Cooperative Game-based Cheating in Full-duplex Relaying-based D2D Communication Underlaying Heterogeneous Cellular Networks

Bowen Wang¹ · Yanjing Sun¹ · Qi Cao¹ · Song Li¹ · Yanfen Wang¹

Published online: 16 November 2017
© The Author(s) 2017. This article is an open access publication

Abstract Device-to-device (D2D) communications allow direct transmissions between two adjacent user devices, which can improve system performance in cellular networks. Considering full-duplex (FD) relaying outperforms half-duplex (HD) relaying in spectrum and energy efficiency, we apply the FD relaying-based D2D communication scheme in heterogeneous cellular network, which allows D2D links to underlay cellular downlink by assigning D2D transmitters as FD relays to assist cellular downlink transmissions. Although the scheme can improve the utilization of spectrum resource dramatically, the unreasonable resource sharing of D2D users will increase the inter-user interference. Therefore, in this paper, we try to optimize the throughput of D2D users on the premise of ensuring the quality of service (QoS) of cellular users. To this end, we first propose a D2D transmit power-allocation scheme and use Gale-Sharpley (GS) algorithm in matching theory to solve the resource allocation problem. Next, a Cheating

algorithm based on cooperative game theory is proposed to further improve the throughput of D2D users. More importantly, due to the NP-hardness of finding the optimal solution of Cheating algorithm, we propose a heuristic algorithm based on depth first search (DFS) which using different colors to represent the different statements of D2D users to find a near-optimal solution. The simulation results show that the proposed algorithm compared with GS algorithm can significantly improve the total throughput of D2D users, and it also has lower complexity and better throughput performance against existing Cheating algorithm.

Keywords Heterogeneous network · Full-duplex relaying · Device-to-device · Matching theory · Game theory · Resource management

1 Introduction

With the rapid development of mobile communication, the demand for multimedia services over heterogeneous networks has soared at a tremendous pace over recent years [10, 28–31]. However, the contradiction between the increasing service demands of users and the congested spectrum resource has become increasingly prominent [9]. Compared to 4G wireless communications, 5G wireless communication is expected to bring nearly a thousand times upgrade in data traffic [16]. Therefore, how to improve the system throughput has become the research hotspot of 5G wireless communication. To solve the problem, we need innovation in both technologies and mathematical tools.

Researchers in academia and industry therefore attempt to explore new valuable technologies that can improve the

✉ Yanjing Sun
yjsun@cumt.edu.cn

Bowen Wang
wbwzt@cumt.edu.cn

Qi Cao
qciao@cumt.edu.cn

Song Li
lisong@cumt.edu.cn

Yanfen Wang
lszwyf@163.com

¹ School of Information and Control Engineering, China University of Mining and Technology, Xuzhou, China

system throughput by spectrum reuse. As one of the alternative technologies of 5G communication, Device-to-device (D2D) technology has many advantages. Firstly, different from the traditional cellular networks (CNs), where all user equipments (UEs) transmit data signals through the base station (BS) directly, UEs may communicate directly via the D2D links by cellular spectrum resources under the control of the BS in D2D communication [33]. Secondly, the distance between two D2D users is closer, which can reduce the transmission power of the terminal and prolong the life of the battery [11]. Finally, the appropriate increase in the number of D2D users, not only offload the base station, but also enhance the system throughput [36]. Meanwhile, the research on wireless relaying has also made a breakthrough, from the early theoretic results to a practical stage in cellular networks. It is found that full-duplex (FD) relaying outperforms half-duplex (HD) relaying in spectrum and energy efficiency [24], since FD relays can transmit and receive simultaneously on the same frequency band. Therefore the combination of FD relaying and D2D technology can greatly improve the throughput.

In fact, few works have been carried out so far to explore FD relaying based D2D communication schemes. In [23], the authors propose an HD relaying-based D2D communication protocol that allows the D2D users to communicate bidirectionally with each other while assisting the two-way communication between the BS and the CU over the same time and frequency resources. However, the scheme decreases the reuse gain of spectrum resource due to the increase of interference. In [35], combining the FD relaying and D2D technology, the author propose an FD relaying-based D2D communication scheme that allows D2D links to underlay cellular downlink by assigning D2D transmitters as FD relays to assist cellular downlink transmissions. However, only the single user scenario is considered, where one D2D pair and one cellular user (CU) exist, whereas, the problem of resource allocation in multi-user scenarios isn't addressed. Inspired by [35], we apply the FD relaying-based D2D communication scheme in a single cell multi-user scenario and consider the resource allocation problem in this scenario to further improve the throughput of D2D users.

Recently, most of the related research applied new mathematical tools such as game theory [22, 26], graph theory [32, 34] and matching theory [14, 15] to solve the resource allocation problems in D2D communications. In [22], considering a small-cell scenario in which full-duplex base station communicates with half-duplex D2D users, an iterative algorithm is proposed based on game theory by modeling the problem as a non-cooperative game between the uplink and downlink channels. In [26], the author apply the Stackelberg game framework in the resource allocation between D2D users and cellular users, which enhance the system throughput and user fairness. Previous works apply

classic algorithm of graph theory such as the Hungarian (also refer to as Kuhn-Munkres, KM) algorithm [32, 34] to pairing D2D users and CUs, which achieves the optimal solution of the system throughput. However, the Hungarian algorithm only considers the optimal welfare instead of the stability. This stability notion implies robustness to deviations that can benefit both the resource owners (e.g. CUs in this paper) and the users (e.g. D2D pairs in this paper) [15]. In fact, an unstable matching may cause the case where D2D users switch to reuse other cellular users' spectrum resource when the switch can benefit all of them. The case will cause the failure of the resource allocation scheme. Gu et al. [14, 15] implement matching theory to find a stable and efficient resource allocation method, which not only takes welfare into consideration but also guarantees stability in D2D communications. Moreover, the authors introduce the idea of cheating in matching to further improve D2D users' throughput. However, the throughput performance of the algorithm in [14] is far from optimal and the complexity of the algorithm in [15] is too high to formulate.

Comparing with the above works, the main contributions of this paper are presented as follows:

- We investigate resource allocation in multi-user heterogeneous cellular network with FD relaying-based D2D communication. In this model, we aim to maximize the throughput of D2D users on the premise of ensuring the QoS of CUs by allocating the power of the D2D transmitter. In addition, to further improve the throughput of D2D users, we solve this resource allocation problem by modeling it as a stable matching problem. The Gale-Shapley (GS) algorithm is utilized to find a stable matching between admissible D2D pairs and CUs.
- Besides, we propose a Cheating algorithm based on cooperative game theory in order to further improve some D2D users' throughput. It is proven that the cheating mechanism benefits a subset of D2D users without hurting the performance of the rest. Due to the NP-hardness of finding the cabal with biggest size, we cannot find the optimal solution within a polynomial time. Hence, using different colors to represent the different statements of D2D users, we develop a heuristic algorithm based on DFS to find a near-optimal solution of Cheating algorithm. Simulation results shows that our algorithm has better throughput and computation complexity performance compared with the existing algorithms.

The rest of this paper is organized as follows. Section 2 introduces the system model and formulate the resource allocation problem. The optimization problem is solved in Section 3 and the cheating problem is discussed in Section 4. Numerical results shown in Section 5 evaluate our proposed algorithms, while Section 6 concludes the paper.

Fig. 1 The heterogeneous cellular network model

2 System model and problem formulation

The heterogeneous cellular network model and FD relaying-based D2D underlying cellular network model are shown in Figs. 1 and 2, respectively. The heterogeneous cellular network consists of a base station (BS), D2D pairs and cellular users (CUs) in a single cell. Here, we use $\mathcal{D} = \{d_1, \dots, d_i, \dots, d_N\}$, $1 \leq i \leq N$ and $\mathcal{C} = \{c_1, \dots, c_j, \dots, c_L\}$, $1 \leq j \leq L$ to denote the index sets of D2D pairs and CUs, respectively. Each D2D pair d_i consists of a D2D transmitter UE- R_i and a D2D receiver UE- D_i . As for the D2D transmitter UE- R_i , we make the following assumptions based on [35] and [25].

- In this system model, each UE- R_i operates in the FD relaying mode. In other words, UE- R_i can transmit to UE- D_i and receive-and-forward for c_j over the same frequency band and same time, which indicates that the D2D transmission from UE- R_i to UE- D_i can underlay the cellular downlink transmission from BS to c_j .

Fig. 2 FD relaying-based D2D communication model

- In addition, each UE- R_i is equipped with isolated receive and transmit antennas. Therefore, the residual loop interference (LI, resulting from the relay transmission to the relay reception) at a D2D transmitter can be controlled at a certain level.

The cellular network operates in the frequency-division duplex mode. Since the available downlink bandwidth for each CU c_j is W_j hertz. We assume that d_i reuse the c_j 's spectrum resource and formulate the optimization problem step by step.

We consider both fast fading due to multi-path propagation effect and slow fading due to shadowing effect. Hence, the channel gain between UE- R_i and BS can be expressed as $h_{B,R_i} = K\delta_{B,R_i}\zeta_{B,R_i}d_{B,R_i}^{-\alpha}$, where K represents a system constant, δ_{B,R_i} represents the fast fading gain as the exponential distribution with unit mean, ζ_{B,R_i} represents the slow fading gain as the log-normal distribution [13], d_{B,R_i} represents the distance between BS and UE- R_i , and α represents the path loss exponent. Similarly, h_{R_i,C_j} denotes the channel gain from UE- R_i to CU $_j$, h_{R_i,D_i} denotes the channel gain from UE- R_i to UE- D_i . h_{B,C_j} denotes the interference channel gains from BS to CU $_j$. h_{B,D_i} denotes the interference channel gains from BS to UE- D_i . Due to imperfect cancellation, we also denote the residual LI channel at UE- R_i by h_{LI} . Assume that the BS transmit with power P_B for each UE- R_i . Let P_R denote the transmit power of each UE- D_i . The transmission powers of all D2D pairs and BS are equal. Let σ^2 denote the variance of the zero-mean additive white Gaussian noise on each channel.

The signal-to-interference-and-noise ratio (SINR) received by UE- R_i can be expressed as:

$$\gamma_{B,R_i} = \frac{P_B h_{B,R_i}}{P_R h_{LI} + \sigma^2}, \quad (1)$$

The signal transmitted by UE- R_i consists of the following two components. The first is the regenerated signal that will be forwarded to the cellular user, i.e., CU $_j$, and the second is the signal to be transmitted directly to the D2D receiver UE- D_i [35]. We assume that UE- R_i adopts the decode-and-forward (DF) protocol [24] to assist the cellular downlink transmission for CU $_j$, and UE- R_i is willing to use the λ ($0 \leq \lambda \leq 1$) fraction of the power P_R to forward the first signal component to CU $_j$. In addition, UE- R_i can use the remaining $1 - \lambda$ fraction power to transmit the second signal component to UE- D_i . The SINRs received by CU $_j$ and UE- D_i can be given by:

$$\gamma_{R_i,C_j} = \frac{\lambda_{i,j} P_R h_{R_i,C_j}}{P_B h_{B,C_j} + (1 - \lambda_{i,j}) P_R h_{R_i,C_j} + \sigma^2}, \quad (2)$$

$$\gamma_{R_i,D_i} = \frac{(1 - \lambda_{i,j}) P_R h_{R_i,D_i}}{P_B h_{B,D_i} + \lambda_{i,j} P_R h_{R_i,D_i} + \sigma^2}, \quad (3)$$

respectively. Based on the modulation and coding scheme in the DF protocol, the instantaneous end-to-end SINR of the link from BS to CU_j via $UE-R_i$ should be determined according to the SINR of weakest hop. So we can get:

$$\gamma_{B,C_j} = \min(\gamma_{B,R_i}, \gamma_{R_i,C_j}), \quad (4)$$

Finally, we can get the throughput for the D2D user d_i and cellular user c_j , respectively, as

$$\mathcal{R}_{i,j}^D = W_j \log_2(1 + \gamma_{R_i,D_i}), \quad (5)$$

$$\mathcal{R}_{i,j}^C = W_j \log_2(1 + \gamma_{B,C_j}), \quad (6)$$

Although the FD relaying-based D2D communication scheme can dramatically improve the spectrum utilization, the unreasonable sharing of spectrum resources may increase the interference between D2D user and CU. If the BS assigns CUs to D2D users randomly, the QoS requirement will be not satisfied due to the interference caused by too close distance. Therefore, an effective resource allocation algorithm will play an important role in FD relaying-based D2D communication.

Considering the fairness of communication, one D2D user can only reuse the spectrum resources of one CU. Assuming that the final channel assignment matrix is $X = [x_{ij}]_{N \times L}$, where x_{ij} is the resource indicator for CU and D2D pair. Here, $x_{ij} = 1$ when D2D pair d_i reuses CU c_j 's bandwidth W_j , and $x_{ij} = 0$ otherwise. W_j is an equal share of spectrum for each CU. The system objective is to optimize the throughput of D2D users while satisfying the QoS requirements of the cellular users. Then, we can formulate the optimization problem and constraints as follows:

$$\begin{aligned} \max_{\{\lambda_{i,j}, x_{ij}\}} \quad & \sum_{i=1}^N \sum_{j=1}^L x_{ij} \mathcal{R}_{i,j}^D \\ \text{s.t.} \quad & \mathcal{C1} : \mathcal{R}_{i,j}^C \geq \mathcal{R}_{\min}^C, \\ & \mathcal{C2} : \sum_{i=1}^N x_{ij} \leq 1; \sum_{j=1}^L x_{ij} \leq 1, \end{aligned} \quad (7)$$

Here, constraint $\mathcal{C1}$ guarantees the SINR requirement of each CU. $\mathcal{C2}$ presents that each CU can only share channel with one D2D user and each D2D user can only reuse one CU's channel, respectively. In order to optimize the throughput of D2D users while satisfying the QoS requirements of CUs, we try to solve the resource allocation problem in two steps, i.e., power allocation of D2D transmitter and one-to-one matching between admitted D2D users and CUs. It can be seen from the Eq. 7 that a D2D pair can share with a CU only when the QoS requirements of this CU is satisfied. Hence, we need to control the power of D2D transmitter to meet the QoS requirements of CUs before matching D2D

users and CUs. We will solve these two problems step by step in the next sections respectively.

3 Power allocation and stable matching

In previous section, we briefly introduce the FD relaying-based D2D communication scheme in a single cell multi-user scenario and formulate the optimization problem. In this section, we try to solve the above resource allocation problem in two steps. The first step is to satisfy the QoS requirement of CUs by power allocation. In the second step, the Gale-sharpley algorithm is adopted in order to find a stable matching. we take both CUs and D2D users' preferences into consideration. Since our system objective is to maximize throughput, we measure the D2D users' and the CUs' preferences by throughput, respectively.

3.1 Power allocation

In cellular communication, QoS is related to the user experience [18]. SINR is an important indicator to measure the performance of QoS. It can be seen from the Eqs. 3 and 4 that the power allocation factor is proportional to the SINR of CUs and inversely proportional to the SINR of D2D links. Therefore, it is necessary to adjust the power allocation factor to meet the SINR requirements of CUs before pairing on D2D users and CUs. According to CUs' minimum achievable rate requirement $\mathcal{R}_{\min}^C = W_j \log_2(1 + \gamma_{\min}^C)$, we get the minimum SINR requirement of CUs $\gamma_{\min}^C$. Hence, we should guarantee that the instantaneous end-to-end SINR of the link from BS to CU_j via $UE-R_i$ is γ_{R_i,C_j} , namely $\gamma_{R_i,C_j} \leq \gamma_{B,R_i}$. So that we can adjust the power allocation factor to meet the SINR requirements of CUs. Since the self-interference is small enough, the equation $\gamma_{B,R_i} \geq \gamma_{\min}^C$ is always guaranteed. So the γ_{R_i,C_j} needs to meet the Eq. 8:

$$\gamma_{\min}^C \leq \gamma_{R_i,C_j} \leq \gamma_{B,R_i}, \quad (8)$$

The satisfaction of Eq. 8 needs to adjust the power allocation factor. So the $\lambda_{i,j}$ needs to meet the Eq. 9:

$$\begin{aligned} \frac{\gamma_{\min}^C (P_B h_{B,C_j} + P_R h_{R_i,C_j} + \sigma^2)}{\gamma_{\min}^C P_R h_{R_i,C_j} + P_R h_{R_i,C_j}} \leq \lambda_{i,j} \leq \\ \min \left(1, \frac{P_B h_{B,R_i} (P_B h_{B,C_j} + P_R h_{R_i,C_j} + \sigma^2)}{P_R h_{R_i,C_j} (P_B h_{B,R_i} + P_R h_{L1} + \sigma^2)} \right) \end{aligned} \quad (9)$$

It can be seen from Eq. 9 that the power allocation factor $\lambda_{i,j}$ is in a certain range. Then we select a feasible solution from the range. In this way, both the QoS requirement and matching requirement of CUs are satisfied. In order to further enhance the throughput of D2D users, a stable matching scheme between D2D users of CUs is necessary.

3.2 Stable matching

In step 2, we give the establishment process of the preference list, which is the critical component of matching model. The preference list is mainly based on maximizing the D2D users' throughput. Here, we introduce a men-optimal (i.e., D2D users optimal) stable matching algorithm called Gale-Sharpley algorithm to further solve the resource allocation problem, which is beneficial to men (i.e., D2D users in this paper).

Since one D2D user can only reuse the spectrum resources of one CU and one CU can only be reused by one D2D user. This one-to-one matching problem can be formulated as the stable marriage problem with preference list in matching theory [21]. In a matching process, D2D pairs and CUs can be regarded as men and women, respectively. The following thing is to establish preference lists for CUs and D2D users. Since our system objective is to maximize throughput (i.e., achievable rate in this paper), we establish D2D users' and the CUs' preference lists by throughput. Let d_i 's throughput $\mathcal{R}_{i,j}^D$ denote d_i 's preference value over c_j , and likewise, c_j 's throughput $\mathcal{R}_{i,j}^C$ as c_j 's preference value over d_i . Hence, the preference list of d_i over c_j , denoted by $\mathcal{PL}_i^D$, is ranked by c_j 's preference value in a descending order, and similarly c_j 's preference list $\mathcal{PL}_j^C$ over d_i , is ranked by d_i 's preference value in a descending order. In order to further combine the resource allocation problem with stable marriage problem, we define the relation "prefer" in this paper as follows:

Definition 1 If $\mathcal{R}_{i,j}^C > \mathcal{R}_{i',j}^C$, it indicates that c_j prefers d_i to $d_{i'}$, denoted by $d_i \succ_{c_j} d_{i'}$. Similarly, if $\mathcal{R}_{i,j}^D > \mathcal{R}_{i',j}^D$, it indicates that d_i prefers c_j to $c_{j'}$, denoted by $c_j \succ_{d_i} c_{j'}$.

D2D users will switch to reuse other cellular users in an unstable matching, where exists more than one blocking pairs. The blocking pair can be defined as follows:

Definition 2 $(d_i, c_{j'})$ is a blocking pair in a one-to-one matching if $c_{j'} \succ_{d_i} c_j$ and $d_i \succ_{c_{j'}} d_{i'}$, where c_j is d_i 's partner and $c_{j'}$ is $d_{i'}$'s partner.

The unstable matching will cause instability of network applications and user dissatisfaction. Hence, finding a stable matching by preference of D2D users and CUs ensures that neither D2D nor cellular users have a better choice than the current one, which is defined as follows:

Definition 3 A matching M without any blocking pair is one-to-one stable.

The Gale-Shapley algorithm can be used to solve the stable matching problem [12]. The main strategy of channel allocation based on GS algorithm can be formulated as follows: 1) Input the D2D users' preference lists $\mathcal{PL}_i^D, \forall d_i \in \mathcal{D}$ and the cellular user's preference list $\mathcal{PL}_j^C, \forall c_j \in \mathcal{C}$. 2) D2D user d_i have a sequence of proposals to the c_j . d_i proposes, in order, to the c_j in its preference list, pausing when c_j agrees to consider d_i 's proposal, but continuing if the proposal is rejected. 3) When receiving a proposal, c_j rejects if the CU already has a better choice, and otherwise agrees to hold it for consideration. 4) The progress ends until each D2D user find a partner. The computation complexity of GS algorithm is $\mathcal{O}(n^2)$, where n is the number of stable matching pairs.

GS algorithm is based on the non-cooperative game to find a stable matching. In other words, that is to pursue the maximization of personal interests. Thus, with the increase of the D2D users and CUs, more D2D users cannot be matched to their first choice according to the preferences list, which decreases the throughput of D2D users. Later in Section 4, the Cheating algorithm based on cooperative game is presented to solve the problem.

4 Cooperative game-based cheating algorithm

In Section 3, we have considered the stable matching problem and solve the problem using GS algorithm. In this section, we discuss some post-matching strategies under the result of GS algorithm, in which some D2D users can lie on their preference lists in order to be matched to more satisfied partners. Hence, we introduces a Cheating algorithm based on cooperative game, which can improve the whole interests by falsifying the D2D user's preference lists. First, we introduce some notation, definitions and terminology of Cheating algorithm, which are the basis of falsifying D2D user's preference lists. Afterwards, we propose a heuristic algorithm to further improve the performance of Cheating algorithm. Finally, we analyze the properties of our proposed algorithm.

4.1 Preliminaries

In this subsection, before developing our proposed Cheating algorithm in details, we establish some notation and terminology. We assume that the Gale-Sharpley men-optimal (i.e., D2D users optimal) algorithm is used and that we know the preference lists of all D2D users and CUs. When everyone is honest, M_0 is the men-optimal stable matching, and M_S represents the men-optimal matching when some subset of people cheat. For any stable matching M and some subset of matching members $S \subseteq \mathcal{D} \cup \mathcal{C}$, the partners

of S can be represented as $M(S)$. For instance, $M_0(d_i)$ is the partner of D2D user d_i in the men-optimal stable matching.

In addition, for D2D user d_i , its preference list in men-optimal stable matching can be divided into $(PL(d_i), M_0(d_i), PR(d_i))$, where $PL(d_i)$ and $PR(d_i)$ are respectively a set of CUs being more or less preferred than $M_0(d_i)$ by D2D user d_i . Since the preference list is ranked by a descending order, the set of CUs in $PL(d_i)$ (or $PR(d_i)$) are on the left (right) of D2D user d_i 's preference list.

Assuming that A is a set of distinct objects, let $\pi(A)$ denote the set of all $|A|!$ permutations and $\pi_r(A)$ denote a random permutation from this set. Similar with the relation of "prefer" in definition 1, if for any D2D user $d_i \in \mathcal{D}$, $M(d_i) \succeq_{d_i} M_0(d_i)$, we can say stable matching M is "at least as good as" stable matching M_0 , which is denoted as $M \succeq M_0$. Moreover, if there exists at least one D2D user and $M(d_i) \succ_{d_i} M_0(d_i)$, stable matching M is said to be strictly better than M_0 , which is denoted as $M \succ M_0$.

4.2 Falsifying preference list

The general idea of Cheating algorithm is presented as follows. 1) Find a cabal consisting of D2D user, in which each member prefers each other's partner to its own. 2) Find the accomplices for the cabal, who need to falsify their preference lists in order to assist the cabal. 3) Run the men-optimal stable matching algorithm with the falsified preferences. Thus, in the resulting matching, all D2D users within cabal are better off while the rest of D2D users keep the same partners [15].

Before plunging into technical details, we present a Theorem and two definitions about Cheating algorithm and give a proof of them based on [17], which presented the Cheating algorithm for the first time.

Theorem 1 For a subset of D2D users $\mathcal{D} = \{d_1, \dots, d_i, \dots, d_l\}$, $1 \leq i \leq l$, if every member $d_i \in \mathcal{D}$, falsifies preference list according to $(\pi_r(PL(d_i) - X), M_0(d_i), \pi_r(PR(d_i) + X))$ then $M_S \succeq M_0$.

Proof Theorem 1 introduces the way of falsifying preference list in Cheating algorithm. For a subset of CUs X , $X \subseteq PL(d_i)$. Since the members in X have already refused the proposal of D2D user d , shifting the set X from $PL(d_i)$ to $PR(d_i)$ and then giving a random permutation of $(PL(d_i) - X)$ and $(PR(d_i) + X)$, the current men-optimal matching after cheating M_S is at least as good as the previous men-optimal matching M_0 . We proceed by contradiction.

In M_S , suppose at least one D2D user d_i gets a worse partner than $M_0(d_i)$. In general, assume that during the

execution of the algorithm with unfalsified lists, d_i is rejected by its M_0 -partner. In other words, d_i 's M_0 -partner has accepted another D2D user d'_i , who ranks higher than d_i in $M_0(d_i)$'s preference list. Now that d'_i has not been accepted by $M_0(d'_i)$, he must prefer $M_0(d_i)$ to $M_0(d'_i)$. Hence, $(d'_i, M_0(d_i))$ compose a blocking pair in M_0 . Since M_0 is a stable matching, the previous assumption is not established. Therefore, M_S is at least as good as the previous men-optimal matching M_0 . $\square$

Definition 4 The cabal $\mathcal{K} = \{d_1, \dots, d_m, \dots, d_k\}$, $1 \leq k \leq N$, is a subset of $\mathcal{D}$, within which each D2D pair d_m , $1 \leq m \leq k$, $m - 1 = k$ when $m = 1$, $M_0(d_{m-1}) \succ_{d_m} M_0(d_m)$.

Definition 4 gives the requirement of cabal members. It indicates that any cabal member $d_m \in \mathcal{K}$ prefers $M_0(d_{m-1})$ to $M_0(d_m)$, where $M_0(d_m)$ is d_m 's current partner and $M_0(d_{m-1})$ is d_m 's desired partner. In other words, that is to find a end to end loop with direction, where each member is eager to be matched to the partner of the member in previous position and the member in the first position is eager to be matched to the partner of the member in the end position.

Definition 5 The accomplices of cabal $\mathcal{K}$ is a set of D2D pairs $\mathcal{H}$, such that $d \in \mathcal{H}$:

- 1: if $d \notin \mathcal{K}$, for $d_m \in \mathcal{K}$, $M_0(d_m) \succ_d M_0(d)$ & $d \succ_{M_0(d_m)} d_{m+1}$;
- 2: if $d = d_l \in \mathcal{K}$, for $d_m \in \mathcal{K}$, $m \neq l$, $M_0(d_m) \succ_{d_l} M_0(d_{l-1})$ & $d_l \succ_{M_0(d_m)} d_{m+1}$.

Definition 5 gives the requirements of the accomplices within or outside the cabal, respectively. It defines the subset of D2D users $\mathcal{H}$ as the accomplices, who need to falsify their preference lists to assist cabal $\mathcal{K}$ to get their desired partners. For any D2D user d outside the cabal, who would have prevented a cabal member d_m from getting its desired partner, is defined as a accomplice of $\mathcal{K}$. Here, we say d prevents d_m when d prefers the $M_0(d_m)$ to its own partner, while $M_0(d_m)$ prefers d to d_m . Similarly, for any D2D user d within the cabal, which is denoted as d_l , who would have prevented another cabal member d_m from getting its desired partner, is defined as an accomplice. Thus, we say d_l prevents d_m when d_l prefers the $M_0(d_m)$ to its desired partner $M_0(d_{l-1})$, while $M_0(d_m)$ prefers d_l to d_m . Since d_l would not get its desired partner d_{l-1} , we say d_m prevents d_l as well.

Briefly, we find a cabal by Definition 4 and the accomplices for the cabal by Definition 5. Then, we falsify the preference list of accomplices by Theorem 1. Finally, we run the men-optimal stable matching algorithm with the falsified preferences. In the resulting M_S matching, D2D users

outside the cabal can hold their M_0 -partners and D2D users in the cabal can get their desired partners.

4.3 Cheating algorithm based on HLLSBD

From analysis above, users in cabal would improve their benefits by cheating, while other users' benefits stay unchanged. Thus the larger the size of cabal is, the more users would have a performance improvement. Hence, to further improve the D2D users' throughput, we try to find a cabal with bigger size. From the definition 4, if we let the D2D user d point to the other D2D users whose M_0 -partners are preferred by the d to d 's M_0 -partner, each user will have a directional relationship. Hence, the D2D pairs are represented as nodes and the directional relationships are denoted by edges, respectively. The cabal can be abstracted as a end to end loop in the graph containing different nodes and edges.

Therefore, the optimization problem can be abstracted as finding a largest loop in a directed graph [3, 8]. Due to the NP-hardness of finding the largest loop in a directed graph [1, 4, 7, 20], we cannot find a largest loop within a polynomial time. Hence, we can only find a near-optimal solution by improving the probability of finding the largest loop. The exist methods mostly use the color-coding technique to find loop with logarithmic length, if they exist. Using color-coding, the dependence on path length can be reduced to singly exponential [2, 6, 19, 27]. But this gives an approximation ratio of only $\mathcal{O}(n/\log n)$ [2].

Inspired by above works, we want to find a largest loop with a relatively high probability and low complexity. It is proved that depth first search (DFS) [5] algorithm can also be used to find loop. However, the key problem is how to apply the DFS to finding the largest loop. To avoid traversing the directional relationship more than once, we propose the HLLSBD (Heuristic Largest Loop Searching Based on DFS) algorithm which using different colors to represent the different statements of D2D users. Since the proposed algorithm traverse the directional relationship only once, it can find a largest loop with higher probability and lower complexity. The computation complexity of the algorithm is $\mathcal{O}(n + v)$, where n is the number of nodes, representing the number of D2D users; v is the number of edges, representing the number of all the directional relationship between the D2D users.

Compared with the algorithm in [15], which randomly find a loop for each possible D2D user and keep the loop with larger size each time after they finish one search, our algorithm can find a larger loop with lower complexity. Because the computation complexity of the algorithm in [15] is unpredictable. Now we present how to use the heuristic algorithm to find a larger loop in Algorithm 1.

Algorithm 1 Heuristic Largest Loop Searching Based on DFS(HLLSBD)

Input: The D2D users' set $\mathcal{D} = \{d_1, \dots, d_i, \dots, d_N\}$.

Output: The cabal $\mathcal{K}$.

```

1: Step 1: Set up the color list  $\mathcal{S} = \{s_1, \dots, s_i, \dots, s_N\}$  for
   the D2D users' set. Every D2D user has three kinds
   of colors: Color white, denoted by number 0, indicates
   that the D2D user has not been explored. Color gray,
   denoted by number 1, indicates that the D2D user is
   being explored. Color black, denoted by number -1,
   indicates that the D2D user has been explored. The col-
   ors of all D2D users in  $\mathcal{S}$  are white at first.  $X(d_i)$  is the
   parent node of  $d_i$  when being explored.
2: Step 2: Begin to traverse.
3: while  $\exists d_i \in \mathcal{D}, s_i \neq -1$  do
4:   Explore  $d_i$  and let  $k = i, s_i = 1$ .
5:   Step 3: Dyeing according to different situations.
6:   if  $d_k$  has any unexplored children node  $d_j$  then
7:     if  $s_j = 0$  then
8:       Explore  $d_j$  and let  $s_j = 1, k = j$ . Goto
       step3.
9:     else if  $s_j = 1$  then
10:      There exists a loop. Record the path and
       explore the next children node. Goto step3.
11:    else if  $s_j = -1$  then
12:      There exists no loop. Explore the next chil-
       dren node. Goto step3.
13:    end if
14:    else
15:      The  $d_k$  has been explored. Let  $s_k = -1$ .
16:      Step 4: Backtracking to parent node
17:      if  $X(d_k) \neq \emptyset$  then
18:        Traverse the other children nodes of  $d_k$ 's
        parent node recursively and let  $k = X(d_k)$ . Goto step3.
19:      end if
20:    end if
21:  end while
22: Choose the largest loop from overall loops found as the
   cabal  $\mathcal{K}$ .

```

The Algorithm 1 can be described briefly as follows.

- First of all, we introduce the meaning of different colors. The white color represents the node has not been explored. Dyeing gray represents the node is being explored. Dying black represents the node has been explored.
- Afterwards, we list all situations that can be encountered in traversal. 1) When finding a white node in traversal, we dye it gray. 2) When finding a gray node in travel, it indicates that the node has been explored before. So we find a end to end loop. Here, we rather backdate than traverse along the node, which ensuring that we don't find a loop repeatedly and find a larger

- loop possibly. 3) When finding a black node, it indicates that the node is not in any loop or the loop containing the node has been explored. Here, the backtracking is necessary to avoid finding a loop repeatedly.
- Finally, the node whose children nodes have been explored is dyed black. The traversal ends until all nodes have been dyed black. We choose the largest loop from overall loops found as cabal.

After finding a cabal, we presents a Cheating algorithm with larger cabal in Algorithm 2. We know that M_0 is the men-optimal stable matching if and only if M_0 contains no cabal, which indicates that D2D users in M_0 not necessarily falsify their preference lists. To further reduce the computation complexity, we add a decision condition before finding accomplices.

Algorithm 2 Cheating Based on HLLSBD

Input: D2D users' set $\mathcal{D} = \{d_1, \dots, d_i, \dots, d_N\}$, Men-Optimal stable matching M_0 .

Output: Stable matching M_S .

```

1: Initialization
2: Set  $M_S = M_0$ .
3: repeat
4: Step 1: Find a loop as  $\mathcal{K}$  for  $M_S$  using Algorithm 1.
5: if  $M_S$  contains a cabal then
6:   Step 2: Find cabal  $\mathcal{K}$ 's accomplices  $\mathcal{H}$  defined in Definition 5.
7:   Step 3: Falsify the preference lists.
8:   for  $d_i \in \mathcal{D}$ ,  $1 \leq i \leq N$  do
9:     if  $d_i \in \mathcal{H} - \mathcal{K}$  then
10:       $d_i$  falsifies preference list by shifting the set
       $X$  from  $\mathcal{PL}(d_i)$  to  $\mathcal{PR}(d_i)$  and then giving a random
      permutation of  $(\mathcal{PL}(d_i) - X)$  and  $(\mathcal{PR}(d_i) + X)$ , where
       $X = \{c | c = M_0(d_m) \in M_0(\mathcal{K}), d_i \succ_c d_{m+1}\}$ .
11:     else if  $d_i \in \mathcal{K}$  then
12:       $d_i$  falsifies preference list by shifting the
      set  $X$  from  $\mathcal{PL}(d_i)$  to  $\mathcal{PR}(d_i)$  and then giving a random
      permutation of  $(\mathcal{PL}(d_i) - X)$  and  $(\mathcal{PR}(d_i) + X)$ , where
       $X = \{c | c = M_0(d_m) \in M_0(\mathcal{K}), c \succ_{d_i} M_0(d_{i-1}) \& d_i \succ_c d_{m+1}\}$ .
13:     end if
14:   end for
15:   Step 4: Run the man-optimal GS algorithm with the
      falsified preference list, and output matching is  $M_S$ .
16: else
17:   Output the men-optimal stable matching  $M_S$ .
18: end if
19: until  $M_S$  contains no cabal.
  
```

In this way, we find a cabal with bigger size to benefit more D2D users with lower complexity, which achieves the optimization target.

4.4 Properties of the cheating algorithm based on HLLSBD

In this subsection, to evaluate the proposed algorithm, we analyze the properties in terms of effectiveness, stability, convergence and complexity in the following.

- 1) *Effectiveness:* The D2D users' throughput increases after our proposed cheating algorithm.

Proof Since we have proceeded the Theorem 1 by contradiction, the throughput performance of our proposed cheating algorithm is at least as good as the previous men-optimal GS algorithm. In addition, after the GS algorithm, some D2D users always can not be matched to their desired partners, which gives us the chance to outperform the GS algorithm in throughput. So we can conclude that our proposed algorithm can always increase the D2D users' throughput. $\square$

- 2) *Stability:* The proposed Algorithm 2 can obtain a one-to-one stable matching M_S .

Proof According to Definition 3, the matching M without any blocking pair is stable. In order to prove the stability of the proposed algorithm 2, we assume that there exists a blocking pair in the final matching M_S satisfying that $\exists d_i, d_{i'} \in \mathcal{D}, M_S(d_{i'}) \succ_{d_i} M_S(d_i) \& d_i \succ_{M_S(d_{i'})} d_{i'}$. According to algorithm 2, the cheating algorithm needs to run the GS algorithm in the end. If $(d_i, M_S(d_{i'}))$ is an blocking pair, there will exist this two cases in matching process. Cases 1 is that d_i never proposed to $M_S(d_{i'})$, which indicates that d_i prefers $M_S(d_i)$ to $M_S(d_{i'})$. Case 2 is that d_i has proposed to $M_S(d_{i'})$ while $M_S(d_{i'})$ rejected d_i . The rejection also indicates that d_i prefers $M_S(d_i)$ to $M_S(d_{i'})$. In either case $(d_i, M_S(d_{i'}))$ can not compose a blocking pair, which causes conflict. Therefore, we conclude the proposed algorithm can reach the one-to-one stability in the end of the algorithm. $\square$

- 3) *Convergence:* From the proof of effectiveness, we find that the D2D users' throughput increases after each successful swap operation in cabal. Since the throughput has an upper bound due to limited spectrum resources, the swap operations stop when the maximum throughput is saturated. Therefore, within limited number of rounds, the matching will converge to the final state which is stable. However, there is little room for us to further increase the throughput after the first swap. In this case, it's not worthwhile to sacrifice complexity performance in exchange for throughput performance. Hence, we just swap once rather than iterate to the final state.

- 4) **Complexity:** The computation complexity of Algorithm 2 consists of the following parts. The first part is finding cabal using Algorithm 1, whose computational complexity is $\mathcal{O}(n + v)$. n is the number of D2D users and v is the number of all the relationship between the D2D users. However, the directional relationships among all D2D users are very simple after GS algorithm, which indicates that v is much less than n^2 . The second part is finding accomplices, whose computation complexity is $\mathcal{O}(n)$. The third part is falsifying preference lists, whose computation complexity is $\mathcal{O}(n)$. The final part is GS algorithm, whose computation complexity is $\mathcal{O}(n^2)$. Hence, the total computation complexity of Algorithm 2 is $\mathcal{O}(n^2)$. More importantly, if M_0 is man-optimal stable matching, the computation complexity of Algorithm 2 is only $\mathcal{O}(n + v)$. It is obvious that the computation complexity of our algorithm is lower than the Hungarian algorithm and the Cheating algorithm in [15].

5 Simulation result

In this section, considering a single cell multi-user scenario, we evaluate the performance our algorithms in the following way: how our algorithms work and where the advantages of our algorithm are shown. First, we show how much the D2D users' throughput is improved by our algorithm compared to the GS algorithm [12], the existing Cheating algorithms [14, 15] and the Hungarian algorithm [32], respectively. Then we show how individual D2D user is improved by our cheating algorithm. Afterwards, to further illustrate the importance of cabal size, we compare the ratio of cabal members in D2D users using different algorithm. Finally, the probability of finding the largest cabal in a more complex case is presented to reflect the advantage of our algorithm. The benchmark algorithms can be briefly described as follows:

The Gale-Sharpley algorithm can obtain a stable matching in a lower computation complexity. The Cheating algorithm in [14] can be described as "Cheating with random cabal", which random search for a cabal ends until a loop is found no matter how large this cabal is. The Cheating algorithm in [15] can be described as "Cheating with larger cabal", which random search for a cabal for each D2D user and keep the loop with larger size each time after they finish one search. The Hungarian algorithm can achieve the theoretical maximum throughput in an unstable matching, which can be used as a benchmark. Simulation parameters are shown in Table 1.

In Fig. 3, we evaluate the D2D users' throughput (i.e., the sum of all D2D users' transmission rate) with different number of D2D users using different algorithms. The 10000 times Monte-Carlo simulation shows that with the

Table 1 Simulation parameters

Parameters	Value
Distance between BS and UE-R	150-200m
Distance between BS and CU	150-200m
Distance between BS and UE-D	150-200m
Distance between CU and UE-R	150-200m
Distance between UE-D and UE-R	25-50m
The number of D2D pairs	20-40
The number of CUs	20-40
Maximum transmission power for D2D user	24dbm
Transmissiom power of BS for each UE-R	30dbm
CNR of the residual LI channel	10dB
Noise spectral density	-173dBm/Hz
Downlink bandwidth assigned to each CU	0.1MHz
Minimum rate requirement of CUs	20kbps
Pathloss factor	4
Simulation times	10000

increase of the number of D2D users/CUs, the total throughput of D2D users increases continuously. Since with the increase of the number, the average throughput is almost unchanged, but the increase in the number of D2D links improve the throughput. The Hungarian algorithm can find a maximum weight matching, which achieves the most D2D users' throughput. The other four curves, i.e., the Gale-Sharpley curve, Cheating with random cabal [14] curve, Cheating with larger cabal [15] curve and Cheating based on HLLSBD curve, they achieve almost 87.01%, 88.74%, 92.18% and 93.82% of the optimal system throughput by Hungarian algorithm, respectively. With such performance, we could say that our algorithms are close optimal while ensuring the system stability. Besides, the computation complexity of our algorithm is less than the Hungarian algorithm and the Cheating with larger cabal [15]. Comparing the three Cheating curves to the one without Cheating, the most important is that with the increasing number of D2D users, the performance of our algorithm becomes better.

Figure 4 shows how exactly an individual D2D user's satisfaction (w.r.t. the ranking of partners) is improved using different algorithms. The 10000 times Monte-Carlo simulation give the average numbers of 20 D2D pairs that match to their k th choice in the preference list. If every user is honest, averagely 5.83 users get their favorite partners and 4.37 users match to the second choice. In case of cheating, we can have more than 7.53 users match to the first choice and 4.57 to the second. Comparing the three Cheating algorithm, our algorithm can make more users match to their first and second choice, which intuitively reflects the advantage of our algorithm.

To further illustrate the importance of cabal size, in Fig. 5, we compare the ratio of cabal members in D2D users

Fig. 3 D2D user Throughput comparison with different number of D2D users

between Cheating with random cabal, Cheating with larger cabal and Cheating based on HLLSBD. With the increase of the number of D2D users/CUs, the ratio of random cabal remains unchanged. The reason is that, the random search for a cabal in [14] ends until a loop is found no matter how large this cabal is. Although the number of D2D users increases, the cabal size with the random search not necessarily increases. As for the search for larger cabal in [15] (in their work, they have gone through all possible D2D members during the search), with the number of D2D users increase, the cabal size increases as well. It is obvious that the ratio increases a lot for exhaustive search. But the trend is increasing slowly. As for our algorithm, the Cheating based on HLLSBD yields a highest ratio among the three

Fig. 5 The ratio of cabal member in D2D users with different number of D2D pairs

algorithm. Meanwhile, the ratio of cabal size keeps a sub-stain increase, which indicates that our algorithm can always find a larger cabal.

In Fig. 6, we evaluate the probability of finding the largest cabal under different number of D2D pairs. It is not hard to understand that the directional relationships among all D2D users are very simple after men-optimal stable matching, which can be abstracted as the degree in graph theory. The degree of the vertex is defined as the number of all the edges containing the vertex. The lower the degree, the simpler the directional relationships. The mathematical expectation of each node degree accounts for 5% of the number of all nodes. In this case, the probability of finding the largest cabal is almost 100%. Therefore, we

Fig. 4 D2D partners' distribution with different algorithm

Fig. 6 The probability of finding the largest cabal with different number of D2D pairs

consider a more complex case. We evaluate the performance of our algorithm when the mathematical expectation of each node degree accounts for 20% of all nodes. It can be seen from Fig. 6 that with the increase of the number of D2D users/CUs, the directional relationships become more and more complex, which resulting in the decrease in the probability of finding the largest cabal. Compared with the algorithm in [14] and [15], our algorithm can find the largest cabal with higher probability. It is proved that our algorithm has the advantage in the case of more complex directional relationships.

6 Conclusion

In this paper, we introduce the FD relaying-based D2D communication scheme in multi-user heterogeneous cellular network, which can improve the spectrum resource utilization dramatically. To further improve the throughput of D2D users, we implement matching theory into this scenario to solve the resource allocation problem. The GS stable matching algorithm is provided to solve the problem. The GS algorithm could reach 87.01% (under our simulation set up) of the maximum D2D throughput by the Hungarian algorithm in polynomial time. In addition, We introduce a Cheating algorithm based on cooperative game in order to further improve some D2D users' throughput. More importantly, due to the NP-hardness of our optimization target, we cannot find the optimal solution within a polynomial time. Hence, using different colors to represent the different statements of D2D users, we develop a heuristic algorithm with low complexity to find a near-optimal solution of Cheating algorithm.

The simulation results show that the throughput performance becomes better with the bigger cabal size. The performance increases by 5.08% and 1.64% for the D2D throughput, respectively, compared to the cheating algorithm in [14] and [15]. It is proved that our algorithm has better complexity and throughput performance compared with the existing algorithms.

Acknowledgements This work is supported by the Fundamental Research Funds for Central Universities (2014ZDPY16).

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

References

1. Combinatorial optimization: polyhedra and efficiency. Springer (2004)
2. Alon N, Yuster R, Zwick U (1994) Color-coding: a new method for finding simple paths, cycles and other small subgraphs within large graphs. In: Twenty-Sixth ACM symposium on theory of computing. Montréal, pp 326–335
3. Bangjensen BJ, Gutin G (2010) Directed graphs: theory, algorithms and applications. Springer Berlin 12(8):44–51
4. Björklund A, Husfeldt T, Khanna S (2004) Approximating longest directed paths and cycles. *Lect Notes Comput Sci* 3142(1–2):222–233
5. Bodlaender HL (1993) On linear time minor tests with depth-first search. *J Algor* 14(1):1–23
6. Chen J, Lu S, Sze SH, Zhang F (2007) Improved algorithms for path, matching, and packing problems. In: Eighteenth Acm-Siam symposium on discrete algorithms, SODA 2007. New Orleans, pp 298–307
7. Cormen TH, Leiserson CE, Rivest RL, Stein C (2001) Introduction to algorithms, 2nd edn. pp 1297–1305
8. Diestel R (2011) Graph theory. *Math Gazette* 173(502):67–128
9. Doppler K, Rinne M, Wijting C, Ribeiro CB (2010) Device-to-device communication as an underlay to lte-advanced networks. *IEEE Commun Mag* 47(12):42–49
10. Duarte M (2012) Full-duplex wireless: design, implementation and characterization. *Dissertations & Theses - Gradworks*
11. Fodor G, Reider N (2011) A distributed power control scheme for cellular network assisted d2d communications. In: Global communications conference, GLOBECOM 2011. Houston, pp 1–6
12. Gale D, Shapley LS (2013) College admissions and the stability of marriage. *Amer Math Monthly* 120(69):9–15
13. Goldsmith A (2005) Wireless communications, 1st edn
14. Gu Y, Zhang Y, Pan M, Han Z (2015) Cheating in matching of device to device pairs in cellular network. In: Global communications conference, pp 4910–4915
15. Gu Y, Zhang Y, Pan M, Han Z (2015) Matching and cheating in device to device communications underlying cellular networks. *IEEE J Select Areas Commun* 33(10):1–1
16. Gupta A, Jha RK (2015) A survey of 5g network: architecture and emerging technologies. *Access IEEE* 3:1206–1232
17. Huang CC (2006) Cheating by men in the Gale-Shapley stable matching algorithm. Springer, Berlin
18. Kaufman B, Aazhang B (2008) Cellular networks with an overlaid device to device network. In: Asilomar conference on signals, systems and computers, pp 1537–1541
19. Koutis I (2008) Faster algebraic algorithms for path and packing problems. In: International colloquium on automata, languages and programming, pp 575–586
20. Lawler EL (2001) Combinatorial optimization. *Netw Matroids* 84(3):461–463
21. Manlove DF (2013) Algorithmics of matching under preferences. world Scientific Pub Co
22. Massiot GCL, Vercouteren J, Men-Olivier LL, Lévy J, Guilhem J, Pascard C (2015) Game theory based radio resource allocation for full-duplex systems. In: Vehicular technology conference, pp 1–5
23. Pei Y, Liang YC (2013) Resource allocation for device-to-device communications overlaying two-way cellular networks. *IEEE Trans Wirel Commun* 12(7):3611–3621

24. Riihonen T, Werner S, Wichman R (2011) Hybrid full-duplex/half-duplex relaying with transmit power adaptation. *Wireless Commun IEEE Trans* 10(9):3074–3085
25. Riihonen T, Werner S, Wichman R (2011) Mitigation of loopback self-interference in full-duplex mimo relays. *Signal Process IEEE Trans* 59(12):5983–5993
26. Wang F, Song L, Han Z, Zhao Q (2013) Joint scheduling and resource allocation for device-to-device underlay communication. In: *Wireless communications and networking conference*, pp 134–139
27. Williams R (2009) Finding paths of length k in $O(2^k)$ time. *Inf Process Lett* 109(6):315–318
28. Wu D, He J, Wang H, Wang C (2015) A hierarchical packet forwarding mechanism for energy harvesting wireless sensor networks. *IEEE Commun Mag* 53(8):92–98
29. Wu D, Wang Y, Wang H, Yang B (2016) Dynamic coding control in social intermittent connectivity wireless networks. *IEEE Trans Veh Technol* 65:1–1
30. Wu D, Yan J, Wang H, Wu D, Wang R (2017) Social attribute aware incentive mechanisms for video distribution in device-to-device communications. *IEEE Trans Multimed*. <https://doi.org/10.1109/TMM.2017.2692648>
31. Wu D, Zhang P, Wang H, Wang C, Wang R (2016) Node service ability aware packet forwarding mechanism in intermittently connected wireless networks. *IEEE Trans Wirel Commun* 15(12):1–1
32. Xu C, Song L, Han Z, Zhao Q (2013) Efficiency resource allocation for device-to-device underlay communication systems: a reverse iterative combinatorial auction based approach. *Select Areas Commun IEEE J* 31(9):348–358
33. Yin R (2014) Joint spectrum and power allocation for d2d communications underlaying cellular networks. In: *ICC 2014 - 2014 IEEE International conference on communications*, pp 5342–5347
34. Yu CH, Doppler K, Ribeiro CB, Tirkkonen O (2011) Resource sharing optimization for device-to-device communication underlaying cellular networks. *Wireless Commun IEEE Trans* 10(8):2752–2763
35. Zhang G, Yang K, Liu P, Wei J (2014) Power allocation for full-duplex relaying-based d2d communication underlaying cellular networks. *Veh Technol IEEE Trans* 64(10):1–1
36. Zhang N, Cheng N, Gamage AT, Zhang K (2015) Cloud assisted hetnets toward 5g wireless networks. *IEEE Commun Mag* 53(6):59–65