $\min\left\{T_{m',m"}r_{m',m"}, c_{m",f}\right\}$ is also concave on c.

Similar with the above proof,

$$\min\left\{\sum_{m"\in\mathcal{N}_{m'}\cup m'} \min\left\{T_{m',m"}r_{m',m"}, c_{m",f}\right\}, r_f\right\} \tag{24}$$

is a concave function on **c**.

Because non-negative weighted sum of concave function remains to be a concave function,

$$\begin{aligned} U_m(\mathbf{c}) = &\sum_{m'\in\mathcal{N}_m\cup m}\left\{w_{m,m'}\sum_{f\in F}p_{m',f}\right. \\ &\left.\times\frac{1}{r_f}\min\left\{\sum_{m"\in\mathcal{N}_{m'}\cup m'}\min\left\{T_{m',m"}r_{m',m"}, c_{m",f}\right\}, r_f\right\}\right\} \end{aligned} \tag{25}$$

is also a function on **c**. ■

ACKNOWLEDGMENT

For future 5G and IoT applications in complex environments, raw measurement data of multiple radio channels at different communication scenarios can be found at an open-source website: www.wise.sh.

REFERENCES

[1] D. Liu, B. Chen, C. Yang, and A. F. Molisch, "Caching at the wireless edge: Design aspects, challenges, and future directions," *IEEE Commun. Mag.*, vol. 54, no. 9, pp. 22–28, Sep. 2016.

[2] Cisco, "Cisco visual networking index: Global mobile data traffic forecast update, 2016–2021," presented at the San Jose, CA, USA, White Paper 1454457600805266, Feb. 2017. [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/mobile-white-paper-c11-520862.html

[3] E. Bastug, M. Bennis, and M. Debbah, "Living on the edge: The role of proactive caching in 5G wireless networks," *IEEE Commun. Mag.*, vol. 52, no. 8, pp. 82–89, Aug. 2014.

[4] Y. Shi, S. Chen, and X. Xu, "MAGA: A mobility-aware computation offloading decision for distributed mobile cloud computing," *IEEE Internet Things J.*, vol. 5, no. 1, pp. 164–174, Feb. 2018.

[5] Y. Yang, J. Xu, G. Shi, and C.-X. Wang, *5G Wireless Systems: Simulation and Evaluation Techniques*. New York, NY, USA: Springer, 2017.

[6] Z. Zhu, S. Jin, Y. Yang, H. Hu, and X. Luo, "Time reusing in D2D-enabled cooperative networks," *IEEE Trans. Wireless Commun.*, vol. 17, no. 5, pp. 3185–3200, Mar. 2018.

[7] Y. Yang, K. Wang, G. Zhang, X. Chen, X. Luo, and M. Zhou, "MEETS: Maximal energy efficient task scheduling in homogeneous fog networks," *IEEE Internet Things J.*, to be published.

[8] N. Golrezaei, A. F. Molisch, A. G. Dimakis, and G. Caire, "Femtocaching and device-to-device collaboration: A new architecture for wireless video distribution," *IEEE Commun. Mag.*, vol. 51, no. 4, pp. 142–149, Apr. 2013.

[9] B. Chen, C. Yang, and Z. Xiong, "Optimal caching and scheduling for cache-enabled D2D communications," *IEEE Commun. Lett.*, vol. 21, no. 5, pp. 1155–1158, May 2017.
[10] Y. Zhu, B. Xu, X. Shi, and Y. Wang, "A survey of social-based routing in delay tolerant networks: Positive and negative social effects," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 1, pp. 387–401, 1st Quart., 2013.
[11] Y. Wu *et al.*, "Challenges of mobile social device caching," *IEEE Access*, vol. 4, pp. 8938–8947, 2016.
[12] S. Hosny, A. Eryilmaz, and H. El Gamal, "Impact of user mobility on D2D caching networks," in *Proc. IEEE Global Commun. Conf.*, Washington, DC, USA, Dec. 2016, pp. 1–6.
[13] Y. Li, P. Hui, D. Jin, L. Su, and L. Zeng, "Evaluating the impact of social selfishness on the epidemic routing in delay tolerant networks," *IEEE Commun. Lett.*, vol. 14, no. 11, pp. 1026–1028, Nov. 2010.
[14] D. Malak, M. Al-Shalash, and J. G. Andrews, "Optimizing the spatial content caching distribution for device-to-device communications," in *Proc. IEEE Int. Symp. Inf. Theory*, Barcelona, Spain, Jul. 2016, pp. 280–284.
[15] J. Rao, H. Feng, and Z. Chen, "Exploiting user mobility for D2D assisted wireless caching networks," in *Proc. 8th Int. Conf. Wireless Commun., Signal Process.*, Yangzhou, China, 2016, pp. 1–5.
[16] M. Chen, Y. Hao, L. Hu, K. Huang, and V. Lau, "Green and mobility-aware caching in 5G networks," *IEEE Trans. Wireless Commun.*, vol. 16, no. 12, pp. 8347–8361, Dec. 2017.
[17] R. Wang, J. Zhang, S. H. Song, and K. B. Letaief, "Mobility increases the data offloading ratio in D2D caching networks," in *Proc. IEEE Int. Conf. Commun.*, Paris, France, May 2017, pp. 1–6.
[18] T. Liu, J. Li, F. Shu, and S. Chu, "D2D-enabled wireless caching using Stackelberg game," in *Proc. 8th Int. Conf. Wireless Commun., Signal Process.*, Yangzhou, China, 2016, pp. 1–5.
[19] Z. Chen, Y. Liu, B. Zhou, and M. Tao, "Caching incentive design in wireless D2D networks: A Stackelberg game approach," in *Proc. IEEE Int. Conf. Commun.*, Kuala Lumpur, Malaysia, May 2016, pp. 1–6.
[20] K. Zhu, W. Zhi, L. Zhang, X. Chen, and X. Fu, "Social-aware incentivized caching for D2D communications," *IEEE Access*, vol. 4, pp. 7585–7593, 2016.
[21] Y. Wang, J. Wu, and M. Xiao, "Hierarchical cooperative caching in mobile opportunistic social networks," in *Proc. IEEE Global Commun. Conf.*, Austin, TX, USA, Dec. 2014, pp. 411–416.
[22] K. Zhu, W. Zhi, X. Chen, and L. Zhang, "Socially motivated data caching in ultra-dense small cell networks," *IEEE Netw.*, vol. 31, no. 4, pp. 42–48, Jul./Aug. 2017.
[23] J. Li *et al.*, "On social-aware content caching for D2D-enabled cellular networks with matching theory," *IEEE Internet Things J.*, to be published.
[24] K. Jung and S. Park, "Collaborative caching techniques for privacy-preserving location-based services in peer-to-peer environments," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Boston, MA, USA, Dec. 2017, pp. 4497–4506.
[25] K. Fawaz, N. Abbani, and H. Artail, "A privacy-preserving cache management system for MANETs," in *Proc. 19th Int. Conf. Telecommun. (ICT)*, Jounieh, Lebanon, Apr. 2012, pp. 1–6.
[26] X. Chen, X. Gong, L. Yang, and J. Zhang, "Exploiting social tie structure for cooperative wireless networking: A social group utility maximization framework," *IEEE/ACM Trans. Netw.*, vol. 24, no. 6, pp. 3593–3606, Dec. 2016.
[27] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.
[28] J. B. Rosen, "Existence and uniqueness of equilibrium points for concave n-person games," *Econometrica, J. Econ. Soc.*, vol. 33, no. 3, pp. 520–534, 1965.
[29] K. Goebel and W. A. Kirk, *Topics in Metric Fixed Point Theory*. Cambridge, U.K.: Cambridge Univ. Press, 1990, vol. 28.
[30] M. E. J. Newman, D. J. Watts, and S. H. Strogatz, "Random graph models of social networks," *Proc. Nat. Acad. Sci. USA*, vol. 99, no. 1, pp. 2566–2572, 2002.
[31] X. F. Wang and G. Chen, "Complex networks: Small-world, scale-free and beyond," *IEEE Circuits Syst. Mag.*, vol. 3, no. 1, pp. 6–20, Sep. 2003.
[32] J. Scott, R. Gass, J. Crowcroft, P. Hui, C. Diot, and A. Chaintreau. (May 2015). *CRAWDAD Trace Cambridge/Haggle/Imote/Infocom2006 (v.2009-05-29)*. [Online]. Available: http://crawdad.cs.dartmouth.edu/cambridge/haggle/imote/infocom2006

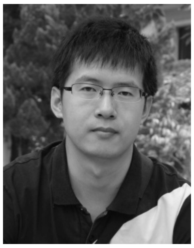
YANG YANG (S'99–M'02–SM'10–F'18) received the B.Eng. and M.Eng. degrees in radio engineering from Southeast University, Nanjing, China, in 1996 and 1999, respectively, and the Ph.D. degree in information engineering from The Chinese University of Hong Kong in 2002.

He has held faculty positions at The Chinese University of Hong Kong, Brunel University, and University College London, U.K. He is currently a Professor with the Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences, also serving as the Director of the CAS Key Laboratory of Wireless Sensor Network and Communication and the Shanghai Research Center for Wireless Communications, and also a Co-Director of the Shanghai Institute of Fog Computing Technology. He is also a Distinguished Adjunct Professor with the School of Information Science and Technology, ShanghaiTech University.

Dr. Yang is a member of the Chief Technical Committee of the National Science and Technology Major Project New Generation Mobile Wireless Broadband Communication Networks from 2008 to 2020, which is funded by the Ministry of Industry and Information Technology of China. In addition, he is on the Chief Technical Committee for the National 863 Hi-Tech R&D Program 5G System R&D Major Projects, which is funded by the Ministry of Science and Technology of China. Since 2017, he has been serving the OpenFog Consortium as a Board Member and the Director of Greater China Region.

YECHENG WU received the B.S. degree from the Department of Optical Information Science and Technology, Nanjing University of Posts and Telecommunications, Nanjing, China, in 2013. He is currently pursuing the Ph.D. degree with the Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences, Shanghai, China. His research interests include wireless caching, social network analysis, mobile edge computing, and resource allocation in fifth-generation networks.

NANXI CHEN received the B.Eng. degree in software engineering from Wuhan University, China, the M.Eng. degree in electronic engineering from Dublin City University, Ireland, and the Ph.D. degree from the Department of Computer Science, Trinity College Dublin, University of Dublin, Ireland. She is currently an Assistant Professor with the Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences. Her research interests include fog computing, mobile and pervasive computing, composition of (mobile) services, and semantic service discovery.



KUNLUN WANG received the Ph.D. degree in electronic engineering from Shanghai Jiao Tong University, Shanghai, China, in 2016. From 2016 to 2017, he was with Huawei Technologies Co., Ltd., where he was involved in energy efficiency algorithm design. Since 2017, he has been an Assistant Professor with the Key Laboratory of Wireless Sensor Network and Communication, Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences, Shanghai, China. He is also with the Shanghai Institute of Fog Computing Technology. His research interests include energy efficient communications, resource allocation, cross-layer design, and fog computing. In 2015, he received the Intel Fellowship from Intel China.

SHANZHI CHEN (SM'04) received the bachelor's degree from Xidian University in 1991 and the Ph.D. degree from the Beijing University of Posts and Telecommunications, China, in 1997. He joined the Datang Telecom Technology and Industry Group and China Academy of Telecommunication Technology (CATT) in 1994, and has served as EVP Research and Development since 2008. He is currently the Director of the State Key Laboratory of Wireless Mobile Communications, CATT, where he conducted research and standardization on 4G TD-LTE and 5G. He has authored and co-authored four books [among them the textbook *Mobility Management: Principle, Technology and Applications*, (Springer Press)], 17 book chapters, approximately 100 journal papers, 50 conference papers, and over 50 patents in these areas. He has contributed to the design, standardization, and development of 4G TD-LTE and 5G mobile communication systems. His current research interests include 5G mobile communications, network architectures, vehicular communication networks, and Internet of Things. He served as a member and a TPC Chair and of many international conferences. His achievements have received multiple top awards by China central government and honors, especially, the Grand Prize of National Award for Scientific and Technological Progress, China, in 2016 (This Grand Prize is the highest category and in some years, it leaves with no winners due to its high standard). He is the Editor of the IEEE *Network* and the IEEE INTERNET OF THINGS, and the Guest Editor for the IEEE WIRELESS COMMUNICATIONS, the IEEE *Communications Magazine*, and the IEEE TVT.

SHA YAO received the B.Sc. degree from Zhejiang University, Hangzhou, China, in 2004, and the M.Sc. and Ph.D. degrees from the Royal Institute of Technology, Stockholm, Sweden, in 2006 and 2011, respectively, all in electrical engineering. He is currently with Enigma Investment, Shanghai, China. His research interests include information-theoretic perspectives on cooperative transmission and relay networks, resource allocation in fifth-generation heterogeneous networks, and task scheduling in crowdsensing.