Efficient Resource Allocation and Power Control for D2D Communication with RAN Sharing

You-Chiun Wang and Wei-Ting Chen
Department of Computer Science and Engineering
National Sun Yat-sen University, Kaohsiung, 804, Taiwan
Email: ycwang@cse.nsysu.edu.tw; arishere326@gmail.com

***Abstract*—Device-to-device (D2D) communication improves the performance and flexibility of mobile networks. It enables direct communication between two *user equipments (UEs)* without the relay via a *base station (BS)*. D2D pairs can share the spectrum resources given to cellular UEs, but doing so causes interference. Thus, how to manage resources and transmitted power for UEs is important. This paper proposes a *D2D resource allocation and power control with RAN sharing (DACS)* framework, where some *service providers (SPs)* each possess a part of the BS's resources. Each SP allots its resources to UEs and decides power to satisfy UEs' SINR demands while reducing interference. SPs can borrow resources from each other for serving more UEs. Then, DACS amplifies the power of senders adaptively to better signal quality. Simulation results reveal that DACS can achieve a high service rate, increase energy efficiency, and keep fairness among SPs.**

***Index Terms*—D2D communication, energy efficiency, fairness, power control, RAN sharing, resource allocation.**

I. Introduction

The rapid growth in the number of *user equipments (UEs)* has made spectrum resources scarce. A promising solution is *device-to-device (D2D) communication*. It implements direct communication between two UEs (referred to as a *D2D pair*) without requiring a *base station (BS)* to relay messages. This technique has benefits: 1) The spectral efficiency is improved, as resources are shared by D2D pairs and *cellular UEs* (*CUs*). 2) UEs in a D2D pair (called *D2D UEs*, or *DUs* for short) are close to each other, so a D2D sender can reduce its transmitted power (below, we call it *power*) to save energy. 3) Using D2D communication expands a BS's service coverage [1].

In D2D communication, both *resource allocation* and *power control* are critical issues [2]. For a BS, its spectrum resources are divided into *resource blocks (RBs)* for allocation [3]. The resource allocation problem asks how to assign the BS's RBs to CUs and D2D pairs. When a CU and D2D pairs share an RB, their senders impose interference. Thus, the power control issue asks how to calculate the power for senders to improve throughput of UEs while mitigating interference.

Considering the massive costs of telecommunications infrastructure, 3GPP proposes *radio access network (RAN)* sharing [4] to help *service providers (SPs)* reduce network deployment and operational expenditure. More concretely, multiple SPs are allowed to share spectrum resources and infrastructure (e.g., BSs). As a result, using RAN sharing can substantially extend services and coverage of a mobile network.

This paper considers D2D communication in a RAN sharing scenario and proposes a *D2D resource allocation and power control with RAN sharing (DACS)* framework. Each SP first allots its RBs to UEs and finds power to meet SINR demands of UEs while reducing interference. In the case that there are UEs not served, an SP can borrow resources from other SPs for serving them. Our RB lending mechanism takes account of both efficiency and fairness. Then, the power of some senders is moderately adjusted to increase UE throughput. Simulation results demonstrate that DACS can attain a high service rate, raise energy efficiency, and maintain fairness among SPs.

II. Related Work

The resource allocation problem for D2D communication is NP-hard [5]. Many studies convert it to a matching issue. The work [6] picks a D2D pair to share each CU's RB based on the Hungarian method. Zhou et al. [7] find pairs of CUs and DUs for sharing RBs to raise energy efficiency. The study [8] uses the Gale-Shapley algorithm to assign DUs to reuse RBs given to CUs. However, each RB can be shared by no more than one D2D pair, resulting in low RB utilization.

How to allot resources and power to UEs via vertex coloring is discussed in [9], [10], [11]. They build a graph to express the interference relationship between UEs. Adjacent vertices (i.e., UEs) cannot share RBs. The work [12] treats UEs as players and uses a Stackelberg game for RB allocation. Raghu et al. [13] propose an iterative-based method to manage resources and power. The work [14] applies deep reinforcement learning to D2D resource allocation.

Some studies take RAN sharing into account. The work [15] allots RBs to CUs of different SPs and lets DUs reuse RBs allocated to CUs with the least interference. With binary integer programming, the study [16] formulates the RB allocation problem that considers each SP's cost. In [17], SPs can trade RBs based on channel conditions. The study [18] assigns RBs to UEs by referring to their types to meet QoS requirements. However, these studies assume fixed power for the BS and DU senders (i.e., no power control). The work [19] adjusts power for senders and allows SPs borrowing RBs from each other to increase throughput. However, some SPs may borrow more RBs from others, causing unfairness on SPs' throughput.

III. System Model

Consider a BS offering a set $\mathcal{R}$ of downlink RBs. There is also a set $\mathcal{S}$ of SPs that cooperate on the BS. Each SP $s_k \in \mathcal{S}$ possesses a subset $\mathcal{R}_k \subset \mathcal{R}$ of RBs with conditions as follows:

all RBs in $\mathcal{R}$ are assigned to SPs in $\mathcal{S}$ (i.e., $\bigcup_{s_k \in \mathcal{S}} \mathcal{R}_k = \mathcal{R}$), each SP holds non-zero RBs (i.e., $|\mathcal{R}_k| > 0, \forall s_k \in \mathcal{S}$), and no two SPs have overlapped RBs (i.e., $\mathcal{R}_k \cap \mathcal{R}_m = \emptyset$, $s_k \neq s_m$). SPs may borrow RBs from each other. This can be done via a *capacity broker*, as proposed by 3GPP, which takes charge of managing requests and leases for resources between SPs [20].

Cellular and D2D communication share the same band, and D2D pairs can reuse RBs assigned to CUs. Each D2D pair has a sender and a receiver (the latter is used to represent the pair). Each DU belongs to only one D2D pair. Let us denote by $\mathcal{C}_k$ and $\mathcal{D}_k$ the sets of CUs and D2D receivers subscribed to SP s_k, where $\mathcal{C}_k \cap \mathcal{D}_k = \emptyset$. Moreover, $\mathcal{C}$ and $\mathcal{D}$ signify all CUs and D2D receivers, that is, $\mathcal{C} = \bigcup_{\forall s_k \in \mathcal{S}} \mathcal{C}_k$ and $\mathcal{D} = \bigcup_{\forall s_k \in \mathcal{S}} \mathcal{D}_k$.

RB allocation needs to adhere to five rules. First, two CUs cannot use the same RB. Second, D2D pairs may reuse a CU's RBs. Third, each SP s_k shall first allot RBs in $\mathcal{R}_k$ to its UEs. Fourth, after s_k allots RBs by the 3rd rule, s_k can lend other SPs RBs in $\mathcal{R}_k$. Finally, if a D2D pair's members belong to different SPs, the receiver's SP handles RB allocation.

Suppose that UE u_i gets data from its sender $\varepsilon(i)$ on RB r_x. The strength of $\varepsilon(i)$'s signal acquired by u_i is $g_{\varepsilon(i),i} \times p^x_{\varepsilon(i),i}$. Here, $g_{\varepsilon(i),i}$ and $p^x_{\varepsilon(i),i}$ denote $\varepsilon(i)$'s channel gain (hereinafter called *gain*) and power on r_x to send data to u_i. When UEs u_i and u_j share r_x, u_j's sender $\varepsilon(j)$ imposes an amount $\psi^x_{\varepsilon(j),i}$ of interference on u_i. Then, we can estimate u_i's SINR on r_x as follows:

$$\sigma^x_i = \frac{z^x_i (g_{\mathtt{BS},i} \times p^x_{\mathtt{BS},i})}{\sum_{u_j \in \mathcal{D}} z^x_j \psi^x_{\varepsilon(j),i} + \varphi} \quad \text{if } u_i \in \mathcal{C} \text{ (i.e., a CU)}, \tag{1}$$

$$\sigma^x_i = \frac{z^x_i (g_{\varepsilon(i),i} \times p^x_{\varepsilon(i),i})}{\sum_{u_j \in \mathcal{C}} z^x_j \psi^x_{\mathtt{BS},i} + \sum_{u_{j'} \in \mathcal{D} \setminus \{u_i\}} z^x_{j'} \psi^x_{\varepsilon(j'),i} + \varphi} \quad \text{otherwise (i.e., a D2D receiver)}, \tag{2}$$

where $z^x_j = 1$ if u_j uses r_x, or $z^x_j = 0$ otherwise. In addition, φ indicates the thermal noise.

Let v^x_i be the indicator to show whether the channel quality of UE u_i on RB r_x fulfills its minimal required SINR $\sigma^{\mathtt{min}}_i$; in particular, $v^x_i = 1$ if $\sigma^x_i \geq \sigma^{\mathtt{min}}_i$, or $v^x_i = 0$ otherwise. Then, our problem can be expressed by

$$\text{maximize} \sum\nolimits_{r_x \in \mathcal{R}} \sum\nolimits_{s_k \in \mathcal{S}} \sum\nolimits_{u_i \in \mathcal{C}_k \cup \mathcal{D}_k} z^x_i v^x_i, \tag{3}$$

subject to constraints as follows:

$$\sum\nolimits_{u_i \in \mathcal{C}} z^x_i \leq 1, \sum\nolimits_{u_i \in \mathcal{C} \cup \mathcal{D}} z^x_i \leq \delta, \qquad \forall r_x \in \mathcal{R}, \tag{4}$$

$$z^x_i \in \{0,1\}, v^x_i \in \{0,1\}, \quad \forall r_x \in \mathcal{R}, \forall u_i \in \mathcal{C} \cup \mathcal{D}, \tag{5}$$

$$p^{\mathtt{min}}_i \leq p^x_{\varepsilon(i),i} \leq p^{\mathtt{max}}_i, \qquad \forall r_x \in \mathcal{R}, \forall u_i \in \mathcal{C} \cup \mathcal{D}. \tag{6}$$

Here, Eq. (3) is to maximize CUs and D2D pairs served, where a UE u_i is *served* if it obtains RBs that satisfy SINR demand $\sigma^{\min}_i$. Then, Eq. (4) indicates that two CUs cannot employ the same RB and gives the maximum number of UEs in $\mathcal{C} \cup \mathcal{D}$ allowed to share each RB ($\delta \in \mathbb{Z}^+$). In Eq. (5), z^x_i and v^x_i are indicators whose values are 0 or 1. Eq. (6) imposes lower and upper bounds (i.e., $p^{\mathtt{min}}_i$ and $p^{\mathtt{max}}_i$) on the power of u_i's sender.

Algorithm 1: Basic Resource Allocation (BRA)

1 **foreach** $u_i \in \mathcal{C} \cup \mathcal{D}$ **do**
2   **foreach** $u_j \in (\mathcal{C} \cup \mathcal{D}) \setminus \{u_i\}$ **do**
3   **if** $g_{\varepsilon(i),j} > g_{\mathtt{th}}$ **then**
4       $\zeta_{i,j} \leftarrow 1$;
5 **foreach** $s_k \in \mathcal{S}$ **do**
6   $\mathcal{C}'_k \leftarrow f^{\mathtt{DEC}}_{\mathtt{SRT}}(\mathcal{C}_k, g_{\mathtt{BS},i})$, $\mathcal{D}'_k \leftarrow f^{\mathtt{DEC}}_{\mathtt{SRT}}(\mathcal{D}_k, g_{\mathtt{BS},i})$;
7   **foreach** $u_i \in \mathcal{C}'_k$ **do**
8   **foreach** $r_x \in \mathcal{R}_k$ **do**
9   **if** $\mathcal{Q}_x \cap \mathcal{C} = \emptyset$ *and* $\zeta_{i,j} = 0,\ \forall u_j \in \mathcal{Q}_x$ **then**
10   Move u_i from $\mathcal{C}'_k$ to $\mathcal{Q}_x$ and break;
11   **foreach** $u_i \in \mathcal{D}'_k$ **do**
12   **foreach** $r_x \in \mathcal{R}_k$ **do**
13   **if** $|\mathcal{Q}_x| < \delta$ *and* $\zeta_{i,j} = 0,\ \forall u_j \in \mathcal{Q}_x$ **then**
14   Move u_i from $\mathcal{D}'_k$ to $\mathcal{Q}_x$ and break;
15   **foreach** $r_x \in \mathcal{R}_k$ **do**
16     $\Theta \leftarrow \mathtt{power\_control}(\mathcal{Q}_x)$;
17   Move DUs in Θ from $\mathcal{Q}_x$ to $\mathcal{D}'_k$;
18   **foreach** $u_i \in \mathcal{Q}_x \cap \mathcal{D}$ **do**
19   **if** $\sigma^x_i < \sigma^{\mathtt{min}}_i$ **then**
20   Move u_i from $\mathcal{Q}_x$ to $\mathcal{D}'_k$;

IV. The Proposed DACS Framework

DACS contains three core algorithms. In the *basic resource allocation (BRA)* algorithm, each SP allots its RBs to its UEs. If there are some UEs not served, the *inter-SP resource sharing (IRS)* algorithm enables resource lending among SPs. The *RB throughput improvement (RTI)* algorithm increases the number of data bits carried by RBs. Moreover, there are three auxiliary procedures. The *power control* procedure adjusts the power of each sender on the same RB. Both *SP prioritization* and *lender selection* procedures implement the RB lending mechanism.

A. Core Algorithms

Given each SP s_k's UEs, the BRA algorithm decides their initial RB and power allocation. Algo. 1 shows its pseudocode. Lines 1–4 check if UEs u_i and u_j are neighbors, that is, u_i's sender, $\varepsilon(i)$, imposes non-neglected interference on u_j. To do so, we check if gain $g_{\varepsilon(i),j}$ exceeds threshold $g_{\mathtt{th}}$. In line 4, indicator $\zeta_{i,j}$ reveals whether u_i and u_j are neighbors ($\zeta_{i,j} = 1$ if so, or $\zeta_{i,j} = 0$ otherwise; the default value of $\zeta_{i,j}$ is zero).

The subsequent for-loop allots RBs to s_k's CUs (i.e., lines 7–10) and DUs (i.e., lines 11–14) and finds their power (i.e., lines 15–20). BRA serves CUs with larger gains first, as their SINR demands can be met using lower power. So, line 6 sorts CUs in $\mathcal{C}_k$ by their gains from the BS decreasingly and stores the result in a set $\mathcal{C}'_k$. Let $\mathcal{Q}_x$ be the set of UEs using RB r_x. For each CU u_i in $\mathcal{C}'_k$, we choose an RB r_x from $\mathcal{R}_k$ for it if two conditions both hold: 1) r_x is not assigned to any CU,

Algorithm 2: Inter-SP Resource Sharing (IRS)

1 $\mathcal{S}_{\mathbf{B}} \leftarrow$ `SP_prioritization`();
2 **foreach** $s_k \in \mathcal{S}_{\mathbf{B}}$ **do**
3 $\quad \mathcal{S}_{\mathbf{L}} \leftarrow$ `lender_selection`(s_k);
4 $\quad \mathcal{C}'_k \leftarrow f^{\mathtt{DEC}}_{\mathtt{SRT}}(\mathcal{C}'_k, g_{\mathtt{BS},i})$, $\mathcal{D}'_k \leftarrow f^{\mathtt{DEC}}_{\mathtt{SRT}}(\mathcal{D}'_k, g_{\mathtt{BS},i})$;
5 $\quad$ **foreach** $u_i \in \mathcal{C}'_k$ **do**
6 $\quad\quad$ **foreach** $s_m \in \mathcal{S}_{\mathbf{L}}$ **do**
7 $\quad\quad\quad$ **foreach** $r_x \in \mathcal{R}_m$ **do**
8 $\quad\quad\quad\quad$ **if** $\mathcal{Q}_x \cap \mathcal{C} = \emptyset$ *and* $|\mathcal{Q}_x| < \delta$ *and* $\zeta_{i,j} = 0,\ \forall u_j \in \mathcal{Q}_x$ **then**
9 $\quad\quad\quad\quad\quad$ Move u_i from $\mathcal{C}'_k$ to $\mathcal{Q}_x$;
10 $\quad\quad\quad\quad\quad$ $\Theta \leftarrow$ `power_control`($\mathcal{Q}_x$);
11 $\quad\quad\quad\quad\quad$ **if** $\Theta \neq \emptyset$ *or* $\sigma^x_j < \sigma^{\mathtt{min}}_j, \exists u_j \in \mathcal{Q}_x$ **then**
12 $\quad\quad\quad\quad\quad\quad$ Move u_i from $\mathcal{Q}_x$ to $\mathcal{C}'_k$ and restore power of UEs in $\mathcal{Q}_x$;
13 $\quad\quad\quad\quad\quad$ **else**
14 $\quad\quad\quad\quad\quad\quad$ **go to** line 5;
15 $\quad$ **foreach** $u_i \in \mathcal{D}'_k$ **do**
16 $\quad\quad$ **foreach** $s_m \in \mathcal{S}_{\mathbf{L}}$ **do**
17 $\quad\quad\quad$ **foreach** $r_x \in \mathcal{R}_m$ **do**
18 $\quad\quad\quad\quad$ **if** $|\mathcal{Q}_x| < \delta$ *and* $\zeta_{i,j} = 0,\ \forall u_j \in \mathcal{Q}_x$ **then**
19 $\quad\quad\quad\quad\quad$ Move u_i from $\mathcal{D}'_k$ to $\mathcal{Q}_x$;
20 $\quad\quad\quad\quad\quad$ $\Theta \leftarrow$ `power_control`($\mathcal{Q}_x$);
21 $\quad\quad\quad\quad\quad$ **if** $\Theta \neq \emptyset$ *or* $\sigma^x_j < \sigma^{\mathtt{min}}_j, \exists u_j \in \mathcal{Q}_x$ **then**
22 $\quad\quad\quad\quad\quad\quad$ Move u_i from $\mathcal{Q}_x$ to $\mathcal{D}'_k$ and restore power of UEs in $\mathcal{Q}_x$;
23 $\quad\quad\quad\quad\quad$ **else**
24 $\quad\quad\quad\quad\quad\quad$ **go to** line 15;

and 2) there is no u_i's neighbor that uses r_x. If r_x is found, we remove u_i from $\mathcal{C}'_k$ and add it to $\mathcal{Q}_x$.

Then, s_k allots RBs to its DUs. BRA first serves DUs whose gains from the BS are larger, since they are more susceptible to the BS's interference. Line 6 sorts DUs in $\mathcal{D}_k$ by gain $g_{\mathtt{BS},i}$ in a decreasing order, and the result is stored in a set $\mathcal{D}'_k$. In lines 11–14, we pick an RB $r_x \in \mathcal{R}_k$ for each DU u_i in $\mathcal{D}'_k$ if two conditions both hold: 1) r_x can accommodate UEs (i.e., $|\mathcal{Q}_x| < \delta$) and 2) there is no u_i's neighbor also using r_x. After finding RB r_x, we remove u_i from $\mathcal{D}'_k$ and add it to $\mathcal{Q}_x$.

The remaining code computes power for UEs that use each RB $r_x \in \mathcal{R}_k$. To do so, we use the power control procedure in line 16, which takes $\mathcal{Q}_x$ as its input parameter. As discussed later in Section IV-B, this procedure returns a subset of DUs, Θ, from $\mathcal{Q}_x$ that cannot allocate power. Hence, line 17 removes these DUs from $\mathcal{Q}_x$ and adds them back to $\mathcal{D}'_k$. The for-loop in lines 18–20 further excludes each DU from $\mathcal{Q}_x$ whose SINR σ^x_i cannot meet its minimum requirement $\sigma^{\mathtt{min}}_i$.

Algorithm 3: RB Throughput Improvement (RTI)

1 **foreach** $r_x \in \mathcal{R}$ **do**
2 $\quad$ **foreach** $u_i \in \mathcal{Q}_x$ **do**
3 $\quad\quad$ $\tilde{g}^{\mathtt{sum}}_i \leftarrow \sum_{u_j \in \mathcal{Q}_x \setminus \{u_i\}} g_{\varepsilon(i),j}$;
4 $\quad$ $\mathcal{Q}_x \leftarrow f^{\mathtt{INC}}_{\mathtt{SRT}}(\mathcal{Q}_x, \tilde{g}^{\mathtt{sum}}_i)$;
5 $\quad$ **foreach** $u_i \in \mathcal{Q}_x$ **do**
6 $\quad\quad$ $\alpha \leftarrow 0$, $\beta \leftarrow (p^{\mathtt{max}}_i - p^x_{\varepsilon(i),i})/\alpha_{\mathtt{max}}$;
7 $\quad\quad$ **while** $\alpha < \alpha_{\mathtt{max}}$ *and* $\beta > \beta_{\mathtt{min}}$ **do**
8 $\quad\quad\quad$ $\Gamma_{\mathtt{pre}} \leftarrow \sum_{u_j \in \mathcal{Q}_x} \gamma^x_j$;
9 $\quad\quad\quad$ $p^x_{\varepsilon(i),i} \leftarrow p^x_{\varepsilon(i),i} + \beta$, $\Gamma_{\mathtt{alt}} \leftarrow \sum_{u_j \in \mathcal{Q}_x} \gamma^x_j$;
10 $\quad\quad\quad$ **if** $\Gamma_{\mathtt{alt}} \leq \Gamma_{\mathtt{pre}}$ *or* $\sigma^x_j < \sigma^{\mathtt{min}}_j, \exists u_j \in \mathcal{Q}_x$ **then**
11 $\quad\quad\quad\quad$ $p^x_{\varepsilon(i),i} \leftarrow p^x_{\varepsilon(i),i} - \beta$; break;
12 $\quad\quad\quad$ $\alpha \leftarrow \alpha + 1$;

When an SP s_k has not enough RBs to serve its UEs (i.e., $\mathcal{C}'_k \cup \mathcal{D}'_k \neq \emptyset$), the IRS algorithm helps s_k borrow RBs from other SPs to serve these UEs. Algo. 2 shows IRS's pseudocode. Line 1 invokes the SP prioritization procedure to get a sorted set $\mathcal{S}_{\mathbf{B}}$, which determines the order of SPs to borrow RBs from others. Based on the order, the subsequent for-loop iteratively picks an SP s_k from $\mathcal{S}_{\mathbf{B}}$. Then, line 3 calls the lender selection procedure to get a sorted set $\mathcal{S}_{\mathbf{L}}$ that decides the order of SPs to lend RBs to s_k, where $\mathcal{S}_{\mathbf{L}}$ does not include s_k.

Like BRA, line 4 sorts CUs in $\mathcal{C}'_k$ and DUs in $\mathcal{D}'_k$ by their gains from the BS decreasingly. Then, we first handle CUs in $\mathcal{C}'_k$ (in lines 5–14) and then DUs in $\mathcal{D}'_k$ (in lines 15–24). For each CU u_i in $\mathcal{C}'_k$, we iteratively pick an SP s_m from $\mathcal{S}_{\mathbf{L}}$ and then find an RB r_x from $\mathcal{R}_m$ (i.e., set of s_m's RBs). Then, r_x is a candidate RB used to serve u_i if three conditions all hold: 1) r_x is not assigned to any CU (i.e., $\mathcal{Q}_x \cap \mathcal{C} = \emptyset$), 2) r_x still has room for serving u_i (i.e., $|\mathcal{Q}_x| < \delta$), and 3) there are no u_i's neighbors also using r_x (i.e., $\zeta_{i,j} = 0,\ \forall u_j \in \mathcal{Q}_x$). If so, line 10 calls the power control procedure to recalculate power for UEs in $\mathcal{Q}_x$ (including u_i). However, if this procedure cannot allocate power to some UEs in $\mathcal{Q}_x$ (i.e., $\Theta \neq \emptyset$) or some UEs in $\mathcal{Q}_x$ cannot meet their SINR demands (i.e., $\sigma^x_j < \sigma^{\mathtt{min}}_j, \exists u_j \in \mathcal{Q}_x$), it is inappropriate to let u_i use r_x. In this case, we remove u_i from $\mathcal{Q}_x$, add u_i back to $\mathcal{C}'_k$, and restore the original power of all UEs in $\mathcal{Q}_x$, as shown in line 12. Otherwise, we go back to line 5 to pick the next UE in $\mathcal{C}'_k$ to serve.

The way to handle DUs of $\mathcal{D}'_k$ by lines 15–24 is similar to that to handle CUs, except that we check only two conditions, $|\mathcal{Q}_x| < \delta$ and $\zeta_{i,j} = 0,\ \forall u_j \in \mathcal{Q}_x$, in line 18.

For each RB $r_x \in \mathcal{R}$, the RTI algorithm in Algo. 3 raises power for those UEs in $\mathcal{Q}_x$ to improve r_x's throughput. Since raising power would increase interference with other UEs, we prioritize raising power for UEs with less impact on other UEs. Lines 2–4 contain the code, where $\tilde{g}^{\mathtt{sum}}_i$ is the sum of gains on all UEs (except for u_i) in $\mathcal{Q}_x$ by u_i's sender $\varepsilon(i)$. A lower $\tilde{g}^{\mathtt{sum}}_i$ value means that $\varepsilon(i)$ has less impact. Hence, line 4 uses

Procedure power_control($\mathcal{Q}_x$):

1 $\Theta \leftarrow \emptyset$;
2 **foreach** $u_i \in \mathcal{Q}_x$ **do**
3   Estimate initial power p_i for u_i's sender;
4   **if** $u_i \in \mathcal{D}$ *and* $p_i > p_i^{\mathtt{max}}$ **then**
5         $\Theta \leftarrow \Theta \cup \{u_i\}$;
6 $\mathcal{Q}'_x \leftarrow \mathcal{Q}_x \setminus \Theta$;
7 **foreach** $u_i \in \mathcal{Q}'_x$ **do**
8     $\tilde{g}_i^{\mathtt{sum}} \leftarrow \sum_{u_j \in \mathcal{Q}_x \setminus \{u_i\}} g_{\varepsilon(i),j}$;
9 $\mathcal{Q}'_x \leftarrow f_{\mathtt{SRT}}^{\mathtt{INC}}(\mathcal{Q}'_x, \tilde{g}_i^{\mathtt{sum}})$;
10 **foreach** $u_i \in \mathcal{Q}'_x$ **do**
11     $\Lambda_i^{\mathtt{LB}} \leftarrow \max\{p_i, p_i^{\mathtt{min}}\}$, $\Lambda_i^{\mathtt{UB}} \leftarrow p_i^{\mathtt{max}}$;
12   **do**
13         $\mathrm{P}_{\mathtt{L}} \leftarrow \Lambda_i^{\mathtt{LB}} + (\Lambda_i^{\mathtt{UB}} - \Lambda_i^{\mathtt{LB}}) \times 1/4$;
14         $\mathrm{P}_{\mathtt{M}} \leftarrow \Lambda_i^{\mathtt{LB}} + (\Lambda_i^{\mathtt{UB}} - \Lambda_i^{\mathtt{LB}}) \times 1/2$;
15         $\mathrm{P}_{\mathtt{H}} \leftarrow \Lambda_i^{\mathtt{LB}} + (\Lambda_i^{\mathtt{UB}} - \Lambda_i^{\mathtt{LB}}) \times 3/4$;
16   **if** $\sigma_i^x \geq \sigma_i^{\mathtt{min}}$ *and decrement in* $\sigma_i^x < 50\%$ *using* $\mathrm{P}_{\mathtt{L}}$ **then**
17             $p_{\varepsilon(i),i}^x \leftarrow \mathrm{P}_{\mathtt{L}}$, $\Lambda_i^{\mathtt{UB}} \leftarrow \Lambda_i^{\mathtt{UB}} \times 3/4$;
18   **else if** *increment in* $\sigma_i^x > 50\%$ *using* $\mathrm{P}_{\mathtt{H}}$ **then**
19             $p_{\varepsilon(i),i}^x \leftarrow \mathrm{P}_{\mathtt{H}}$, $\Lambda_i^{\mathtt{LB}} \leftarrow \Lambda_i^{\mathtt{LB}} \times 5/4$;
20   **else**
21             $p_{\varepsilon(i),i}^x \leftarrow \mathrm{P}_{\mathtt{M}}$;
22             $\Lambda_i^{\mathtt{UB}} \leftarrow \Lambda_i^{\mathtt{UB}} \times 3/4$, $\Lambda_i^{\mathtt{LB}} \leftarrow \Lambda_i^{\mathtt{LB}} \times 5/4$;
23   **while** $\mathrm{P}_{\mathtt{H}} - \mathrm{P}_{\mathtt{L}} \geq \rho$;
24 **return** Θ;

this value to sort UEs in $\mathcal{Q}_x$ increasingly.

For each UE $u_i \in \mathcal{Q}_x$, its sender can increase power by at most $p_i^{\mathtt{max}} - p_{\varepsilon(i),i}^x$, which we divide into $\alpha_{\mathtt{max}}$ equal parts (i.e., β) and $\alpha_{\mathtt{max}} > 1$. Line 7 presents two conditions for whether to run the while-loop: 1) $\varepsilon(i)$'s power is below its maximum value (i.e., $\alpha < \alpha_{\mathtt{max}}$) and 2) the adjustable power β is above threshold $\beta_{\mathtt{min}}$ (otherwise, it means that $\varepsilon(i)$'s power is close to $p_i^{\mathtt{max}}$, and there is no point to increase $\varepsilon(i)$'s power). Then, we use $\Gamma_{\mathtt{pre}}$ and $\Gamma_{\mathtt{alt}}$ to represent r_x's throughput before and after increasing $\varepsilon(i)$'s power by β, where γ_j^x is the number of data bits that a UE u_j can get from r_x. If raising $\varepsilon(i)$'s power has no benefit (i.e., $\Gamma_{\mathtt{alt}} \leq \Gamma_{\mathtt{pre}}$) or forces some UEs in $\mathcal{Q}_x$ to fail to meet their SINR demands (i.e., $\sigma_j^x < \sigma_j^{\mathtt{min}}, \exists u_j \in \mathcal{Q}_x$), we should restore the previous power value for $\varepsilon(i)$ and break the while-loop (as $\varepsilon(i)$'s power can no longer increase).

B. Auxiliary Procedures

Given a set $\mathcal{Q}_x$ of UEs that share RB r_x, the power control procedure computes the amount of power for the sender $\varepsilon(i)$ of each UE u_i in $\mathcal{Q}_x$ such that u_i's SINR demand (i.e., $\sigma_i^{\mathtt{min}}$) is met while $\varepsilon(i)$'s interference can be reduced. This procedure will return a subset Θ of DUs in $\mathcal{Q}_x$ whose senders cannot be allocated power. The for-loop in lines 2–5 estimates the initial power (denoted by p_i) for u_i's sender. There are two cases. If

Procedure SP_prioritization():

foreach $s_k \in \mathcal{S}$ **do**
    $\Omega_k^{\mathtt{sum}} \leftarrow 0$;
    foreach $r_x \in \mathcal{R}_k$ **do**
        foreach $s_m \in \mathcal{S} \setminus \{s_k\}$ **do**
            $\Omega_k^{\mathtt{sum}} \leftarrow \Omega_k^{\mathtt{sum}} + \Omega_{k,m}^x$;
foreach $s_k \in \mathcal{S}$ **do**
    $\Psi_k^{\mathtt{sum}} \leftarrow 0$;
    foreach $s_m \in \mathcal{S} \setminus \{s_k\}$ **do**
        foreach $r_x \in \mathcal{R}_m$ **do**
            $\Psi_k^{\mathtt{sum}} \leftarrow \Psi_k^{\mathtt{sum}} + \Omega_{m,k}^x$;
if *mode* $= 1$ **then**
    return $f_{\mathtt{SRT}}^{\mathtt{DEC}}(\mathcal{S}, \Omega_k^{\mathtt{sum}})$;
return $f_{\mathtt{SRT}}^{\mathtt{INC}}(\mathcal{S}, \Psi_m^{\mathtt{sum}})$;

u_i is a CU, we consider that it is interfered by thermal noise φ. Hence, p_i can be derived as follows: $(g_{\mathtt{BS},i} \times p_i)/\varphi \geq \sigma_i^{\mathtt{min}} \Rightarrow p_i = (\sigma_i^{\mathtt{min}} \times \varphi)/g_{\mathtt{BS},i}$. When u_i is a DU, we take account of the BS's interference:

$$\frac{g_{\varepsilon(i),i} \times p_i}{g_{\mathtt{BS},i} \times p_j^{\mathtt{min}} + \varphi} \geq \sigma_i^{\mathtt{min}} \Rightarrow p_i = \frac{\sigma_i^{\mathtt{min}}(g_{\mathtt{BS},i} \times p_j^{\mathtt{min}} + \varphi)}{g_{\varepsilon(i),i}}.$$

However, if $p_i > p_i^{\mathtt{max}}$, it means that $\varepsilon(i)$'s power cannot be allocated, so we add u_i to Θ. The code is given in lines 4–5.

Let $\mathcal{Q}'_x \subseteq \mathcal{Q}_x$ denote the set of UEs whose senders' power are allocatable. Since raising power may increase interference with other UEs, we prioritize raising power for UEs with less effect. To do so, the for-loop in lines 7–8 computes the sum of gains $\tilde{g}_i^{\mathtt{sum}}$ on all UEs (excluding u_i) in $\mathcal{Q}_x$ by u_i's sender $\varepsilon(i)$, and line 9 sorts $\mathcal{Q}'_x$ increasingly by the $\tilde{g}_i^{\mathtt{sum}}$ value.

The next for-loop decides power for the sender of each UE u_i in $\mathcal{Q}'_x$ based on a ternary search. Line 11 sets a power lower bound $\Lambda_i^{\mathtt{LB}}$ to the maximum of p_i (i.e., initial power) and $p_i^{\mathtt{min}}$ (i.e., minimal power) and a power upper bound $\Lambda_i^{\mathtt{UB}}$ to $p_i^{\mathtt{max}}$. In addition, we set three power values $\mathrm{P}_{\mathtt{L}}$, $\mathrm{P}_{\mathtt{M}}$, and $\mathrm{P}_{\mathtt{H}}$, located at $1/4$, $1/2$, and $3/4$ of the range from $\Lambda_i^{\mathtt{LB}}$ to $\Lambda_i^{\mathtt{HB}}$, in lines 13–15. Three cases are used to adjust power. In case 1 (i.e., lines 16–17), u_i meets its SINR demand using a lower power, and the decrement in σ_i^x is less than a half. Hence, it is safe to set $p_{\varepsilon(i),i}^x = \mathrm{P}_{\mathtt{L}}$, and we lower the power upper bound $\Lambda_i^{\mathtt{UB}}$. In case 2 (i.e., lines 18–19), using a high power improves u_i's SINR, so we set $p_{\varepsilon(i),i}^x$ to $\mathrm{P}_{\mathtt{H}}$ and raise the power lower bound $\Lambda_i^{\mathtt{LB}}$. Otherwise (i.e., case 3 in lines 20–22), we set $p_{\varepsilon(i),i}^x$ to $\mathrm{P}_{\mathtt{M}}$ (i.e., the medium power) and shrink the range between $\Lambda_i^{\mathtt{LB}}$ and $\Lambda_i^{\mathtt{UB}}$. The above iterations are repeated until the difference between $\mathrm{P}_{\mathtt{H}}$ and $\mathrm{P}_{\mathtt{L}}$ is below a threshold ρ.

The SP prioritization procedure finds the order of SPs that can borrow RBs from others. Given an RB r_x in $\mathcal{R}_k$, we define the resource amount of r_x that s_k lends to another SP s_m by $\Omega_{k,m}^x = (|\mathcal{Q}_x \cap (\mathcal{C}_m \cup \mathcal{D}_m)|)/|\mathcal{Q}_x|$. Specifically, the numerator is the number of s_m's UEs using r_x and the denominator gives

Procedure `lender_selection`(s_k):

1 **foreach** $s_m \in \mathcal{S} \setminus \{s_k\}$ **do**

2 $\quad \Psi_m \leftarrow 0$;

3 $\quad$ **foreach** $r_x \in \mathcal{R}_k$ **do**

4 $\quad\quad \Psi_m \leftarrow \Psi_m + \Omega^x_{k,m}$;

5 **return** $f^{\mathtt{DEC}}_{\mathtt{SRT}}(\mathcal{S} \setminus \{s_k\}, \Psi_m)$;

the number of total UEs that share r_x. Then, this procedure has two modes:

1. Lines 1–5 and 11–12: When s_k has lent more resources to others (i.e., $\Omega^{\mathtt{sum}}_k$ is larger), it has a higher priority to borrow RBs. This mode is to encourage SPs to lend RBs.
2. Lines 6–10 and 13: If s_k has borrowed fewer resources from other SPs (i.e., $\Psi^{\mathtt{sum}}_m$ is smaller), it possesses a higher priority to borrow RBs. This mode is for fairness.

We use mode 1 by default and periodically switch modes.

Given SP s_k, the lender selection procedure finds the order of other SPs that lend RBs to s_k. Generally, if an SP s_m has borrowed more resources from s_k, s_m shall lend back its RBs to s_k first. In this way, we can better maintain fairness among SPs. Except s_k, for every SP s_m in $\mathcal{S}$, the for-loop in lines 1–4 computes the total resource amount (denoted by Ψ_m) that s_m has borrowed from s_k. Then, line 5 sorts all SPs in $\mathcal{S} \setminus \{s_k\}$ decreasingly by their Ψ_m values and returns the sorted result.

V. Experimental Evaluation

Our simulation is built on MATLAB, whose parameters are given in Table I. One BS offers 45 RBs/ms, where SPs s_1, s_2, and s_3 possess 10, 15, and 20 RBs. Each SP needs to serve 10 CUs and 5 to 30 D2D pairs. The minimum power for the BS and a D2D sender is set to -40 dBm. For path loss, $L(\cdot,\cdot)$ is the distance between a sender and a receiver, as measured in km. Shadowing fading is modeled using a log-normal distribution [21], with a mean and standard deviation of 0 dB and 8 dB.

We pick two methods for comparison. The *wireless resource virtualization with D2D communication (WRVD)* method [15] allocates RBs to CUs for different SPs. Each RB is reused by a D2D pair with the minimum interference. The *joint resource allocation and power control with RAN sharing (JAPS)* method [19] adjusts senders' power and lets SPs borrow RBs from others to improve throughput. We also consider a reduced version of our DACS framework (called *R-DACS*), a combination of only BRA and RTI algorithms. By comparing DACS with R-DACS, we can evaluate the impact extent of the IRS algorithm. In JAPS, R-DACS, and DACS, we set $\delta = 30$.

A. Comparison on Performance

Fig. 1(a) compares the service rate. In general, the service rate reduces when the number of D2D pairs grows, since more DUs compete for the fixed resource. WRVD makes each RB be reused by at most a D2D pair. Once there are more D2D pairs, many DUs cannot be allocated RBs. Therefore, WRVD has the lowest service rate. R-DACS allows an RB to be shared by at most δ UEs ($\delta = 30$), thereby improving RB utilization and the service rate. When an SP has insufficient RBs to serve its UEs, JAPS and DACS make the SP borrow RBs from others. They achieve nearly 0.98 of a service rate. This result demonstrates that both JAPS and DACS can exploit RAN sharing to increase the service rate significantly.

In Fig. 1(b), we evaluate energy efficiency. As WRVD has the lowest service rate and adopts fixed power for the BS and D2D senders, its energy efficiency is greatly lower than other methods. Though JAPS has a similar service rate with DACS, its energy efficiency is significantly lower than that of DACS. This result reflects that DACS performs much better than JAPS in power control.

B. Comparison on Fairness

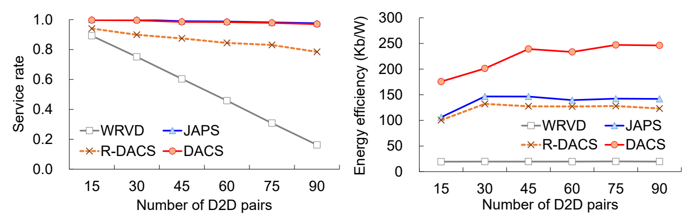
Both JAPS and DACS support RB lending mechanisms to let SPs borrow resources from each other to improve the service rate. In this section, we judge whether their mechanisms can maintain fairness among SPs.

Fig. 2(a) and (b) show each SP's throughput when there are 30 and 90 D2D pairs. Among 45 RBs offered by the BS (per ms), SPs s_1, s_2, and s_3 own 10, 15, and 20 RBs, which occupy 22.22%, 33.33%, and 44.45% of the BS's resources. Evidently, the gap between s_1's throughput and s_3's throughput is much larger in JAPS than that in DACS. Fig. 2(c) exhibits the ratio of each SP's throughput. Compared to JAPS, this ratio in DACS can be closer to the ratio of SPs' owned resources.

Fig. 3 gives the resource amount that each SP borrows and lends. JAPS will assign a high priority to an SP that has lent more RBs to others. Since SP s_3 owns the most RBs, it has a good chance of loaning out more RBs in the early stage. This

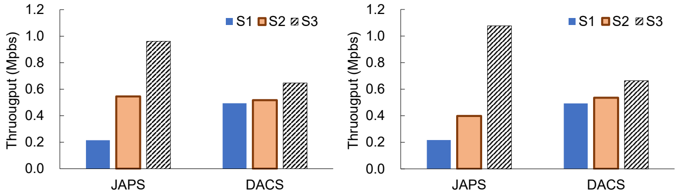
TABLE I
Simulation parameters.

BS parameters:	
cell range	500 m (channel bandwidth: 180 kHz)
maximum power	46 dBm ($\approx$ 40 W)
RBs per ms	45 (SP s_1, s_2, s_3 own 10, 15, 20 RBs)
UE parameters:	
number of UEs	CUs: 30, D2D pairs: 15, 30, 45, 60, 75, 90
maximum power	23 dBm ($\approx$ 0.2 W)
throughput demand	1 Mbps (i.e., $\sigma^{\min}_i \approx 17$ dB)
Channel parameters:	
path loss	BS to UE: $128.1 + 37.6 \log_{10} L(\text{BS}, u_i)$
	UE to UE: $148 + 40 \log_{10} L(\varepsilon(i), u_i)$
shadowing	log-normal distribution
thermal noise	-174 dBm/Hz



(a) service rate (b) energy efficiency

Fig. 1. Comparison of performance in WRVD, JAPS, R-DACS, and DACS.

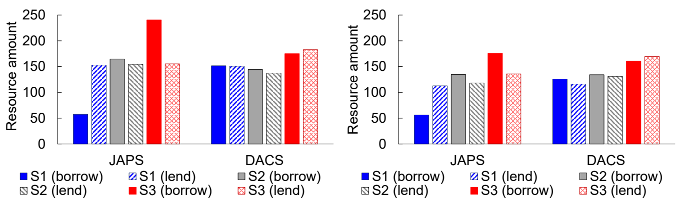


(a) 30 D2D pairs (b) 90 D2D pairs

pairs	JAPS	DACS
30	12.49% : 31.69% : 55.82%	29.78% : 31.22% : 39.00%
90	12.81% : 23.54% : 63.65%	29.13% : 31.64% : 39.23%

(c) ratio of each SP's throughput ($s_1 : s_2 : s_3$)

Fig. 2. Comparison of each SP's throughput.



(a) 30 D2D pairs (b) 90 D2D pairs

Fig. 3. Comparison of resource amount that each SP borrows and lends.

makes s_3 have a high priority for a long time. Eventually, s_3 will borrow much more RBs than it has lent. On the contrary, SP s_1 is forced to lend much more RBs than it borrows. These phenomena verify the unfairness of JAPS in RB borrowing.

In DACS, the SP prioritization procedure has two modes: 1) assigning a high priority to an SP that lent more resources, and 2) assigning a high priority to an SP that borrowed fewer resources. By switching between these two modes periodically, DACS can efficiently deal with the above problem. Moreover, the lender selection procedure asks an SP that has borrowed more resources from SP s_k to first repay RBs to s_k. As can be seen from Fig. 3, the resource amount that each SP borrows is similar to that it lends, which verifies that DACS can maintain fairness among SPs.

VI. Conclusion

This paper proposes the DACS framework to handle D2D resource allocation and power control with RAN sharing. Each SP allots its RBs to UEs and borrows RBs from others when it has not enough RBs. Then, the power of senders is raised for throughput improvement. In DACS, the power control procedure uses a ternary search to find suitable power. Moreover, the RB lending mechanism considers balancing between efficiency and fairness. Through simulations in MATLAB, we show that DACS not only attains a high service rate but also increases energy efficiency, as compared to WRVD, R-DACS, and JAPS. Moreover, DACS can keep fairness among SPs.

Acknowledgment

This work was supported by National Science and Technology Council, Taiwan under Grant 113-2221-E-110-056-MY3.

References

[1] P. Mach and Z. Becvar, "Device-to-device relaying: Optimization, performance perspectives, and open challenges towards 6G networks," *IEEE Communications Surveys & Tutorials*, vol. 24, no. 3, pp. 1336–1393, 2022.

[2] T. Islam and C. Kwon, "Survey on the state-of-the-art in device-to-device communication: A resource allocation perspective," *Ad Hoc Networks*, vol. 136, pp. 1–19, 2022.

[3] Y. C. Wang and C. W. Chou, "Efficient coordination of radio frames to mitigate cross-link interference in 5G D-TDD systems," *Computer Networks*, vol. 232, pp. 1–13, 2023.

[4] ETSI, "Network sharing; Architecture and functional description," 3GPP TS 23.251 V18.0.0, Mar. 2024.

[5] Y. C. Wang, "Resource and power management for in-band D2D communications," in *Horizons in Computer Science Research*. Hauppauge: Nova Science Publishers, 2022, vol. 21, ch. 5, pp. 131–183.

[6] F. Hussain, M. Y. Hassan, M. S. Hossen, and S. Choudhury, "An optimal resource allocation algorithm for D2D communication underlaying cellular networks," in *IEEE Consumer Communications & Networking Conference*, 2017, pp. 867–872.

[7] Z. Zhou, K. Ota, M. Dong, and C. Xu, "Energy-efficient matching for resource allocation in D2D enabled cellular networks," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 6, pp. 5256–5268, 2017.

[8] R. Gour and A. Tyagi, "Semi-distributed resource management for underlay D2D communication with user's cooperation," *International Journal of Communication Systems*, vol. 33, no. 4, pp. 1–18, 2020.

[9] L. Zhao, H. Wang, and X. Zhong, "Interference graph based channel assignment algorithm for D2D cellular networks," *IEEE Access*, vol. 6, pp. 3270–3279, 2018.

[10] W. K. Lai, Y. C. Wang, H. C. Lin, and J. W. Li, "Efficient resource allocation and power control for LTE-A D2D communication with pure D2D model," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 3, pp. 3202–3216, 2020.

[11] S. Guo, B. J. Hu, and Q. Wen, "Joint resource allocation and power control for full-duplex V2I communication in high-density vehicular network," *IEEE Transactions on Wireless Communications*, vol. 21, no. 11, pp. 9497–9508, 2022.

[12] Z. Liu, J. Su, Y. A. Xie, K. Ma, Y. Yang, and X. Guan, "Resource allocation in D2D enabled vehicular communications: A robust Stackelberg game approach based on price-penalty mechanism," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 8, pp. 8186–8200, 2021.

[13] T. V. Raghu and M. Kiran, "An iterative-based optimum power and resource allocation in application-dependent scenarios for one-to-one D2D communication," *IEEE Access*, vol. 12, pp. 38 972–38 984, 2024.

[14] H. M. F. Noman, K. Dimyati, K. A. Noordin, E. Hanafi, and A. Abdrabou, "FeDRL-D2D: Federated deep reinforcement learning-empowered resource allocation scheme for energy efficiency maximization in D2D-assisted 6G networks," *IEEE Access*, vol. 12, pp. 109 775–109 792, 2024.

[15] A. Moubayed, A. Shami, and H. Lutfiyya, "Wireless resource virtualization with device-to-device communication underlaying LTE network," *IEEE Transactions on Broadcasting*, vol. 61, no. 4, pp. 734–740, 2015.

[16] Y. Cheng and L. Yang, "Cost-oriented virtual resource allocation for device-to-device communications underlaying LTE networks," in *IEEE International Conference on Communication Software and Networks*, 2017, pp. 580–583.

[17] S. Gendy and Y. Gadallah, "A resource trading scheme in slice-based M2M communications," in *IEEE Vehicular Technology Conference*, 2019, pp. 1–5.

[18] L. Nadeem, Y. Amin, J. Loo, M. A. Azam, and K. K. Chai, "Quality of service based resource allocation in D2D enabled 5G-CNs with network slicing," *Physical Communication*, vol. 52, pp. 1–10, 2022.

[19] W. K. Lai, Y. C. Wang, and H. B. Lei, "Joint resource and power management for D2D communication across multiple service providers," *IEEE Systems Journal*, vol. 16, no. 3, pp. 3488–3499, 2022.

[20] G. Tseliou, K. Samdanis, F. Adelantado, X. C. Perez, and C. Verikoukis, "A capacity broker architecture and framework for multi-tenant support in LTE-A networks," in *IEEE International Conference on Communications*, 2016, pp. 1–6.

[21] Y. C. Wang and C. A. Chuang, "Efficient eNB deployment strategy for heterogeneous cells in 4G LTE systems," *Computer Networks*, vol. 79, pp. 297–312, 2015.