# Intelligent Water Drops Based Joint Subcarrier Pairing and Power Allocation with Fairness in Cooperative Relay Networks

Guiyan Liu, Songtao Guo*, Huan Zhao, and Fei Wang

*Abstract:* **In this paper, we propose a joint subcarrier pairing and power allocation (JS2PA) scheme with fairness based on the *intelligent water drop (IWD)* optimization method for *orthogonal frequency division multiple access* cooperative relay networks. The proposed scheme consists of a subcarrier pairing and selection algorithm and a power allocation algorithm. We first formulate the JS2PA problem as a mixed integer programming problem aiming to maximize the total network utility under the constraints of individual power, subcarrier fairness requirement and pairing. To solve the non-convex JS2PA problem, firstly, we propose a subcarrier pairing and selection algorithm based on Hungarian method so as to select the appropriate subcarrier pairs for relaying. Secondly, we provide a power allocation algorithm based on the IWD method in which water drops act as the agents to find the optimal power allocation for each node. In particular, to improve the optimality of power allocation, we integrate network utility and power requirement into heuristic function to measure the desirability of the IWD selecting the next visiting node. Finally, we conduct simulations to validate the proposed algorithms and the results show that the proposed JS2PA scheme outperforms the existing methods in terms of convergence, total network utility and fairness.**

*Index Terms:* **Cooperative relay networks, fairness, intelligent water drop (IWD), power allocation, subcarrier pairing.**

## I. INTRODUCTION

COOPERATIVE relaying has recently attracted a lot of research interests as an emerging transmit strategy for next generation wireless communication networks. In cooperative relaying, relay nodes are employed to assist the transmission of source node by relaying the replica of the information. Such cooperative communication can effectively improve the reliability of wireless transmission and extend the coverage of nodes by exploiting the inherent spatial and multiuser diversities.

In cooperative relay networks, network performance depends on careful resource allocation, such as subcarrier pairing and power allocation, due to the scarcity of bandwidth, subcarrier and power. The subcarrier paring problem is to match the incoming and outgoing subcarriers at relay node based on channel dynamics to provide good system performance. The optimal solution to the problem can be obtained by using the well-known Hungarian algorithm in combinatorial optimization [1]. Some resource allocation algorithms, such as the Greedy algorithm [2], and the improved Greedy algorithm, including the worst user first (WUF) Greedy algorithm [3], and the Maximal Greedy algorithm [4], have been proposed, in which either users select their desired subcarriers one-by-one from the available pool based on the channel qualities of subcarriers, or the subcarriers are allocated by ordering the users in different ways. These methods can significantly improve or even achieve near-optimal error performance after a sufficient number of iterations. However, their complexity may become very high when the number of subcarriers and/or the number of iterations are large.

On the other hand, proper power allocation among mobile stations (MSs) and relay stations (RSs) can significantly reduce power consumption and extend network lifetime. Capacity and throughput can also be enhanced through cooperative resource sharing and scheduling among nodes within a network. At present, the joint subcarrier and power allocation problem is often formulated as an optimization problem of maximizing the system throughput or the weighted sum rate under the total sum power constraint and the individual power constraint, respectively [5]–[8]. The proposed strategies are based on Lagrangian dual decomposition method. The optimal solution by means of the dual method depends on the convexity of the investigated optimization problem. However, the optimization problem formulated by this way is generally difficult to solve due to that the discrete nature of subcarrier assignment usually leads to an integer programming problem, which is NP-hard.

Recently, a new swarm-based, nature-inspired optimization method, called *intelligent water drops (IWD)* method [9], has become a popular method for solving complex nonlinear optimization problem. The IWD method is a population-based optimization algorithm that imitates some of the processes that occur in nature between the water drops of a river and the soil of river bed and uses a constructive approach to find optimal solution of a given maximization/minimization problem. The IWD algorithm has been used for solving several NP-hard combinatorial optimization problems, such as robot path planning problem [10], multidimensional knapsack problem (MKP) [11], job-shop scheduling problem [12], and real-life waste collection problem [13]. Moreover, in [14], Alijla *et al.* proposed two

Manuscript received February 2, 2016; approved for publication by Prof. Kwang Soon Kim, Division II Editor, October 23, 2016.

This work was supported by the National Natural Science Foundation of China (No. 61373179, 61373178, 61402381), Natural Science Key Foundation of Chongqing (cstc2015jcyjBX0094), the Fundamental Research Funds for the Central Universities (XDJK2013C094, XDJK2013A018, XDJK2015C010, XDJK2015D023 ,XDJK2016A011), and Science and Technology Leading Talent Promotion Project of Chongqing (cstc2013kjrc-ljrccj40001).

G. Liu, S. Guo, and F. Wang are with the College of Electronic and Information Engineering, Southwest University, Chongqing, 400715, P. R. China, email: 1491257901@qq.com, {songtao_guo, wsf0107}@163.com.

H. Zhao is with the College of Computer, Chongqing University, Chongqing, 400044, P. R. China, email: zhaohuan@cqu.edu.cn.

Songtao Guo is the corresponding author.

Digital object identifier: 10.1109/JCN.2017.000004

ranking-based selection methods, namely linear ranking and exponential ranking, and investigated their effectiveness in the solution construction phase of the IWD algorithm. Furthermore, in [15], they proposed an IWD ensemble known as the master-river, multiple-creek IWD (MRMC-IWD) model to improve the exploration capability of the IWD algorithm. In [16], Mokhtari provided a nature inspired intelligent water drops evolutionary algorithm for parallel processor scheduling with rejection. A main advantage of this method is that it is able to solve the non-convex optimization and has low computational complexity and fast convergence.

In this paper, we formulate the joint subcarrier paring and power allocation (JS2PA) problem with fairness as a utility-based optimization framework with the objective of maximizing the total network utility subject to (i) the maximum transmission power of individual users, (ii) the subcarrier requirement constraint of each user, and (iii) the subcarrier paring constraint. Here, the fairness is achieved by associating each user with a fair utility function and setting a lower and upper bound on the number of subcarriers occupied by each user. In particular, the formulated JS2PA problem is a mixed integer programming problem, therefore, the conventional Lagrangian dual method is not suitable for solving this problem.

To solve the JS2PA problem in orthogonal frequency division multiple access (OFDMA) based cooperative relay networks, we propose a subcarrier pairing and selection algorithm based on Hungarian method to fairly select appropriate subcarriers for relaying, and a power allocation algorithm based on the IWD method, called PA-MIWD algorithm, which improves the IWD algorithm [9], by randomly initializing diverse soil and velocity, employing a bounded local soil update and using network utility and average power requirement to measure the desirability of IWD selection of the next visiting edge. To the best of our knowledge, this is the first work applying the IWD algorithm to solving the JS2PA problem in OFDMA-based cooperative relay networks. Finally, we conduct simulations to validate the proposed algorithms and the simulation results demonstrate that the proposed subcarrier pairing and selection algorithm plays an imperative role in improving network utility.

## II. SYSTEM MODEL AND PROBLEM FORMULATION

In this section, we present system model and formulate the optimization problem for joint subcarrier pairing and power allocation with fairness.

### A. System Model

We consider a multiuser OFDMA-based relay network with $K$ subcarriers, $J$ base stations (BSs)/source nodes, $N$ RS nodes and $M$ MS nodes, as shown in Fig. 1, and a downlink transmission scenario where a source node (BS node) is transmitting data to $M$ destination nodes (MS nodes) using $K$ subcarriers dynamically. A MS is served either by a direct connection between the BS and the MS or a two-hop connection from the BS, in which there is a link between the BS and a RS and a link between the RS and the MS. A RS cannot transmit on a subcarrier and concurrently receive on another subcarrier to eradicate well-built intercarrier interference. We assume that only the decode-and-forward (DF) cooperative relay is used. A transmission period is divided into two transmission phases. Since these two transmission phases are orthogonal to each other in the time domain, we call them the first time slot and the second time slot, respectively. In the first time slot, the BS transmits data to the MS or RS while the MS and RS receive data. In the second time slot, the RS transmits the data received from the BS in the first time slot to the MSs while the BS keeps transmitting data to the MSs via a direct link without relaying. Each subcarrier is subject to frequency-selective block fading, i.e., the channel state remains the same within two time slots. For clarity, the variables and notations used in this paper are summarized in Table 1.

For each subcarrier, only one node transmits data in a given time slot. Each subcarrier used by the BS in the first time slot

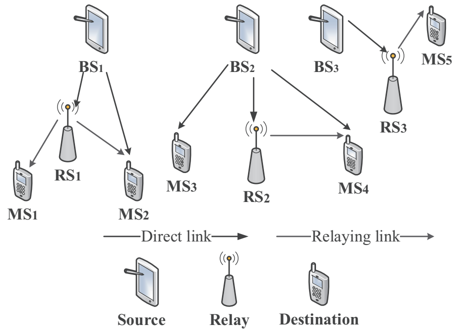



Fig. 1. A multiuser relay network with 3 relay stations and 5 mobile stations.

Table 1. List of notations.

| Notation | Definition |
|---|---|
| $K$ | Number of subcarriers |
| $N$ | Number of relay stations |
| $M$ | Number of mobile stations |
| $h_{irk}^{BR}$ | Channel gain from BS $i$ to RS $r$ on subcarrier $k$ |
| $h_{rjm}^{RM}$ | Channel gain from RS $r$ to MS $j$ on subcarrier $m$ |
| $h_{ijk}^{BM}$ | Channel gain from BS $i$ to MS $j$ on subcarrier $k$ |
| $\lambda_{ijk}^{BM}$ | Channel power gain between BS $i$ and MS $j$ |
| $\lambda_{irk}^{BR}$ | Channel power gain between BS $i$ and RS $r$ |
| $\lambda_{rjm}^{RM}$ | Channel power gain between RS $r$ and MS $j$ |
| $P_{ijk}^{BS}$ | BS power in the first time slot for $SP(k,m)$ |
| $P_{rjm}^{RS}$ | Relay power in the second time slot for $SP(k,m)$ |
| $soil(i,j)$ | Amount of soil on the link between two nodes |
| $soil^{IWD_g}$ | Soil of $g$th IWD |
| $soil^{IWD}$ | Soil of an IWD |
| $vel^{IWD_g}$ | Velocity of $g$th IWD |
| $N_{IWD}$ | Number of IWDs |
| $iter_{\max}$ | Maximum number of iterations |
| $p_{i,j}^{IWD_g}$ | Probability of choosing node $j$ for node $i$ |
| $\Delta soil(i,j)$ | Soil that an IWD loads from edge $(i,j)$, i.e., soil changes |
| $\Delta soil_{\min}$ | Lower bound for soil changes in any edge $(i,j)$ |
| $\Delta soil_{\max}$ | Upper bound for soil changes in any edge $(i,j)$ |
| $\rho_{IWD_g}$ | Global soil updating parameter |
| $\rho_L$ | Local soil update parameter |
| $T^{IB}$ | Iteration best schedule |
| $T^{TB}$ | Total (global) best solution |

is paired with another subcarrier used by the RS in the second time slot. Thus, the number of subcarrier pairs (SP) in the transmission is $K$. If subcarrier $k$ in the first time slot and subcarrier $m$ in the second time slot are paired, we denote them as $SP(k,m)$. The packets transmitted on different SPs are assumed to be independent. Subcarrier pair $SP(k,m)$ might not be the actual pair participating in the communication. If $SP(k,m)$ actually participates in the communication, it is said to be selected. We define the power gains of the channel between BS $i$ and MS $j$ as $\lambda_{ijk}^{BM} = \left|h_{ijk}^{BM}\right|^2/\sigma_{BM,ijk}^2$, BS $i$ and RS $r$ as $\lambda_{irk}^{BR} = \left|h_{irk}^{BR}\right|^2/\sigma_{BR,irk}^2$ on subcarrier $k$ and RS $r$ and MS $j$ as $\lambda_{rjm}^{RM} = \left|\lambda_{rjm}^{RM}\right|^2/\sigma_{RM,rjm}^2$ on subcarrier $m$, where $h_{ijk}^{BM}$, $h_{irk}^{BR}$ and $\lambda_{rjm}^{RM}$ are BS $i$ - MS $j$ and BS $i$ - RS $r$ channel gains on subcarrier $k$, and RS $r$ - MS $j$ channel gain on subcarrier $m$, respectively. Also, $\sigma_{BM,ijk}^2$, $\sigma_{BR,irk}^2$ and $\sigma_{RM,rjm}^2$ are the variances of the additive white Gaussian noise (AWGN) in the corresponding channels.

Note that the scheme of joint subcarrier paring and power allocation depends on not only the channel state but also the degree of knowledge of channel state. In this paper, the channel state information (CSI) at the BSs is required for cooperative relay transmission to gain their benefits. To obtain the knowledge of channel state, we utilize time-division duplex (TDD) for cooperative relay system since in TDD system, channel reciprocity can be used to train on the reverse link and obtain an estimate of the channel at the BS. In other words, in TDD system, channel reciprocity allows each BS to obtain the downlink channel to each user by estimating the uplink channel from the user. We formulate the optimization problem for the scenario with imperfect CSI, i.e., the transmitter knows the instantaneous CSI (ICSI) of the first hop and the statistical CSI (SCSI) of the second hop. To emphasize the impact of imperfect channel reciprocity on the benefits of data sharing, we assume that the channels are fully shared among the BSs via noiseless and zero latency backhaul links, and the data intended to the users are partially shared. The sharing of data and channel information among the coordinated BSs is achieved by the transmission via backhaul links.

### *B. Problem Formulation*

We assume that each subcarrier pair $SP(k,m)$ works in either the relay mode or the direct-link mode in a selective DF relay. In the relay mode, the half-duplex relay is active and forwards the decoded packets on subcarrier $k$ in the second time slot. In the direct-link mode, the relay does not forward packets and only the BS-MS link in the first time slot is used to transmit packets. Then the achievable end-to-end weighted rate of the link from BS $i$ to a MS by RS $r$ forwarding over $SP(k,m)$ under imperfect CSI can be expressed as [17]

$$R_{k,m}^{ij} = \begin{cases} \frac{w_k}{2}\log\left(1+\lambda_{ijk}^{BM}P_{ijk}^{BS}\right), \text{for direct-link mode} \\ \frac{w_k}{2}\min\{\log(1+\lambda_{irk}^{BR}P_{ijk}^{BS}), \\ \log(1+\lambda_{ijk}^{BM}P_{ijk}^{BS}+\lambda_{rjm}^{RM}P_{rjm}^{RS})\}, \text{for relay mode} \end{cases} \tag{1}$$

where $P_{ijk}^{BS}$ and $P_{rjm}^{RS}$ represent the power of BS $i$ in the first time slot and the power of RS $r$ in the second time slot, respectively. A weight $w_k$ is assigned to the rate on subcarrier $k$ to reflect QoS requirements. The rate is scaled by $1/2$ since the transmission takes two time slots. In the following, we given a criterion to decide the working mode of $SP(k,m)$ in selective DF mode similar to that in [20].

**Theorem 1:** Using relay model is beneficial when

$$\min\left(\lambda_{irk}^{BR},\lambda_{rjm}^{RM}\right) > \lambda_{ijk}^{BM}. \tag{2}$$

Otherwise, the relay keeps inactive on subcarrier $k$ in the relay phase.

*Proof:* The detailed proof can be found in Appendices A. □

Let $P_{k,m}^{ij} = P_{ijk}^{BS} + P_{rjm}^{RS}$ for $SP(k,m)$. We first consider rate $R_{k,m}^{ij}$ in the relaying mode. According to [20], we have that assuming the system has $N$ hops,

$$R_{k,m} = \min\{R_{k,m}^1, R_{k,m}^2, \cdots, R_{k,m}^N\}$$

The rate reaches its maximum when

$$R_{k,m}^1 = R_{k,m}^2 = \cdots = R_{k,m}^N.$$

That implies that the maximum achievable rate $R_{k,m}^{ij}$ for relay mode in (1) is reached when the following condition is satisfied:

$$\log(1+\lambda_{irk}^{BR}P_{ijk}^{BS}) = \log(1+\lambda_{ijk}^{BM}P_{ijk}^{BS}+\lambda_{rjm}^{RM}P_{rjm}^{RS}).$$

That is

$$\lambda_{irk}^{BR}P_{ijk}^{BS} = \lambda_{ijk}^{BM}P_{ijk}^{BS}+\lambda_{rjm}^{RM}P_{rjm}^{RS}. \tag{3}$$

Together with $P_{k,m}^{ij} = P_{ijk}^{BS} + P_{rjm}^{RS}$, we obtain

$$\begin{cases} P_{ijk}^{BS} = \frac{\lambda_{rjm}^{RM}}{\lambda_{irk}^{BR}+\lambda_{ijm}^{RM}-\lambda_{ijk}^{BM}}P_{k,m}^{ij}, \\ P_{rjm}^{RS} = \frac{\lambda_{irk}^{BR}-\lambda_{ijk}^{BM}}{\lambda_{ijk}^{BR}+\lambda_{rjm}^{RM}-\lambda_{ijk}^{BM}}P_{k,m}^{ij}. \end{cases} \tag{4}$$

In the direct-link mode, we can easily obtain $P_{ijk}^{BS} = P_{k,m}^{ij}$ and $P_{rjm}^{RS} = 0$. Let $\lambda_{k,m}^{ij}$ be the equivalent channel gain given by

$$\lambda_{k,m}^{ij} = \begin{cases} \frac{\lambda_{irk}^{BR}\lambda_{rjm}^{RM}}{\lambda_{irk}^{BR}+\lambda_{ijm}^{RM}-\lambda_{ijk}^{BM}}, \text{relaying mode}, \\ \lambda_{ijk}^{BM}, \text{direct-link mode}. \end{cases} \tag{5}$$

Accordingly, we can unify the rate of the link from BS $i$ to the MS $j$ by RS $r$ forwarding over $SP(k,m)$ as

$$R_{k,m}^{ij} = \frac{w_k}{2}\log\left(1+\lambda_{k,m}^{ij}P_{k,m}^{ij}\right). \tag{6}$$

We assume that each MS user $i$ is associated with a utility function $U_i$, which is a function of the total rate $R^i$, which is given by

$$R^i = \sum_{j=1}^{N}\sum_{k,m=1}^{K} R_{k,m}^{ij}$$

A utility function can reflect the satisfaction with resource allocation. Different shapes of utility functions can lead to different types of fairness among users [21]. Take the following utility function, parameterized by $\alpha > 0$, as an example

$$U_i^\alpha(R^i) = \begin{cases} (1-\alpha)^{-1}(R^i)^{1-\alpha}, & \text{if } \alpha \neq 1; \\ \log R^i, & \text{otherwise}. \end{cases} \tag{7}$$

If $\alpha = 0$, system throughput maximization is achieved; if $\alpha = 1$, proportional fairness among users is attained; aiming to maximize the total utility of the system can be formulated as follows:

$$\textbf{OPT-1:} \quad \max_{\mathbf{P},\mathbf{T}} \sum_{i=1}^{M} U_i(\sum_{j=1}^{N} \sum_{k,m=1}^{K} R_{k,m}^{ij}). \tag{8}$$

Subject to

$$\begin{aligned} \text{C1} &: \ P_{k,m}^{\min} \le t_{k,m} P_{k,m}^{ij} \le P_{k,m}^{\max}, \\ \text{C2} &: \ \rho_j^{\min} \le \sum_{m \in \phi(j)} t_{k,m} \le \rho_j^{\max}, \\ \text{C3} &: \ \sum_{k=1}^{K} t_{k,m} = 1, \forall m, \\ \text{C4} &: \ \sum_{m=1}^{K} t_{k,m} = 1, \forall k, \\ \text{C5} &: \ t_{k,m} \in \{0,1\}, P_{k,m} \ge 0, \forall k, m, \end{aligned} \tag{9}$$

where $P_{k,m}^{\min}$ and $P_{k,m}^{\max}$ are the lower bound and the upper bound of each subcarrier power $P_{k,m}^{ij}$ respectively, and $\phi(j)$ denotes the set of subcarriers allocated to user $j$. $\mathbf{P} \in \mathbf{R}_{+}^{\mathbf{K}\times\mathbf{K}}$ (with $\mathbf{R}_+$ denoting the set of nonnegative real numbers) and $\mathbf{T} \in \{\mathbf{0},\mathbf{1}\}^{\mathbf{K}\times\mathbf{K}}$ are matrices with entries $P_{k,m}^{ij}$ and $t_{k,m}$, respectively. C1 specifies the individual power constraint, i.e., power consumption over the subcarrier pair $(k,m)$ is bounded by $P_{k,m}^{\min}$ and $P_{k,m}^{\max}$ in order to guarantee quality of service and save energy. C2 is the subcarrier requirement constraint, i.e., the number of subcarriers allocated to user $j$ is bounded, which reflects the fairness of resource allocation and guarantees that each user may have an opportunity to access some subcarriers. C3 and C4 correspond to the pairing constraint that each subcarrier $k$ in the first time slot only pairs with one subcarrier $m$ in the second time slot.

Next, we will solve optimization problem **OPT-1**, which is divided into two subproblems. The one is the subcarrier pairing and selection subproblem, i.e. for a given power allocation strategy $\mathbf{P} = \{\mathbf{P}_{\mathbf{k},\mathbf{m}}^{\mathbf{ij}}\}$, we need to solve the following optimization problem:

$$\textbf{OPT-2:} \quad \max_{\mathbf{T}} \sum_{i=1}^{M} U_i(\sum_{j=1}^{N} \sum_{k,m=1}^{K} R_{k,m}^{ij})$$

subject to C2-C5, which will be solved in Section III. Here, the optimization variable is only $t_{k,m}$.

The other is the power allocation problem, i.e., for a given subcarrier pairing $SP(k,m)$ and subcarrier selection strategy $\mathbf{T} = \{\mathbf{t}_{\mathbf{k},\mathbf{m}}\}$, we need to solve the following optimization problem

$$\textbf{OPT-3:} \quad \max_{\mathbf{P}} \sum_{i=1}^{M} U_i(\sum_{j=1}^{N} \sum_{k,m=1}^{K} R_{k,m}^{ij})$$

subject to C1, which will be solved in Section IV. Here, the optimization variable is only $P_{k,m}^{ij}$.

It can be observed that OPT-1 is a mixed integer programming problem, which is generally unsolvable in polynomial time. Moreover, the utility functions in (8) may be nonconcave and nondifferentiable, the Lagrangian dual method may not be appropriate. Therefore, to solve OPT-1, we have to decompose the problem into two subproblems, OPT-2 and OPT-3, and further solve the two subproblems in a separate manner. However, the decomposition and separate optimization may cause performance loss, which is because the subcarrier selection variable $t_{k,m}$ and the power allocation variable $P_{k,m}^{ij}$ are coupled in OPT-1 so that the optimal solution to OPT-1 requires the joint optimization of the two variables $t_{k,m}$ and $P_{k,m}^{ij}$. This means that our proposed solution is suboptimal which sacrifices the performance for reducing complexity. In fact, the solution obtained by our method is asymptotically optimal and the gap between the suboptimal solution and optimal one can be negligible as the number of subcarriers becomes sufficiently large [24], [25].

## III. SUBCARRIER PAIRING AND SELECTION

The subcarrier pairing and selection problem aims at selecting appropriate subcarriers for the relay and finding the best subcarrier pairing between the source and the relay by solving the integer programming problem **OPT-2**, which is equivalent to determining an optimal matrix $\mathbf{T}$. In this section, we propose an algorithm based on Hungarian method to deal with the subcarrier pairing and selection problems jointly.

Subcarrier pairing and selection algorithms were studied in [20] and [26], where the proposed solutions are based on ordering the subcarrier gains: The best source-relay gain is paired with the best relay-destination. However, those subcarrier pairing algorithms are optimal only when all the subcarriers are available for relaying. In terms of subcarrier selection, a straightforward algorithm was introduced in [20], where a particular pair is used for relaying if $\lambda_{irk}^{BR}, \lambda_{rjm}^{RM} > \lambda_{ijk}^{BM}$; otherwise, direct link mode is used to transmit data. However, the straightforward algorithm is effective only for the total power constrained system.

To overcome the above shortcomings, we present a unified algorithm of subcarrier pairing and selection based on Hungarian method. Since the channel gains under imperfect CSI are deterministic, we can define the following matrices about subcarrier's channel power gains:

$$\mathbf{D}(\mathbf{k},\mathbf{m}) = \begin{cases} \frac{\lambda_{irk}^{BR}\lambda_{rjm}^{RM}}{\lambda_{irk}^{BR}+\lambda_{ijm}^{RM}-\lambda_{ijk}^{BM}}, & \text{if (2) is satisfied;} \\ \lambda_{ijk}^{BM}, & \text{otherwise,} \end{cases} \tag{10}$$

$$\mathbf{F}(\mathbf{k},\mathbf{m}) = \begin{cases} 1, & \text{if (2) is satisfied;} \\ -1, & \text{otherwise.} \end{cases} \tag{11}$$

We then have the following result on the integer programming problem.

**Theorem 2:** The integer programming problem for subcarrier pairing and selection can be reduced to an assignment problem on the matrix $\mathbf{D}$.

*Proof:* As aforementioned in Section II.A, each subcarrier is only assigned to one node to transmit in a given time slot and each subcarrier used by the source in the first time slot is paired with another subcarrier used by the relay in the second time slot to convey a packet. Fig. 2 depicts a simple subcarrier pairing and selection system with 4 subcarriers. As shown in

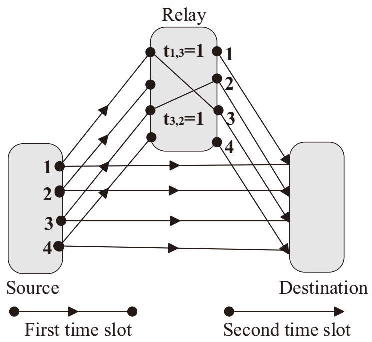



Fig. 2. Subcarrier pairing and selection system with 4 subcarriers.

Fig. 2, the source broadcasts data to the relay and destination on all the subcarriers in the first time slot, and subcarriers 1 and 3 are selected for relaying in the second time slot. Especially, subcarriers 1 in the first time slot is paired with subcarriers 3 in the second time slot, thus we have $t_{1,3} = 1$. Similarly, we have $t_{3,2} = 1$, which means that $\min\left(\lambda_3^{BR}, \lambda_2^{RM}\right) > \lambda_3^{BM}$, i.e., the inequality (2) is satisfied. Thus it is not difficult to observe that the key of the subcarrier pairing and selection is how to pair $K$ subcarriers for relaying and assign the paired subcarriers to $J$ source nodes and $N$ relay nodes by the known subcarriers' channel power gains such that a subcarrier is only assigned to one node for a given time slot, which is equivalent to finding an optimal assignment on the matrix $\mathbf{D}$ so that the total channel power gain or transmission rate is maximized. □

The above problem can be solved by applying the Hungarian method [1] and the corresponding algorithm of subcarrier pairing and selection is shown in Algorithm 1. It is worth noting that since the Hungarian method was initially designed for finding the minimum weight assignment, whereas finding the maximum is required in our subcarrier pairing and selection problem, we update the matrix $\mathbf{D}$ by replacing each $\mathbf{D}(\mathbf{k}, \mathbf{m})$ with $\max(\mathbf{D}(\mathbf{k}, \mathbf{m})) - \mathbf{D}(\mathbf{k}, \mathbf{m})$. Beside matrix $\mathbf{T}$ determined by the subcarrier pairing and selection algorithm, we can also obtain which subcarriers are used for direct link mode by flag matrix $\mathbf{F}$. The time complexity of the Hungarian based subcarrier pairing and selection algorithm is $O(K^3)$ for each source and relay. The fairness of the subcarrier assignment policy is achieved by removing the subcarrier with low channel power gain from the available subcarrier set $\phi(n)$ when the number of subcarriers allocated to a relay node $n$ exceeds the upper bound $\rho_n^{\max}$.

**Algorithm 1** Subcarrier Pairing and Selection

1: Initialize the channel power gains $\lambda_{ijk}^{BM}$, $\lambda_{irk}^{BR}$ and $\lambda_{rjm}^{RM}$ of all the channels between the BS $i$ and the RS $j$ over all subcarriers $k, m = 1, \cdots, K$; Initialize $\rho_j^{\min}$, $\rho_j^{\max}$ and $\phi(j)$ for each RS $j$;
2: **for** each source $i$ and relay $j$ **do**
3: //Subcarrier pairing
4: Construct the channel power gain matrix $\mathbf{D_{K\times K}}$ and flag matrix $\mathbf{F_{K\times K}}$ by (10) and (11) respectively;
5: Compute the maximum channel power gain $d_{\max} = \max(\mathbf{D}(\mathbf{k}, \mathbf{m}))$;
6: Let $\mathbf{D_{K\times K}} = [\mathbf{d}_{\max}]_{\mathbf{K\times K}} - \mathbf{D_{K\times K}}$;
7: Find the minimum of each row in the matrix $\mathbf{D_{K\times K}}$, and subtract it from all the entries of this row. At least one zero will appear on each row;
8: Find the minimum of each column in $\mathbf{D_{K\times K}}$, and subtract it from all the entries of this column. At least one zero will appear on each column;
9: Draw lines through appropriate rows and columns with the minimum number of such lines (horizontal or vertical) so that all the zero entries of the reduced matrix $\mathbf{D_{K\times K}}$ are covered;
10: Test for Optimality: (i) If the minimum number of covering lines is $K$, an optimal assignment of zeros is possible and the final solution is reached; (ii) If the minimum number of covering lines is less than $K$, an optimal assignment of zeros is not yet possible. In that case, proceed to Step 11;
11: Determine the minimum element not covered by any line. Subtract this element from each uncovered row, and then add it to each covered column. Return to Step 3;
12: Obtain the matrix $\mathbf{T}$ from the matrix $\mathbf{D}$ using the flag matrix $\mathbf{F}$, i.e., $\mathbf{T}(\mathbf{k}, \mathbf{m}) = \mathbf{1}$ if the $(k, m)$th term in the matrix $\mathbf{D}$ is equal to zero and selected in the optimal assignment of zeros; otherwise, $\mathbf{T}(\mathbf{k}, \mathbf{m}) = \mathbf{0}$.
13: //Subcarrier selection
14: Build the set $\phi(j)$ of subcarriers by inserting the elements with the value of "1" in the matrix $\mathbf{T}$ into the set;
15: **while** the number of subcarriers in $\phi(j) > \rho_j^{\max}$ **do**
16: Remove the subcarrier with low channel power gain from $\phi(j)$ while setting the term with the corresponding subcarrier pair as zero in the matrix $\mathbf{T}$;
17: **end while**
18: **Return** the set $\phi(j)$ and the matrix $\mathbf{T}$.
19: **end for**

## IV. POWER ALLOCATION

Having considered subcarrier pairing and selection, in this section, we focus on how to allocate appropriate power for a given subcarrier pair by solving the optimization problem **OPT-3** under the individual power constraint C1. To this end, we will propose a power allocation algorithm based on the IWD method, called PA-MIWD algorithm. We will first briefly review the IWD algorithm and then present our improved algorithm PA-MIWD.

### *A. Overview of IWD Algorithm*

The IWD algorithm is inspired by the movement of natural water drops which flow in rivers, lakes and seas. In the original IWD algorithm [9], an IWD is associated with two attributes, namely, the amount of soil it carries and the velocity that it is moving. The velocity of an IWD flowing over a path determines the amount of soil that is removed from the path. Faster water drops can gather and transfer more soil from the river beds. Besides, the velocity of the IWDs is also affected by the path condition. The amount of soil in a path has an impact on the IWD's soil collection and movement. A path with little amount of soil increases the velocity of the IWD more than a path with a considerable amount of soil, and the IWDs can attain a higher speed and collect more soil from that path, while a path with

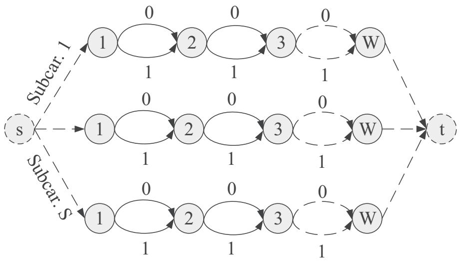



Fig. 3. A directed graph of IWDs travelling for a user with $S$ subcarriers.

more soil is the opposite.

In IWD algorithm, the movement of IWDs from a source to a destination is performed in discrete finite steps. From its current position to its next position, the IWD velocity is increased by the amount nonlinearly proportional to the inverse of the soil between the two positions. The soil increase is inversely and nonlinearly proportional to the time needed for the IWDs to pass the two locations. The time duration to travel from one location to the second location is calculated by simple laws of physics for linear motion, which is proportional to the velocity of the IWD and inversely proportional to the distance between the two positions. In the original IWD algorithm, the desirability of a path is reflected by the amount of soil in the path. When an IWD has to choose a path among several candidate paths, it would prefer an easier path, i.e., a path with less soil. The IWDs select a path based on a probabilistic function such that the probability to choose the next path is inversely proportional to the soils of the available paths.

### *B. PA-MIWD Algorithm*

To facilitate the operation of the IWD method, we need to build a directed graph $G = (V, E)$ to represent the power allocation for the JS2PA problem, where $V$ is the set of $S \times W$ nodes, and $E$ denotes the set of $2 \times (S \times W)$ directed edges, which resembles the rivers in the IWD algorithm. Here, $S$ is the number of the selected subcarriers for user $n$, i.e., $S = |\phi(n)|$ and $W$ denotes the precision that is employed to divide the allowed range for each subcarrier power $P_{k,m}^{ij}$. Assume that the range of the search space for $P_{k,m}^{ij}$ is between the low bound $P_{k,m}^{\min}$ and the upper bound $P_{k,m}^{\max}$, with $P_{k,m}^{\min} > 0$ and $P_{k,m}^{\max} < P_{ij}^{\max}$. Fig. 3 shows a directed disjunctive graph of IWDs travelling for a user with $S$ subcarriers, where the nodes $s$ and $t$ are fictitious nodes denoting the starting and ending positions of each IWD, respectively. A directed edge $e_{u,v}(P)$ connecting node $u$ to node $v$ is devoted to the bit with a subcarrier power $P_{k,m}^{ij}$, which can be 0 or 1. As a result, there will be two directed edges connecting $u$ to $v$. In each iteration, each IWD starts from node $s$ and visits every node in the disjunctive graph until it reaches node $t$. There will be at least one IWD to find the optimal power on a subcarrier. The number of IWDs is determined by the user. The solutions to power allocation are represented by the edges the IWDs have visited. Then every $W$ consecutively visited nodes in the graph represent a binary string with $W$ bits, i.e., a solution for the power value on a subcarrier.

We let $Iter_{\max}$ be the maximum number of iterations, and $N_{IWD}$ be the number of IWDs. In the PA-MIWD algorithm, the initial soil amount of each edge is a random number and the initial velocity of each IWD is also randomly chosen, which aims at providing the PA-MIWD algorithm with a diverse initial solution space. The pseudo-code of the PA-MIWD algorithm is given in Algorithm 2. As in Algorithm 2, the algorithm contains $Iter_{\max}$ iterations (Lines 3–15). For each iteration in the for loop, $N_{IWD}$ IWDs travel from the first node to the last node in graph $G$. The path of an IWD can produce a feasible solution (a power allocation strategy). The soils on the edges the IWDs pass, soils of the IWDs and velocities of the IWDs are updated during the traveling of the IWDs (Lines 5–8). After each iteration, all the IWDs have constructed their solutions and then each solution undergoes mutation based local search to improve its fitness value (Line 10). Furthermore, we find the iteration-best solution $T^{IB}$ from all the solutions (Line 11). Finally, the global best solution $T^{TB}$ is updated (Lines 12–13).

**Algorithm 2** PA-MIWD Algorithm

```
1:  Initialize an IWDs group A; // A population of IWDs
2:  Initialize static and dynamic parameters;
3:  while k < Iter_max do
4:      for each IWD_g ∈ A, for which a feasible solution has not been discovered do
5:          Select next edge (u, v) to visit according to a conditional probability (12) for the IWD_g;
6:          Update the velocity vel^{IWD_g} of the IWD_g by (15);
7:          Calculate the amount of soil, Δsoil(u, v), that the IWD_g removes from edge (u, v) by (19);
8:          Update the soil of the edge the IWD_g traversed, soil(u, v), by (20) and the soil contained in the IWD_g, soil^{IWD_g}, by (21);
9:      end for
10:     Execute mutation based local search to improve the fitness value of all discovered feasible solutions;
11:     Find the iteration-best solution T^{IB} from all the solutions T^{IWD} found by the IWDs by (22);
12:     Update the soils on the path associated with the best iteration solution T^{IB} by (23);
13:     Update the total best solution T^{TB} by the current iteration-best solution T^{IB} by (24);
14:     k++;
15: end while
```

We now discuss the main components of the PA-MIWD algorithm in more details.

#### B.1 Initializing Static and Dynamic Parameters

Static parameters are those that remain constant during the lifetime of the IWD algorithm, such as the maximum number of iterations $Iter_{\max}$, the number of water drops $N_{IWD}$, the initial soil amount of each edge, the initial velocity of each IWD, the quality of the total-best solution $T^{TB}$, the velocity updating parameters $a_v$, $b_v$, and $c_v$, the soil updating parameters $a_s$, $b_s$, and $c_s$, the local soil updating parameter $\rho_L$, the global soil updating parameter $\rho_{IWD_g}$, the upper bound $\Delta soil_{\max}$ and lower bound $\Delta soil_{\min}$ of soil change of each edge, etc.

Dynamic parameters are those parameters that are re-initialized after each iteration of the IWD algorithm, such as the visited node list $vc(IWD)$ of each IWD, the velocity and soil of each IWD, etc.

#### B.2 Choosing the Next Edge to Visit

An important function that each IWD must perform is to select its next node which does not violate any constraint of the power allocation problem and is not in the visited node list

$vc(IWD)$ of an IWD. An IWD prefers a path that contains less amount of soil and this preference is implemented by assigning a probability to each edge from the current node to its next node. Thus an IWD goes from one node to another node through one of edges (either 0 or 1-edge) depending on the probability which in turn is determined by the amount of soil on both edges. If the $g$th IWD, $IWD_g$, moves from node $u$ to its next node $v$ via the edge $e_{u,v}(P)$ (either 0 or 1-edge), the probability to go through the edge, $e_{u,v}(P)$, is given by

$$p_{u,v}^{IWD_g}(P) = \frac{f(soil(e_{u,v}(P))) \cdot (1/c_{u,v}(P))}{\sum_{l=0,1,v' \notin vc(IWD)} f(soil(e_{u,v'}(l)))}, \quad (12)$$

where $soil(e_{u,v}(P))$ denotes the amount of the soil on edge $e_{u,v}(P)$ and $f(soil(e_{u,v}(P)))$ computes the inverse of the soil on edge $e_{u,v}(P)$, i.e.,

$$f(soil(e_{u,v}(P))) = \frac{1}{\varepsilon_s + g(soil(e_{u,v}(P)))}. \quad (13)$$

The constant parameter $\varepsilon_s$ is a small positive number to avoid the possible division by zero in function $f(\cdot)$. $g(soil(e_{u,v}(P)))$ is used to shift the $soil(e_{u,v}(P))$ on edge $e_{u,v}(P)$ toward positive values and is calculated by

$$g(soil(e_{u,v}(P))) = \begin{cases} soil(e_{u,v}(P)), \text{ if } \phi \geq 0; \\ soil(e_{u,v}(P)) - \phi, \text{ else,} \end{cases} \quad (14)$$

where $\phi = \underset{l=0,1,v' \notin vc(IWD)}{min} (soil(e_{u,v'}(l)))$. It is worth noting that the original IWD algorithm computes this probability based on the soil on edges. In the PA-MIWD algorithm, to increase the convergence speed of the IWDs, namely, the speed of finding a best path, the probability is computed based on the soil of edges and the processing cost $c_{u,v}(P)$ of the candidate edge $e_{u,v}(P)$. By repeatedly applying the above rule, each IWD builds its own path.

B.3 Updating the Velocity of the IWD

For the $g$th IWD in node $u$ moving to the next node $v$ via edge $e_{u,v}(P)$, its velocity at time $t$ is updated by

$$vel^{IWD_g}(t+1) = vel^{IWD_g}(t) + \frac{a_v}{b_v + c_v \cdot soil^2(e_{u,v}(P))}, \quad (15)$$

where $a_v$, $b_v$, and $c_v$ are the velocity updating parameters to ensure that the value of the velocity is increased in the same scale of magnitude as the original velocity. According to the velocity updating in (15), the velocity of the IWD increases because $soil^2(e_{u,v}(P)) \geq 0$. The more the amount of the soil $soil(e_{u,v}(P))$ is, the less the updated velocity $vel^{IWD_g}(t+1)$ will be. If the value of the velocity increased is too large, the IWDs may be trapped in the local optima; if the value of the velocity increased is too small, the IWDs may need more time to obtain an allocation assignment. Besides, $b_v$ also guarantees that the equation is not divided by 0.

B.4 Calculating the Removed Soil from the Edge

We need to introduce a local heuristic function $HUD_{PA}$ to measure the undesirability of the $g$th IWD to move from node $u$ to node $v$ via edge $e_{u,v}(P)$ , which determines the local search ability. In our power allocation problem, we define the local heuristic function as the function of the utility in (7) and the power consumption, i.e.,

$$HUD_{PA}(e_{u,v}(P_v)) = \frac{P_v}{U_v(R)} \quad (16)$$

where $U_u(P_v)$ is the profit (utility) as defined in (7), when the $g$th IWD moves to node $v$ via edge $e_{u,v}(P)$, and $P_v$ denotes the power value obtained by IWD moving to node $v$. It is clear that for high utility, the undesirability measured by the heuristic function $HUD_{PA}$ becomes small whereas for low utility, it becomes large. On the contrary, high power consumption will lead to high undesirability. It is reminded that powers with high levels of undesirability are chosen fewer times than powers with low levels of undesirability.

The time taken for the $g$th IWD with velocity $vel^{IWD_g}(t+1)$ to move from the current node $u$ to its next node $v$ via edge $e_{u,v}(P)$, denoted by $time(u, v; vel^{IWD_g})$, is proportional to the heuristic function $HUD_{PA}$, i.e.,

$$time(u, v; vel^{IWD_g}) = \frac{HUD_{PA}(e_{u,v}(P)}{vel^{IWD_g}} \quad (17)$$

such that

$$vel^{IWD_g} = vel^{IWD_g}(t+1) + \begin{cases} \varepsilon, \text{ if } \left|vel^{IWD_g}(t+1)\right| < \varepsilon; \\ 0, \text{ otherwise.} \end{cases} \quad (18)$$

We can observe that for the JS2PA problem, the amount of the soil that the $g$th IWD with velocity $vel^{IWD_g}$ removes from its current path from node $u$ to its next node $v$ via edge $e_{u,v}(P)$, namely, $\Delta soil(e_{u,v}(P))$, is calculated by

$$\Delta soil(e_{u,v}(P)) = \frac{a_s}{b_s + c_s \cdot time(u, v; vel^{IWD_g})}, \quad (19)$$

where $a_s$, $b_s$, and $c_s$ are constant soil updating parameters to ensure the soil is increased in the same scale of magnitude as the original soil. Similar to the effect of the value of the velocity on IWDs, too much increase of the value of soil will cause IWDs to get trapped in the local optima, in contrast, the IWDs need more time to obtain an optimal allocation.

B.5 Updating the Soil of the Link the IWD Traverses and the Soil It Carries

The soil update model is one of the most important components of an IWD algorithm. To make full use of the guiding information and control, and the convergence rate of finding a path, compared to the soil-updating model in the previous IWD algorithm, we propose a bounded soil-update model by applying a lower and upper bound to the soil update process.

Let $\Delta soil_{\max}$ and $\Delta soil_{\min}$ be the upper bound and lower bound of soil changes respectively when the IWDs pass through any link. The lower bound (a small positive constant) prevents the algorithm from slow convergence, while the upper bound prevents the algorithm from getting to the local optima too quickly. Now, after the $g$th IWD moves from node $u$ to node $v$ via edge $e_{u,v}(P)$, the amount of soil on the edge can be reduced by

$$soil(e_{u,v}(P))=\begin{cases}(1-\rho_L)soil(e_{u,v}(P))-\rho_L\Delta soil_{\min}, \\ \quad \text{if } \Delta soil(e_{u,v}(P))<\Delta soil_{\min}; \\ (1-\rho_L)soil(e_{u,v}(P))-\rho_L\Delta soil_{\max}, \\ \quad \text{if } \Delta soil(u,v)>\Delta soil_{\max}; \\ (1-\rho_L)soil(e_{u,v}(P))-\rho_L\Delta soil(e_{u,v}(P)), \\ \quad \text{otherwise,}\end{cases} \tag{20}$$

where $\rho_L$ is the local soil update parameter.

The $g$th IWD that have moved from node $u$ to node $v$, increases the soil $soil^{IWD_g}$ it carries by

$$soil^{IWD_g}=\begin{cases}soil^{IWD_g}+\Delta soil_{\min}, \\ \quad \text{if } \Delta soil(u,v)<\Delta soil_{\min}; \\ soil^{IWD_g}+\Delta soil_{\max}, \\ \quad \text{if } \Delta soil(u,v)>\Delta soil_{\max}; \\ soil^{IWD_g}+\Delta soil(u,v), \text{ otherwise.}\end{cases} \tag{21}$$

Therefore, the movement of an IWD between two nodes reduces the soil on the path between the two nodes and increases the soil of the moving IWD.

B.6 Executing Mutation-Based Local Search

After all the IWDs discover their feasible solutions, these solutions need to undergo the mutation-based local searches. In this process, an edge $e_{u,v}(P)$ is randomly selected from the edges of a solution and is replaced by another edge connecting node $u$ to node $v$ if this edge replacement improves the fitness of the solution, namely, the utility. Otherwise, the previous solution remains the same. This process has to be done for several times for each solution. After completing the mutation-based local search, a solution (i.e., values of powers allocated for the selected subcarriers) can be obtained by the following steps.

- Convert every $W$ bits into a decimal number using the binary coding (gray coding may be better). Let $y$ denote the decimal number.
- Calculate the power of the $k$th subcarrier, $P_k$, by

$$P_k=P_k^{\min}+\frac{\left(P_k^{\max}-P_k^{\min}\right)*y}{2^W-1}$$

- For all the $S$ selected subcarriers, $S$ power values are calculated to form the power vector $P=(P_1,P_2,\cdots,P_k,\cdots,P_S)^T$.

B.7 Finding the Iteration-Best Solution $T^{IB}$

For the power allocation problem, a quality function is needed to measure the fitness of solutions. In this paper, we choose the utility function $U(\cdot)$ as the quality function, and the quality of a solution $T^{IWD_g}$ founded by the $g$th IWD is given by $U(T^{IWD_g})$. As we know, once an iteration of the PA-MIWD algorithm is completed, all the IWDs have constructed their solutions, $T^{IWD}$. Then the best solution $T^{IB}$ of the iteration found by the IWDs is obtained by

$$T^{IB}=\arg\max_{\forall T^{IWD_g}} U(T^{IWD_g}). \tag{22}$$

Therefore, the iteration-best solution $T^{IB}$ is the solution that has the highest utility over all the solutions $T^{IWD}$ found by the IWDs.

B.8 Updating the Soil on the Paths with the Iteration-Best Solution $T^{IB}$

After identifying the iteration-best solution $T^{IB}$ from all the solutions $T^{IWD}$ found by the IWDs, the amount of soil, $soil(e_{u,v}(P))$, on edge $e_{u,v}(P)$ with the iteration-best solution $T^{IB}$ is updated by

$$soil(e_{u,v}(P))=(1+\rho_{IWD_g})soil(e_{u,v}(P))-\rho_{IWD_g}\frac{1}{W-1}soil_{IB}^{IWD_g}, \tag{23}$$

where $\rho_{IWD_g}$ is the global soil updating parameter which is chosen from $[0,1]$, and $soil_{IB}^{IWD_g}$ represents the soil of the iteration-best IWD. The iteration-best IWD is the IWD that has constructed the iteration-best solution $T^{IB}$ in the current iteration. The first term on the right-hand side of (23) is the amount of soil that remains from the previous iteration. Meanwhile, the second term of the right-hand side of (23) represents the quality of the current solution obtained by the IWD. This way of updating soil assists the reinforcement of the best-iteration solutions gradually, and thus, the IWDs are guided to search near good solutions with the expectation of finding the global optimum.

B.9 Updating the Global Best Solution

At the end of each iteration of the algorithm, the total best solution $T^{TB}$ is updated by the current iteration-best solution $T^{IB}$ as follows

$$T^{TB}=\begin{cases}T^{TB}, \text{if } U(T^{TB})>U(T^{IB}); \\ T^{IB}, \text{otherwise.}\end{cases} \tag{24}$$

By doing this, it is guaranteed that $T^{TB}$ holds the best solution obtained so far by the PA-MIWD algorithm.

### *C. Convergence Properties of PA-MIWD Algorithm*

In this subsection, we show that the PA-MIWD algorithm is able to find the optimal solution at least once during its lifetime if the number of iterations, $Z$, is sufficiently large. In the following, we study the convergence property of the PA-MIWD algorithm.

It is known that any solution of the PA-MIWD algorithm is composed of a number of nodes selected by an IWD in an iteration of the algorithm. As a result, as long as the probability of selecting any node in graph $G$ of the power allocation problem is great than zero, the chance for any feasible solution to be found by an IWD from all solutions of the problem is nonzero in an iteration of the algorithm and it is guaranteed that the optimal solution can be found. Because once an IWD finds an optimal solution, that solution becomes the iteration-best solution in the algorithm and thus the total-best solution is updated to the newly found optimal solution by (24). As a result, the convergence for the PA-MIWD algorithm can be proven if the probability of choosing any node in graph $G$ in a solution is nonzero.

In the soil updating of the algorithm, two extreme cases are considered:

**Case 1:** Only those terms of the PA-MIWD algorithm, which increase soil to an edge of graph $G$, are considered.

**Case 2:** Only those terms of the PA-MIWD algorithm, which decrease soil to an edge of graph $G$ are considered.

For each case, the worst-case is followed. Let $k$ denote the number of iterations that the algorithm has been repeated so far. For Case 1, the largest possible value of soil that an edge can hold after $k$ iterations, $soil(edge_{\max})$, is [11]

$$soil(edge_{\max}) = \left((1-\rho_L)(1+\rho_{IWD_g})\right)^k S_0, \quad (25)$$

where $edge_{\max}$ denotes the edge with the largest possible value of soil, $S_0$ is the initial soil of edge $edge_{\max}$ and $\rho_L$ is used in (20) and $\rho_{IWD_g}$ is used in (23).

For Case 2, the lowest possible value of soil for an edge is computed. That edge is denoted by $edge_{\min}$. Then, after $k$ iterations, the soil of $edge_{\min}$ is [27]

$$soil(edge_{\min}) = k(\rho_{IWD_g} - \rho_L N_{IWD})\frac{a_s}{b_s}, \quad (26)$$

where soil updating parameters $a_s$ and $b_s$ are defined in (19).

Based on (25) and (26), the following lemma holds.

**Lemma 1:** The soil of any edge in graph $G$ of the power allocation problem after $k$ iterations of the PA-MIWD algorithm remains in the interval $[soil(edge_{\min}), soil(edge_{\max})]$.

Consider the algorithm is at the stage of choosing next node $v$ for an IWD in node $u$. The value $g(soil(e_{u,v}(P)))$ of edge $e_{u,v}(P)$ is calculated by (14), which positively shifts $soil(e_{u,v}(P))$ by the amount of the lowest negative soil value of any edge, $\underset{l=0,1,j'\notin vc(IWD)}{min}(soil(e_{u,v'}(l)))$, as explained earlier. To consider the worst case, we let the lowest negative value be $soil(edge_{\min})$ and the $soil(e_{u,v}(P))$ be equal to $soil(edge_{\max})$. As a result, we have

$$g(soil(e_{u,v}(P))) = soil(edge_{\max}) - soil(edge_{\min}) \quad (27)$$

with the assumption that $soil(edge_{\min})$ is negative, which is the worst case.

Furthermore, $f(soil(e_{u,v}(P)))$ needs to be computed by (13), which yields:

$$f(soil(e_{u,v}(P))) = \frac{1}{\varepsilon_s + (soil(edge_{\max}) - soil(edge_{\min}))}. \quad (28)$$

Then the denominator of formula (12) has its largest possible value when it is assumed that each $soil(e_{u,v'}(l))$ in (12) is zero. Consequently, the probability, $p_{u,v}^{IWD_g}(k)$, of the $IWD_g$ going from node $u$ to node $v$ via edge $e_{u,v}(P)$ will be larger than $p_{lowest}$ such that

$$p_{u,v}^{IWD_g}(k) > p_{lowest}, \quad (29)$$

where $p_{lowest} = \frac{\varepsilon_s \cdot (1/c_{u,v}(k))}{(N_c-1)(\varepsilon_s + (soil(edge_{\max}) - soil(edge_{\min})))} > 0$ and $N_c = S * W$.

The probability of finding any feasible solution by an IWD in iteration $k$ is $(p_{lowest})^{(N_c-1)}$. Since there are $N_{IWD}$ IWDs, the probability that $p(s,k)$ finds any feasible solution $s$ by the IWD in iteration $k$ is

$$p(s,k) = N_{IWD}\,(p_{lowest})^{(N_c-1)}. \quad (30)$$

The probability of finding any feasible solution $s$ at the end of $Z$ iterations of the PA-MIWD algorithm is

$$Prob(s,Z) = 1 - \prod_{k=1}^{Z}(1-p(s,k)). \quad (31)$$

Since $0 < p(s,k) \le 1$, by making $Z$ sufficiently large, it is concluded that

$$\lim_{Z\to\infty}\prod_{k=1}^{Z}(1-p(s,k)) = 0. \quad (32)$$

Therefore, we have $\lim_{Z\to\infty} Prob(s,Z) = 1$. This fact indicates that any solution $s$ of the power allocation problem can be found at least once by at least one IWD of the algorithm if the number of iterations of the algorithm, $Z$, is sufficiently large. From the above analysis, we can conclude the following theorem.

**Theorem 3:** Let $Prob(s,Z)$ represent the probability of finding any feasible solution $s$ within $Z$ iterations of the PA-MIWD algorithm. As $Z$ gets larger, $Prob(s,Z)$ approaches to one, i.e.,

$$\lim_{Z\to\infty} Prob(s,Z) = 1$$

Based on the fact that the optimal solution $s^*$ is a feasible solution of the power allocation problem, from the above result, the following theorem holds.

**Theorem 4:** The PA-MIWD algorithm finds the optimal solution $s^*$ of the power allocation problem with probability one if the number of iterations $Z$ is sufficiently large.

It is worth noting that the required number of iterations $Z$ to find the optimal solution $S^*$ should be decreased by carefully adjusting parameters of the PA-MIWD algorithm for the power allocation problem. Finally, it is not difficult to deduce that the time complexity of the PA-MIWD algorithm is $O(N_{IWD} * K * W * M))$.

## V. PERFORMANCE EVALUATION

In this section, we evaluate the performance of the proposed algorithms by conducting simulations on the ns-2 simulator.

We assume that the MS users are randomly distributed in the cell with 0.5 km radius. There are a total of 20 MS users, 4 RS nodes, 3 BS nodes and 64 subcarriers over 3.2 MHz band. The bandwidth of every subcarrier is 50 kb/s within the coherence bandwidth. The channel coefficients for the subcarriers and for different links, $h^{BM}$, $h^{BR}$ and $h^{RM}$, are generated from independent and identically distributed (i.i.d.) Rician fading channel with $K$-factor = 1. And the variance AWGN is assumed to be 1, i.e., $\sigma^2_{BM,ijk}$, $\sigma^2_{BR,irk}$ and $\sigma^2_{RM,ijm}$ equal 1. We let the lower bound $P_{k,m}^{\min}$ and the upper bound $P_{k,m}^{\max}$ of each subcarrier power $P_{k,m}^{ij}$ be 0.5 mW and 5 mW respectively, and weight $w_k = 2$. Furthermore, unless otherwise specified, the minimum subcarrier requirement $\rho_j^{\min}$ is set to 1 as well as the maximum subcarrier requirement $\rho_j^{\max}$ is set to 6 for each node $n$. The parameters used by the PA-MIWD algorithm are listed in Table 2. The initial soil amount of each edge and the initial velocity of each IWD are random numbers uniformly distributed in $[1, 10]$. Without loss of generality, we let $W = 32$.

### A. Convergence and Total Utility Comparison

In this subsection, we study the convergence property of the PA-MIWD algorithm proposed in Section IV and compare the total utility among JS2PA, ESPA [18], and JOSPA [8]. The efficient subcarrier, power and rate allocation (ESPA) algorithm

Table 2. Parameters for PA-MIWD algorithm.

| Parameter | Value | Parameter | Value | Parameter | Value |
|---|---|---|---|---|---|
| $N_{IWD}$ | 20 | $Iter_{\max}$ | 100 | $\Delta soil_{\min}$ | 10 |
| $a_v$ | 1 | $b_v$ | 0.01 | $c_v$ | 1 |
| $a_s$ | 1 | $b_s$ | 0.01 | $c_s$ | 1 |
| $\varepsilon_s$ | 0.01 | $\varepsilon$ | 0.001 | $U(T^{TB})$ | -∞ |
| $\Delta soil_{\max}$ | 100 | $\rho_L$ | 0.9 | $\rho_{IWD_g}$ | 0.9 |

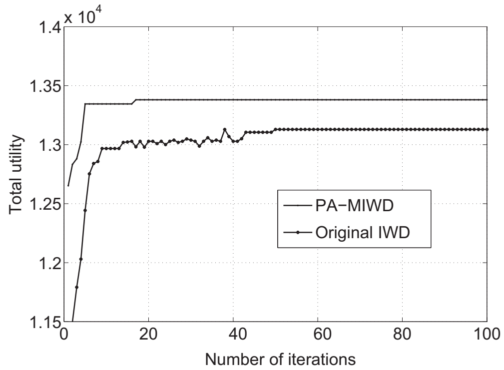



Fig. 4. Convergence comparison between PA-MIWD algorithm and the original IWD algorithm.

with fairness consideration for OFDMA uplink in [18] aims to maximize the sum rate under individual rate and transmit power constraints, where the complexity of the proposed subcarrier allocation algorithm is $O(K * M)$ and the complexity of power allocation algorithm is the same as that of water filling power allocation algorithm. The joint optimal subcarrier and power allocation (JOSPA) scheme in [8] aims to achieve subcarrier assignment and power control for each relay node and to minimize a cost function of average relay power for multiuser wireless OFDM cooperative networks, where the convexity is $O(K * N * M)$. However, both ESPA, and JOSPA did not consider subcarrier pairing, which is the main reason that ESPA and JOSPA have lower complexity than our JS2PA.

We first verify the convergence of the proposed PA-MIWD algorithm compared with that of the original IWD algorithm. Fig. 4 shows the convergence of the network utility versus iterations by using the PA-MIWD algorithm and the original IWD algorithm for 25 trails. It is observed from Fig. 4 that the PA-MIWD algorithm takes 25 iterations for convergence whereas the original IWD approach takes almost 50 iterations and the PA-MIWD algorithm provides more network utility. The underlying reason for such superiority of the PA-MIWD algorithm is that the probability that an IWD chooses next node to visit not only depends on the amount of soil on the edges but also the processing cost of the candidate edges.

Fig. 5 plots the convergence and the total utilities achieved by JS2PA, ESPA and JOSPA when the number of subcarriers is 5. We can observe that both JS2PA and JOSPA have better convergence, whereas ESPA fluctuates heavily. In the meanwhile, the proposed JS2PA algorithm can obtain higher total network utility compared with other two algorithms. The underlying reason is that instead of only considering subcarrier allocation, we take into account of the joint subcarrier pairing and selection aiming at maximizing the overall channel power gains and the network

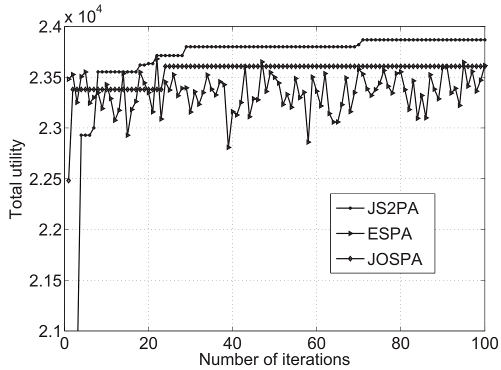


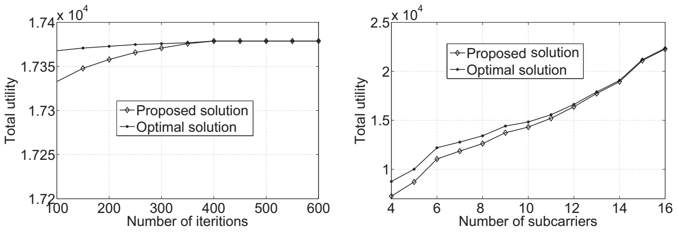

Fig. 5. Convergence and total utility comparison among JS2PA, ESPA, and JOSPA, when the number of subcarriers is 5.

utility. Moreover, we apply the proposed PA-MIWD algorithm to achieve the optimal power allocation for each selected subcarrier. However, JOSPA [8] solves the nonconvex subcarrier and power allocation problem by relaxing the discrete subcarrier allocation policy $T \in \{0, 1\}^{K\times K}$ to a continuous real number $T \in [0, 1]^{K\times K}$ and adopts the "winner-takes-all" policy in subcarrier allocation instead of considering the fairness of subcarrier allocation. Although ESPA [18] considers the fairness of subcarrier allocation, it does not employ the joint subcarrier pairing and selection.

Fig. 6 plots the comparison of the proposed solution and the optimal one to OPT-3. In the computation of the optimal solution, we first give the optimal subcarrier pairing and selection for 64 subcarriers by Algorithm 1 in section III. We then adopt the brute force search method to find the optimal transmission power from the lower bound $P^{\min}_{k,m} = 0.5$ mW and the upper bound $P^{\max}_{k,m} = 5$ mW of each subcarrier power $P^{ij}_{k,m}$, where the stepsize of searching is $1.0 * 10^{-5}$, such that the objection function in OPT-3 reaches maximum. Clearly, the computation cost is very high. We can observe from Fig. 6(a) that the maximum relative error between them is 0.21% when the number of iterations is 100 and the obtained total utility by the proposed PA-MIWD algorithm is gradually close to the optimal solution to OPT-3 as the number of iterations increases. Both are equal when the number of iterations reaches about 350, which verifies the correctness of Theorem 4. Another observation from Fig. 6(a) is that the maximum relative error between the proposed solution by the JS2PA algorithm and the optimal one to OPT-3 is 10.1% when the number of subcarriers is 4 and the difference between both gradually decreases as the number of subcarriers increases. Both are equal when the number of subcarriers reaches 12. The latter shows that the proposed solution gradually approaches to the optimal solution as the number of subcarriers become large.

### B. Effect of Relay Location on System Performance

To exploit the effect of relay location on system performance, we assume that the relay RS locates on a line between the source BS and the destination MS, and the BS-MS distance is one unit. Denote $d$ $(0 < d < 1)$ as BS-RS distance. Thus, the RS-MS distance is $1 - d$. It is clear that the larger $d$ is, the closer the re-




(a) (b)

Fig. 6. Comparison of the proposed solution and the optimal one to OPT-3: (a) Number of iterations when the number of subcarriers is 5 and (b) number of subcarriers.

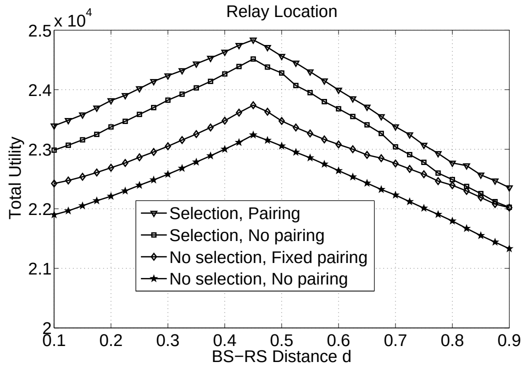



Fig. 7. Effect of relay node location on system performance.

lay is to the destination. The resulting utility for scenarios where subcarrier pairing and selection are activated and inactivated are shown in Fig. 7. Analyzing the results in Fig. 7, we observe that any scenario without considering subcarrier selection has lower total utility for all positions of the relay, which means that subcarrier selection has an evident effect on network utility. In the meanwhile, we find also from Fig. 7 that as the BS-RS distance $d$ increases, the total utility becomes larger and reaches maximum at about $d = 0.45$, then decreases gradually. These results show that subcarrier pairing is more effective than subcarrier selection when the relay is closer to the source, and subcarrier selection becomes dominant when the relay goes towards the destination. This is because that when the relay is close to the source, both BS-RS channel and RS-MS channel are better than the BS-MS channel and most of subcarriers are used for relaying regardless subcarrier selection is activated or not.

### *C. Effect of Subcarrier Requirement on Resource Allocation*

This subsection will verify the effect of the subcarrier requirement constraints on resource allocation.

Fig. 8 compares the total utilities and the number of starved users and starved subcarriers of JS2PA, ESPA, and JOSPA as the minimum and maximum numbers of subcarriers, $\rho_n^{\min}$ and $\rho_n^{\max}$. From Fig. 8(a)–(c), we can see that with the increase of the minimum number of subcarriers, the total utility of all three algorithms decreases. This is because as the minimum number of subcarriers increases, the users with poor channel conditions will have less opportunity to be allocated subcarriers and be paired with other users, which leads to the reduction of utility. However, compared to other two algorithms, JS2PA has higher utility. In the meanwhile, we can also observe that as the maximum number of subcarriers increases, the number of starved users and starved subcarriers will increase at the same time.

Fig. 8(d)–(f) show that as the maximum number of subcarriers increases, JS2PA still has higher utility than other two algorithms and the number of starved subcarriers will decrease while the number of starved users varies slightly. The reason is that the users with a good communicating environment will be allocated more subcarriers, and the users with a poor communicating environment have already own a minimum number of subcarriers, and may always fail to obtain its subcarriers in future.

### *D. Effect of Power Allocation on Total Utility*

In this subsection, we study the performance of the proposed PA-MIWD algorithm for the JS2PA problem with individual power constraints compared to other two algorithms, ESPA and JOSPA. Fig. 9 illustrates the total utility versus the number of subcarriers for the selective DF relaying mode. We consider the cases that subcarrier number $N$ is set to an integer from 4 to 16.

We can observe from Fig. 9 that the total utility increases with the number of subcarriers for all three algorithm, however, JS2PA always outperforms other two algorithms without subcarrier pairing. This is because more subcarriers can provide the PA-MIWD algorithm with more frequency diversity and more flexibility in pairing. Moreover, we introduce the utility (benefit) and average power requirement in the heuristic function $HUD_{PA}$ (16) to measure the desirability of the $g$th IWD to move from a node to its next node. We can also find from Fig. 9 that the larger the subcarrier number is, the bigger the performance gap among JS2PA, ESPA, and JOSPA.

## VI. CONCLUSIONS

In this paper, we have presented a JS2PA scheme for OFDMA cooperative relay networks. We first formulate the JS2PA problem as an optimization problem of network utility maximization under the constraints of transmission power, subcarrier requirement and pairing. Since the formulated maximization problem is a mixed integer programming problem, the traditional dual and subgradient method is not appropriate for solving this optimization. To solve the optimization problem, furthermore, we propose the subcarrier pairing and selection algorithm based on Hungarian method and the power allocation algorithm based on the IWD method, in which compared with the original IWD algorithm in [9], we improve the probability of choosing next visiting node, the desirability measure function for power allocation and the soil update strategy to accelerate convergence speed and avoid local optima. Simulations verify the convergence of the proposed JS2PA scheme and the effect of relay location, subcarrier requirement and power allocation on system performance. Numerical results show that the proposed JS2PA scheme outperforms other two existing algorithms in terms of convergence and total utility.

## APPENDICES

## I. PROOF OF THEOREM 1

Similar to [28], we introduce a binary indicator $\rho_{k,m} \in \{0, 1\}$ that is 1 if $SP(k, m)$ works in direct-link mode, and 0 if it works

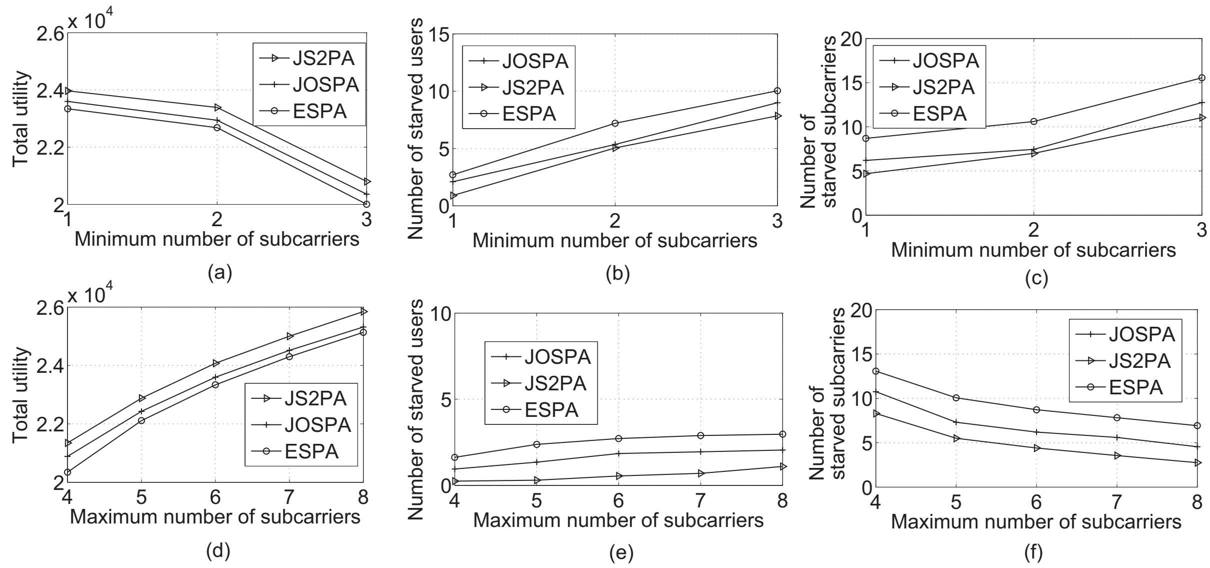



Fig. 8. Effect of subcarrier requirement on resource allocation for the algorithms of JS2PA, ESPA, and JOSPA.

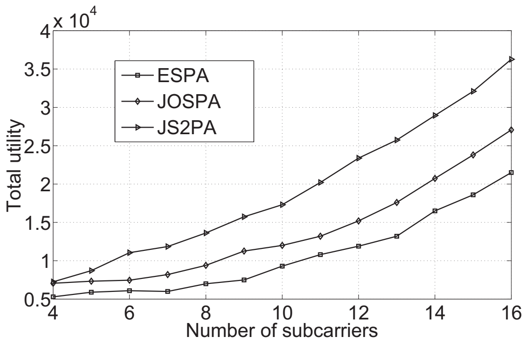



Fig. 9. Effect of power constraint on total utility for the algorithms of JS2PA, ESPA, and JOSPA.

in relay mode. Furthermore, by (3), we have $P_{ijk}^{BS} = \alpha_{ij} P_{rjm}^{RS}$, where $\alpha_{ij} = \frac{\lambda_{rjm}^{RM}}{\lambda_{irk}^{BR} - \lambda_{ijk}^{BM}}$. Hence the total amount of power $P_{k,m}^{ij}$ allocated to subcarrier pair $SP(k,m)$ will be given by $P_{k,m}^{ij} = P_{ijk}^{BS} + P_{rjm}^{RS} = (1+\alpha_{ij}) P_{rjm}^{RS} = P_{ijk}^{BS}(1+\alpha_{ij})/\alpha_{ij}$. Then (1) can be rewritten as

$$R_{k,m}^{ij} = \frac{w_k}{2}[\rho_{k,m} \log\left(1 + \lambda_{ijk}^{BM} P_{k,m}^{ij}\right) + (1-\rho_{k,m}) \log(1 + \lambda_{irk}^{BR} P_{k,m}^{ij} \alpha_{ij}/(1+\alpha_{ij}))]. \quad (33)$$

where for the direct-link mode, $P_{k,m}^{ij} = P_{ijk}^{BS}$ and $P_{rjm}^{RS} = 0$, while for the relay mode, $P_{k,m}^{ij} = P_{ijk}^{BS} + P_{rjm}^{RS}$ with $P_{ijk}^{BS} = \alpha_{ij} P_{rjm}^{RS}$.

The solution of mode selection can be given by taking the derivatives of $R_{k,m}^{ij}$ with respect to the indicators. We have

$$\frac{\partial R_{k,m}^{ij}}{\partial \rho_{k,m}} = \frac{w_k}{2} \log\left(\frac{1 + \lambda_{ijk}^{BM} P_{k,m}^{ij}}{1 + \lambda_{irk}^{BR} P_{k,m}^{ij} \alpha_{ij}/(1+\alpha_{ij})}\right), \quad (34)$$

$$\begin{cases} > 0, \rho_{k,m} = 1, \\ < 0, \rho_{k,m} = 0. \end{cases} \quad (35)$$

It is clear that when

$$\lambda_{ijk}^{BM} \leq \lambda_{irk}^{BR} \alpha_{ij}/(1+\alpha_{ij}), \quad (36)$$

i.e., $\frac{\partial R_{k,m}^{ij}}{\partial \rho_{k,m}} < 0$, we have $\rho_{k,m} = 0$ and $SP(k,m)$ should work in relay mode.

It is not difficult to obtain from (1) that when one has $\lambda_{ijk}^{BM} \geq \lambda_{irk}^{BR}$, due to the min operation, the rate obtained in direct-link mode will always be higher than that in relay mode. In the following, we will show that if $\lambda_{ijk}^{BM} \leq \lambda_{irk}^{BR}$, then the inequality between $\lambda_{ijk}^{BM}$ and $\lambda_{irk}^{BR}\alpha_{ij}/(1+\alpha_{ij})$ is equivalent to the inequality between $\lambda_{ijk}^{BM}$ and $\lambda_{rjm}^{RM}$. In fact, with the definition of $\alpha_{ij}$,

$$\frac{\alpha_{ij}}{1+\alpha_{ij}} \lambda_{irk}^{BR} = \frac{\lambda_{irk}^{BR} \lambda_{rjm}^{RM}}{\lambda_{irk}^{BR} - \lambda_{ijk}^{BM} + \lambda_{rjm}^{RM}}.$$

Then by (36) we have

$$\frac{\lambda_{irk}^{BR} \lambda_{rjm}^{RM}}{\lambda_{irk}^{BR} - \lambda_{ijk}^{BM} + \lambda_{rjm}^{RM}} \geq \lambda_{ijk}^{BM}.$$

Furthermore, we can obtain

$$\lambda_{irk}^{BR}(\lambda_{rjm}^{RM} - \lambda_{ijk}^{BM}) \geq \lambda_{ijk}^{BM}(\lambda_{rjm}^{RM} - \lambda_{ijk}^{BM}).$$

Clearly, the above analysis shows that

$$\lambda_{ijk}^{BM} \leq \lambda_{irk}^{BR} \alpha_{ij}/(1+\alpha_{ij}) \iff \lambda_{ijk}^{BM} \leq \lambda_{rjk}^{RM}. \quad (37)$$

This means that when $\lambda_{ijk}^{BM} \leq \lambda_{irk}^{BR}$, the selection of working in direct or relay mode may be based on either comparison in (37) because they are equivalent. In short, to work in relay mode, subcarrier pair $SP(k,m)$ should fulfil the following two conditions simultaneously: $\lambda_{irk}^{BR} \geq \lambda_{ijk}^{BM}$ and $\lambda_{rjk}^{RM} \geq \lambda_{ijk}^{BM}$.

## REFERENCES

[1] H. W. Kuhn, "The Hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, vol. 2, pp. 83–97, 1995.

[2] Y. Kim and J. Kim, "A 2-D subcarrier allocation scheme for capacity enhancement in a clustered OFDM system," *IEICE Trans. Commun.*, vol. E90–B, no. 7, pp. 1880–1883, July 2007.

[3] N. Y. Ermolova and B. Makarevitch, "Performance of practical subcarrier allocation schemes for OFDMA," *in Proc. IEEE PIMRC*, Sept. 2007.

[4] O. Nwamadi, X. Zhu, and A. Nandi, "Dynamic subcarrier allocation for single carrier – FDMA systems," *in Proc. EUSIPCO*, Aug. 2008.

[5] G. Bansal, M. Hossain, V. Bhargava, and T. Le-Ngoc, "Subcarrier and power allocation for OFDMA-based cognitive radio systems with joint overlay and underlay spectrum access mechanism," *IEEE Trans. Veh. Technol.*, vol. 62, no. 3, pp. 1111–1122, 2013.

[6] K. Xie, J. Cao, X. Wang, and J. Wen, "Optimal resource allocation for reliable and energy efficient cooperative communications," *IEEE Trans. Wireless Commun.*, vol. 12, no. 10, pp. 1293–1297, Oct. 2013.

[7] H. Boostanimehr and V. Bhargava, "Selective subcarrier pairing and power allocation for DF OFDM relay systems with perfect and partial CSI," *IEEE Trans. Wireless Commun.*, vol. 10, no. 12, pp. 4057–4067, 2011.

[8] D. Wang, Z. Li, and X. Wang, "Joint optimal subcarrier and power allocation for wireless cooperative networks over OFDM fading channels," *IEEE Trans. Veh. Technol.*, vol. 61, no. 1, pp. 249–257, 2012.

[9] H. Shah-Hosseini, "Problem solving by intelligent water drops," *IEEE Congr. Evol. Comput.*, 2007, pp. 3226–3231.

[10] H. Duan, S. Liu, X. Lei, and J. Wu, "Novel intelligent water drops optimization approach to single ucav smooth trajectory planning," *Aerospace Sci. Technol.*, vol. 13, no. 8, pp. 442–449, 2009.

[11] H. Shah-Hosseini, "Intelligent water drops algorithm: A new optimization method for solving the multiple knapsack problem," *International J. Intelligent Comput. Cybernetics*, vol. 1, no. 2, pp. 193–212, 2008.

[12] S. H. Niu , S. K. Ong, and A. Y. C. Nee, "An improved Intelligent Water Drops algorithm for achieving optimal job-shop scheduling solutions," *Int. J. Prod. Research*, vol. 50, no. 15, pp. 4192–4205, Aug. 2012.

[13] M. Islam and M. Rahman, "An improved intelligent water drop algorithm for a real-life waste collection problem," in *Lecture Notes in Computer Science: Advances in Swarm Intelligence*, Springer Berlin Heidelberg, 2013, vol. 7929, pp. 472–479.

[14] B. O. Alijla, L. P. Wong, C. P. Lim, A. T. Khader, and M. A. Al-Betar, "A modified Intelligent Water Drops algorithm and its application to optimization problems," *Expert Systems with Applications*, vol. 41, no. 15, pp. 6555–6569, Nov. 2014.

[15] B. O. Alijla, L. P. Wong, C. P. Lim, A. T. Khader, and M. A. Al-Betar, "An ensemble of intelligent water drop algorithms and its application to optimization problems," *Inform. Sci.*, vol. 325, pp. 175–189, Dec. 2015.

[16] H. Mokhtari, "A nature inspired intelligent water drops evolutionary algorithm for parallel processor scheduling with rejection," *Applied Soft Comput.*, vol. 26, pp. 166–179, Jan. 2015.

[17] C. N. Hsu, H. J. Su, and P. H. Lin, "Joint subcarrier pairing and power allocation for OFDM transmission with decode-and-forward relaying," *IEEE Trans. Signal Process.*, vol. 59, no. 1, pp. 399–414, 2011.

[18] L. Gao and S. Cui, "Efficient subcarrier, power, and rate allocation with fairness consideration for OFDMA uplink," *IEEE Trans. Wireless Commun.*, vol. 7, no. 5, May 2008, pp. 1507–1511.

[19] X. Li, Q. Zhang, G. Zhang, and J. Qin, "Joint power allocation and subcarrier pairing for cooperative OFDM AF multi-relay networks," *IEEE Commun. Lett.*, vol. 17, no. 5, pp. 872–875, May 2013.

[20] W. Ying, Q. Xin-chun, W. Tong, and L. Bao-ling, "Power allocation and subcarrier pairing algorithm for regenerative OFDM relay system," in *Proc. IEEE VTC*, Apr. 2007, pp. 2727–2731.

[21] J. W. Lee, M. Chiang, and A. R. Calderbank, "Jointly optimal congestion and contention control based on network utility maximization," *IEEE Commun. Lett.*, vol. 10, no. 3, pp. 216–218, 2006.

[22] C. N. Long *et al.*, "The end-to-end rate control in multiple-hop wireless networks: cross-layer formulation and optimal allocation," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 4, pp. 719–731, 2008.

[23] J. Papandriopoulos, S. Dey, and J. Evans, "Optimal and distributed protocols for cross-layer design of physical and transport layers in MANETs," *IEEE/ACM Trans. Netw.*, vol. 16, no. 6, pp. 1392–1405, 2008.

[24] W. Dang, M. Tao, H. Mu, and J. Huang, "Subcarrier-pair based resource allocation for cooperative multi-relay OFDM systems," *IEEE Trans. Wireless Commun.*, vol. 9, no. 5, pp. 1640–1649, May 2010.

[25] W. Yu and R. Lui, "Dual methods for nonconvex spectrum optimization of multicarrier systems," *IEEE Trans. Commun.*, vol. 54, no. 7, pp. 1310–1322, July 2006.

[26] Y. Li *et al.,* "Power allocation and subcarrier pairing in OFDM-based relaying networks," in *Proc. IEEE ICC*, May 2008, pp. 2602–2606.

[27] H. Shah-Hosseini, "Optimization with the nature-inspired intelligent water drops algorithm," *Evolutionary Computation* edited by W.P.D. Santos, InTech: Vienna, Austria, Oct. 2009, pp. 297–320.

[28] V. Luc, L. Jerome, O. Onur, and Z. Abdellatif, "Rate-optimized power allocation for DF-relayed OFDM transmission under sum and individual power constraints," *EURASIP J. Wireless Commun. Network.*, vol. 2009, pp. 6, Feb. 2009.



**Guiyan Liu** was born in Chongqing, China, in 1992. She received the B.S. degree in Telecommunications Engineering from Southwest University, Chongqing, China, in 2014. She is currently working toward the Master's degree in Signal and Information Processing, Southwest University. Her research interests include stream scheduling in data center networks and software defined networking.



**Songtao Guo** received the B.S., M.S., and Ph.D. degrees in Computer Software and Theory from Chongqing University, Chongqing, China, in 1999, 2003, and 2008, respectively. He was a Professor from 2011 to 2012 at Chongqing University. He is currently a Full Professor at Southwest University, China. He was a Senior Research Associate at the City University of Hong Kong from 2010 to 2011, and a Visiting Scholar at Stony Brook University, New York, from May 2011 to May 2012. His research interests include wireless sensor networks, wireless ad hoc networks and parallel and distributed computing. He has published more than 70 scientific papers in leading refereed journals and conferences. He has received many research grants as a principal investigator from the National Science Foundation of China and Chongqing and the Postdoctoral Science Foundation of China. He is a Member of the IEEE.



**Huan Zhao** received his B.S., M.S., and Ph.D degrees from the College of Chongqing University, China, in 2005, 2008, and 2016 respectively. His research interests include wireless sensor networks, energy harvesting, and resource management strategies of the next generation wireless networks.



**Fei Wang** was born in Shandong, China in 1983. She received her B.S. degree in Mathematics from Liaocheng University, Liaocheng, China, in 2005, her M.S. degree from Wuhan University, Wuhan, China, in 2007, and her Ph.D. degree in Computer Science of Chongqing Uni versity in 2012. She is currently a Lecturer of Southwest University, Chongqing, China. Her research interests include congestion control and cross-layer optimization and distributed algorithm design in wire less ad hoc networks.