Rate-Splitting and Common Message Decoding in Hybrid Cloud/Mobile Edge Computing Networks

Robert-Jeron Reifert*, Hayssam Dahrouj†, Alaa Alameer Ahmad‡, Aydin Sezgin*,
Tareq Y. Al-Naffouri†, Basem Shihada† and Mohamed-Slim Alouini†
*Institute of Digital Communication Systems, Ruhr-Universität Bochum, Germany
†Communication Theory Lab, King Abdullah University of Science and Technology, Thuwal, Saudi Arabia
‡Volkswagen Infotainment GmbH, Bochum, Germany

***Abstract*—This paper proposes, and evaluates the benefits of, a hybrid central cloud (CC) and mobile edge computing (MEC) platform, especially introduced to balance the network resources for joint computation and communication. The transmission is further empowered by splitting the users' messages into private and common parts, to mitigate the interference within the CC and MEC platforms. While several power-hungry, computationally-limited unmanned aerial vehicles (UAVs) are deployed at the cell-edge to boost the CC connectivity and relieve part of its computation burden, the CC connects to the base-stations via capacity-limited fronthauls. The paper then considers the problem of maximizing the weighted sum-rate subject to fronthaul and computation capacity, achievable rates, power, delay, and data-split constraints, so as to determine the beamforming vectors associated with the private and common messages, the computation allocations, and the data-split factors. Such intricate non-convex optimization problem is tackled using an iterative algorithm that relies on ℓ_0-norm relaxation, successive convex approximation, and fractional programming, and can be compellingly implemented in a distributed fashion. The simulations illustrate the proposed algorithm's capabilities for empowering joint communication and computation, and highlight the pronounced role of rate-splitting and common message decoding in alleviating large-scale interference in hybrid CC/MEC networks.**

***Index Terms*—Rate-splitting, central cloud, mobile edge computing, hybrid networks, unmanned aerial vehicles.**

I. Introduction

A. Motivation

Today's Internet of Things (IoT) applications involve many relevant consumer and industry use cases, e.g., smart cities, modular plants, etc. For future beyond the fifth generation (B5G) of wireless communication systems, massive IoT is forecasted to make up to 51% of cellular IoT connections by 2027, while other use-cases extend to augmented reality, vehicular to anything, etc. [1]. With massively increased number of connected devices and increased service requirements, massive IoT raises further challenges towards future B5G networks. That is, B5G systems need to handle diverse requirements, such as reliability, network density, and quality of service [2]. Conventional multiple access and interference management schemes, e.g., space division multiple access (SDMA) and treating interference as noise (TIN) strategies, are unsuited for handling the enormous densification and network load development. More specifically, SDMA and TIN are suboptimal in cases of networks with a large number of network participants and moderate to strong interference levels [3].

In this context, the rate-splitting (RS) multiple access scheme emerges as a promising architecture for B5G physical and medium access control layer [3]. First proposed in [4], RS was used for simple interference channels with two transmitters and two receivers. In principle, RS relies on splitting messages into multiple parts at the transmitter side and successively decoding such messages at the receiver side [5]. Such decoding scheme is referred to as successive interference cancellation (SIC). Specifically, in multi-user networks, each user's message is split into two parts, whereas only one part, i.e., the common message, may be decoded by a subset of users [6], [7]. Under such scheme, referred to as RS and common message decoding (CMD), the determination of common message decoding sets becomes a part of the network design. RS multiple access is indeed provably able to flexibly bridge the gap between treating the interference as noise and decoding the interference; thereby covering the conventional schemes as special cases [8]. In addition to its improved interference management, RS is able to boost data rates, reliability, and latency [3].

While RS has been proven to empower communication architectures operation, and is able to serve B5G network needs, the practicality of the developed resource management techniques under RS-enabled networks necessitates further considerations, especially those related to the availability of the computational resources. That is, network users, e.g., IoT devices, are often dependent on offloading their computation tasks, especially for computation-intensive applications, due to their finite energy and computation resources [9]. A suitable technique for satisfying such massive data demand is the cloud-based network architecture, which enables centralized management of communication and computation resources. However, a drawback of cloud-based networks is the long propagation delay [10] and the need for costly, limited-capacity fronthaul links to connect the cloud to the base-stations (BSs). Especially for latency-critical and bandwidth-hungry B5G use cases, moving computation and management capabilities towards the network edge enables both latency reduction and service quality enhancement. Such cost-efficient and energy-saving paradigm, referred to as mobile edge computing (MEC), is subject, however, to strict constraints on power and computational resources.

e To this end, this paper considers one particular hybrid network architecture, where the central cloud (CC) connects to central BSs so as to serve the central network users. The cell-edge users, on the other hand, are served by resource-limited MEC devices, especially deployed to boost the system connectivity. The RS paradigm then imposes splitting each user's message into one private part decodable by the intended user only, and a common part decodable by a subset of users. For the ease of practical implementation, i.e., to minimize the message passing between the CC and the multiple MEC platforms, the paper assumes that the common message decoding is restricted to users being served by the same platform exclusively (i.e., either the CC or the MEC platform), so as to handle the undesired intra-platform interference. The paper then adopts such a hybrid CC/MEC architecture to empower joint communication and computation by means of maximizing a network-wide sum-rate, the performance of which is a function of the allocated computation and communication resources, i.e., computation capacity, the beamforming vectors associated with the private and common messages, the common message decoding sets, and the data-split factors. We refer to the RS scheme utilized in this work as RS-CMD. The paper then focuses on solving a weighted sum-rate maximization problem subject to fronthaul capacity, computation capacity, achievable private and common rate, per-platform power, per-user delay, and data-split constraints. An efficient iterative algorithm is proposed to tackle the intricate problem in a decentralized manner across the CC and MEC platforms.

B. Related Works

The hybrid CC/MEC computation and RS-based communication resource management problem considered in this paper is related to recent works on RS, resource management in cloud-radio access networks (C-RANs), MEC, UAV-aided MEC, and hybrid network architectures.

RS has been recognized as a promising interference management and multiple access scheme in the recent literature, e.g., [3], [8], [11]–[13]. For instance, the work [3] provides a comprehensive overview on RS multiple access including a survey of the related state-of-the-art, discussing various architectures and techniques, as well as providing future research directions in the field. The work [11] considers a multi-user multiple input single output (MISO) system under the RS strategy, where users decode parts of the interference, and shows increased data rate performance under RS. In [8], RS multiple access is compared to TIN and non-orthogonal multiple access (NOMA). RS is shown to bridge TIN and NOMA schemes, i.e., RS is able to bridge between treating all the interference as noise and decoding the interference completely, for different setups considering network load and user deployment. Various additional studies provide theoretical insights and promising results of the RS paradigm [12], [13]; however, the study of RS in future sophisticated networks remains limited in breadth and depth, and so the current paper investigates the performance of optimized RS-enabled networks in the context of a hybrid CC/MEC setup.

From a network architecture perspective, C-RANs have been recognized as a strong enabler to support the highly anticipated densification and network load [14]. Resource management in C-RANs has been the topic of various related studies [15]–[17]. For instance, to tackle the network densification challenge, the work [15] proposes a multi-cloud architecture introducing intra- and inter-cloud interference. However, as [15] employs SDMA, the promising interference mitigation capabilities of the RS paradigm are rather ignored in [15]. Along the same lines, RS is shown to provide a multitude of benefits compared to conventional system strategies in multicell systems [7], [18], and cloud-based systems [6], [19]–[25]. More specifically, RS is shown to outperform various reference schemes in terms of sum-rate [6], [18], [20], [22], minimum rate [7], [21], [23], spectral efficiency [19], [26], and energy efficiency [25]–[28], to only name a few. In [7], the authors consider a practical multicell setup and illustrate the increased minimum rate for cell-edge users. The authors in [21] further consider a practical RS setup under hierarchical clustering for determining the common message decoding sets. Such scheme is shown to outperform SDMA, NOMA, a single common message-based RS scheme, and clustering based on random selection. The work [26] shows that one-layer RS yields improved performance without increasing the receiver complexity. Such results are especially interesting from an IoT perspective, where devices are usually constrained by battery power and receiver complexity [20], [29].

Motivated by the scarce computational resources in IoT devices, the massive network load, and the densification of B5G networks, the area of mobile edge computing promises to provide several complementary services to the network operators. Under MEC, the cloud's intelligence is moved towards the network edge in the vicinity of the users [30]. Thus, MEC is able to support a variety of B5G use cases, especially those related to latency-sensitive and critical components. Several MEC-based works focus on the computation [9], [31], [32] and communication [10] aspects. In more detail, reference [9] emphasizes the appreciable improvement of computation performance and execution latency under the MEC paradigm, and [10] provides a survey on the communication perspectives of MEC.

Unlike the previous works on MEC networks which adopt orthogonal access schemes, e.g., [33], [34], this paper adopts a more general non-orthogonal transmission scheme, i.e., RS multiple access, so as to mitigate the multi-user inter and intra-platform interference. More specifically, the application of RS to MEC-based networks remains limited to a few recent works, e.g., [35]–[37]. In [35], a closed-form expression for the computation probability in RS-aided MEC systems is proposed. By optimizing offloading ratios and times, the authors in [35] show that RS outperforms several orthogonal multiple access techniques. The work [36] proposes an unmanned aerial vehicle (UAV)-aided MEC network, where the UAVs act as relays and collect data from IoT devices. The considered model in [36] focuses on minimizing the UAV's energy consumption subject to transmission rate and computation time constraints. The interplay of aerial users and high-altitude platforms for MEC is also analyzed in [37], which adopts a reinforcement learning problem to minimize latency and energy consumption.

Similar to [36], UAV-assisted MEC has been recognized as a promising B5G network technique for allowing flexible deployment, on-demand service, and enhanced connectivity [10], [38], [39]. The utilization of UAVs, with typically strict power constraints, calls for sophisticated joint management of communication and computation resources in order to optimize the system performance. Due to weak received power and strong adjacent network interference, especially in the B5G ultra-dense deployment, cell-edge users are often prone to inferior service quality, which makes the development of communication and computation resource management techniques vital. Related works in this field include UAV-aided communication [40], [41] and computation [31], [42] networks. Additional studies consider the RS scheme in UAV networks [40], [43], [44]. These works, however, do not capture the joint consideration of the communication and computation aspects, i.e., constraint-wise and variable-wise, e.g., see [36], [38].

As B5G communication networks are envisioned to include multiple access technologies in a hybrid manner, the need to consider the interplay of a central network and edge devices arises. Studying and optimizing the interplay of RS together with such hybrid networks emerges, therefore, as a timely direction to assess the true gain of augmenting hybrid CC/MEC networks with rate-splitting access schemes, which is studied in this paper in detail. More specifically, in contrast to related UAV-focused literature, e.g., [38], [39], [45], this current work extends the joint communication and computation paradigm toward RS-aided hybrid cloud and MEC networks. The existing works on the topic either employ RS in C-RAN, e.g., [21], utilize UAVs to aid the RS-enabled network, e.g., [40], or employ RS-based MEC, e.g., [35]. The current paper becomes, therefore, the first of its kind which considers the benefits of rate-splitting and common message decoding in a hybrid CC/MEC system, by means of optimizing the network resources subject to delay, computation, and transmissions constraints, considerations that fall at the intersection of communication, computing, and networking.

C. Contributions

Unlike the aforementioned references, in this paper, we propose a downlink hybrid CC/MEC network, where several multi-antenna BSs and UAVs serve single-antenna network users under the RS-CMD paradigm. The BSs are connected to the central cloud via capacity limited fronthaul links, while the UAVs perform computation and communication functions on their own. The users' messages are then split into both private and common messages, for the sole purpose of mitigating the interference within the CC and MEC platforms. The common message decoding sets are chosen to suit the reasonable cloud-edge separation, i.e., common message decoding is enabled among either CC users, or edge computer users only. We address a weighted sum-rate maximization problem by jointly managing private and common messages' beamforming vectors, allocated rates, computation capacity, and data-split fractions, subject to per-BS fronthaul capacity, per-computing platform maximum computation capacity, per-user achievable private and common rate, per-BS and per-UAV power, and per-user delay and data-split constraints. Specifically, the UAV power constraints account for the joint computation and communication aspect including operational power. The per-user delay constraints account for computation, fronthaul, and transmission delays, all given a non-trivial RS-CMD transmission scheme. Such mixed discrete-continuous non-convex optimization problem is tackled using ℓ_0-norm relaxation, successive convex approximation (SCA), and fractional programming (FP) resulting in an efficient decentralized algorithm. Insightful simulations verify the gains of the proposed network architecture in terms of data rate, delay, and power consumption. The proposed decentralized algorithm is particularly shown to overcome the centralized version in terms of computational complexity, runtime, and scalability, and to provide an appreciable gain as compared to conventional system strategies that employ private information transmission only, i.e., TIN. The contributions of this study can be summarized as follows

1) *Joint Computation and Communication Aspect in RS-Based Hybrid CC/MEC Networks:* In the studied system model, we propose the joint consideration of network resources allocated for computation and communication tasks. That is, the considered network variables include allocated computation capacities by the CC and MEC platforms, the beamforming vectors of private and common messages, and the data-split factors. In this context, the CC and all MEC platforms are able to flexibly adjust such variables for maximizing the overall weighted sum-rate performance, all while finding trade-offs for different constraint considerations: (1) Adjusting computation and transmission power at the battery-limited UAVs; (2) Leveling computation, fronthaul, and transmission delay for each user; (3) Finding reasonable data-splits between private and common messages.
2) *Optimization Framework:* We formulate a weighted sum-rate maximization problem subject to per-BS fronthaul capacity, per-computing platform maximum computation capacity, per-user achievable private and common rate, per-BS and per-UAV power, and per-user delay and data-split constraints. The general solution to the non-convex problem firstly relaxes the binary variables using ℓ_0-norm approximation, then utilizes an SCA approach to obtain convex constraints, and finally utilizes FP to exploit the fractional structure of the constraints. An efficient iterative algorithm for resource management in hybrid CC/MEC networks is consequently proposed.
3) *Distributed Approach:* A distributed solution relying on a decentralized implementation of the proposed algorithm across the network computing platforms, i.e., the CC and the edge computers (ECs), illustrates the suitability of the proposed solution for practical implementation in hybrid CC/MEC networks. Specifically, the design of RS and common message decoding sets suits the reasonable cloud-edge separation, as common messages are decodable only at users within the same associated computing platform.
4) *Numerical Simulations:* Various numerical simulations illustrate the appreciable improvements of the proposed

methods as compared to benchmark schemes. In particular, it is shown that the proposed hybrid CC/MEC framework significantly outperforms schemes which employ a central cloud only. Rate gains of up to around 356% from utilizing hybrid CC/MEC are reported. Most importantly, the paper results highlight how the RS-CMD scheme significantly outperforms the private information transmission only, i.e., TIN, in terms of average rate, power consumption, per-user average delay, feasibility of solution, and relative runtime decrease for various network parameters, e.g., strict delay and computation capacity constraints. In addition, the distributed implementation of the proposed algorithm is shown to perform close to the centralized solution in a wide-range of parameters, all while achieving up to 50% of runtime savings.

D. Notations and Organization

This paper denotes vectors, matrices, and sets as boldface lower-case, capital, and calligraphic letters (e.g., $\mathbf{h}$, $\mathbf{H}$, and $\mathcal{H}$), respectively. Further, let $\mathbf{0}_N$ be the all-zero vector of length N, $\mathbb{R}$ be the real and $\mathbb{C}$ be the complex field, and the real part of a complex number be $\Re\{\cdot\}$. Finally, hermitian transpose, transpose, the absolute value, and the ℓ_p-norm are denoted as $(\cdot)^\dagger$, $(\cdot)^T$, $|\cdot|$, and $\|\cdot\|_p$, respectively.

The paper is organized as follows, Section II introduces the system model in terms of the received signal model, network constraints and variables, and the RS-CMD scheme, and formulates the problem. Subsequently, Section III follows the problem reformulation steps and provides both a centralized and decentralized algorithm for resource management in hybrid CC/MEC networks. Section IV then provides various numerical results analyzing the performance of the proposed schemes and several benchmarks. Finally, concluding remarks are given in Section V.

II. System Model

In this work, we consider the downlink of a hybrid cloud/MEC-based network architecture under the RS-CMD paradigm. Under such framework, cloud processors (CPs) coordinate the users operations within the core-network. The UAVs, on the other hand, with on-chip computation capabilities, act as mobile ECs to serve the cell-edge users. The RS-CMD scheme enables splitting downlink data into a common and private part. Both common and private messages are transmitted in parallel, whereas common messages are eligible to be decoded by a multitude of users, which are associated with the same platform, i.e., the CC or ECs, for the purpose of interference mitigation. The core network consists of a single cloud, i.e., the CC, connected via fronthaul links to B multi-antenna BSs, with L_c antennas each, while the UAVs at the edge are equipped with L_e antennas each. The network functions at the CC follow a *data-sharing* approach, where the CP at the CC performs most network functions, e.g., encoding and precoder design, leaving the modulation, precoding, and radio tasks to the BSs [16]. The ECs perform all those tasks

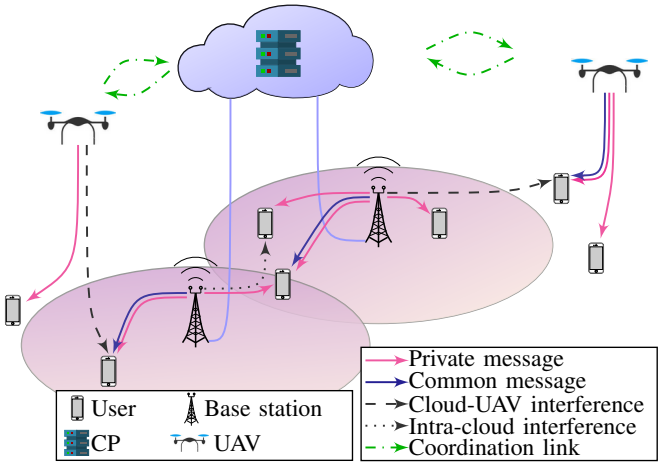


Fig. 1. Network of 2 UAVs, 2 BSs, and 6 users.

on their own. Contrary to related works on RS-CMD in C-RAN, under the current system model, the rate-splitting design happens on a per-computing platform (i.e., CC or ECs) basis. That is, common message decoding is only possible among CC users, treating all edge users as noise, or among each EC's users, treating all CC users and all other ECs users as noise. The rationale behind such a restriction is that a network-wide common message decoding would require a signal-level cooperation, which is unreasonable in a distributed hybrid CC/MEC network, where the CC and the ECs can solely rely upon control-level coordination schemes. Fig. 1 shows an example of the considered system, which illustrates a network of 2 BSs serving 4 central users, and 2 UAVs each serving one user.

Let E be the number of deployed ECs, and let $\mathcal{E} = \{1, \cdots, E\}$ be the set of ECs. Since each EC is implemented on a UAV, the edge network consists of E UAVs. Note that throughout this work, the terms ECs and UAVs are interchangeably used without loss of generality. The set of BSs is given by $\mathcal{B} = \{1, \cdots, B\}$. The set of single-antenna users is denoted by $\mathcal{K} = \{1, \cdots, K\}$, where K is the total number of users. In the context of CC and EC coexistence, this paper assumes disjoint user-clusters, which are covered by the user sets $\mathcal{K}_c \subseteq \mathcal{K}$ and $\mathcal{K}_e \subseteq \mathcal{K}$, with $\mathcal{K}_c \cap \mathcal{K}_e = \emptyset, \forall e \in \mathcal{E}$ and $\mathcal{K}_e \cap \mathcal{K}'_e = \emptyset, \forall e \neq e'$. In other terms, the set of users served by the CC is denoted by $\mathcal{K}_c$, while the set of users served by each EC e is denoted by $\mathcal{K}_e$, $\forall e \in \mathcal{E}$. Similarly, the CC serves K_c users, while EC e serves K_e users.

We further denote the channel vector from BS b to user k by $\boldsymbol{h}_{b,k} \in \mathbb{C}^{L_c}$, and the channel vector from UAV e to user k by $\bar{\boldsymbol{h}}_{e,k} \in \mathbb{C}^{L_e}$, where $e \in \mathcal{E}$. The aggregate channel vector from all BSs and UAVs towards user k is given by $\boldsymbol{h}_k = [\boldsymbol{h}_{1,k}^T, \cdots, \boldsymbol{h}_{B,k}^T, \bar{\boldsymbol{h}}_{1,k}, \cdots, \bar{\boldsymbol{h}}_{E,k}]^T$. For mathematical tractability, the paper assumes full knowledge of channel state information at the transmitters (CSIT).

This work's communication architecture makes use of the RS-CMD paradigm. More specifically, the message of each user is split into a common and a private part. In more detail, each user's data is split into the private part using $\nu_k D_k$, with D_k being the data size of user k and $0 \leq \nu_k \leq 1$ being the split factor. The common data size becomes thus $(1-\nu_k)D_k$.

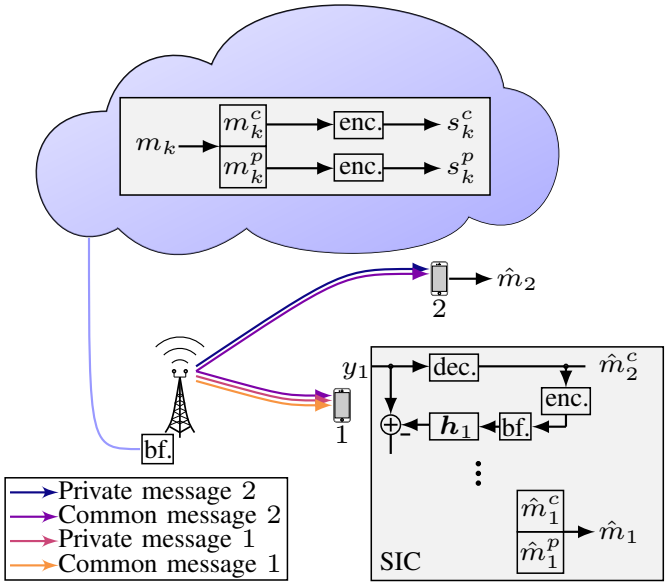


Fig. 2. Details on the RS-CMD transceiver design for a two user network. Here enc., bf., dec. refer to encoding, beamforming, and decoding, respectively.

Additionally, the splitted signals s_k^p and s_k^c, i.e., the respective private and common signals of user k, are transmitted by BS b using beamforming vectors $\boldsymbol{w}_{b,k}^p$ and $\boldsymbol{w}_{b,k}^c$, or by UAV e using beamforming vectors $\boldsymbol{v}_{e,k}^p$ and $\boldsymbol{v}_{e,k}^c$, respectively. The data rate of user k consists of r_k^p, the private rate, and r_k^c, the common rate. The beamforming vectors $\boldsymbol{w}_{b,k}^p$, $\boldsymbol{w}_{b,k}^c$, $\boldsymbol{v}_{e,k}^p$, and $\boldsymbol{v}_{e,k}^c$ are occasionally denoted in the paper by $\boldsymbol{w}_{b,k}^\varphi \in \mathbb{C}^{L_c \times 1}$ and $\boldsymbol{v}_{e,k}^\varphi \in \mathbb{C}^{L_e \times 1}$, respectively, where $\varphi \in \{p, c\}$ is used to point to the respective private and common messages. Without loss of generality, the paper assumes that the signals are zero mean, unit variance complex Gaussian variables which are independent, identically distributed and circularly symmetric. Under RS-CMD, s_k^p is intended to only be decoded by user k and s_k^c, i.e., the common signal of user k, may be decoded by multiple users, which are part of the subset of users served by the same platform, to mitigate parts of their interference, including k itself. Therefore, consider the following list of network definitions related to RS-CMD:

- Let the set of users decoding user k's common message be $\mathcal{M}_k = \{j \in \mathcal{K} | \text{user } j \text{ decodes } s_k^c\}$.
- Let the set of users, whose common messages are decoded by user k, be $\mathcal{I}_k = \{j \in \mathcal{K} | k \in \mathcal{M}_j\}$.
- Let the decoding order at user k be π_k, where $\pi_k(m) > \pi_k(i)$ refers to user k decoding common message i before common message m.
- Let the set of users, whose common messages pose interference to user k during the decoding of user i's common message, be $\mathcal{I}'_{i,k} = \{m \in \mathcal{I}_k | \pi_k(m) > \pi_k(i)\}$. These messages are decoded in a later stage of SIC.

For illustration, Fig. 2 pictures an example of the RS-CMD transceiver structure at the CC in details. The message m_k of user k is split into common and private parts and encoded at the cloud. Afterwards, these signals and beamforming vectors are forwarded over the fronthaul link to the BSs. At the receiver side, the user decodes the messages according to the decoding order π_k and performs SIC, i.e., encoding the previously decoded message, beamforming, adding the channels, and subtracting this from the remaining received signal. Note that the design and transmission aspects are combined for each EC. Under the specific setup in Fig. 2, we have $\mathcal{I}_1 = \{1, 2\}$, $\mathcal{I}_2 = \{2\}$, and $\pi_1(1) > \pi_1(2)$, i.e., user 1 decodes its own messages and user 2's common message, user 2 decodes only its own messages, and user 1 decodes first user 2's common message and then its own, respectively. The adopted method to determine these RS-CMD sets, namely, $\mathcal{M}_k$, $\mathcal{I}_k$, the decoding order, and $\mathcal{I}'_{i,k}$, mirrors the method used in [6] in the context of C-RANs, and is presented in Appendix A of this paper in the context of hybrid CC/MEC networks for completeness. More specifically, similar to [6], this paper adopts a common message set determination algorithm that relies on two specific parameters: (1) The percentage of users being chosen for common message decoding denoted by η; (2) The maximum number of SIC steps at any user D. Network operators may choose to set D sufficiently low to lessen the receiver complexity. More details on determining the common message set can be found in Appendix A of the paper.

Given the above considerations, the received signal at user k can be mathematically expressed as

$$y_k = \boldsymbol{h}_k^\dagger \boldsymbol{a}_k^p s_k^p + \sum_{j \in \mathcal{I}_k} \boldsymbol{h}_k^\dagger \boldsymbol{a}_j^c s_j^c + \sum_{m \in \mathcal{K} \setminus \{k\}} \boldsymbol{h}_k^\dagger \boldsymbol{a}_m^p s_m^p + \sum_{l \in \mathcal{K} \setminus \mathcal{I}_k} \boldsymbol{h}_k^\dagger \boldsymbol{a}_l^c s_l^c + n_k, \quad (1)$$

where $\boldsymbol{a}_k^\varphi = [(\boldsymbol{w}_{1,k}^\varphi)^T, \cdots, (\boldsymbol{w}_{B,k}^\varphi)^T, (\boldsymbol{v}_{1,k}^\varphi)^T, \cdots, (\boldsymbol{v}_{E,k}^\varphi)^T]^T$ is the aggregate beamforming vector of user k and $n_k \sim \mathcal{CN}(0, \sigma^2)$ is the additive white Gaussian noise. Specifically, equation (1) consists of the following components, the received private signal of k, the common signals to be decoded by k, all other users' private signals, all common signals, which are not decoded by k, and the noise, in order of their appearance. Note that the first two terms are desired signals, while the latter three are interference and noise.

To jointly manage the communication and computation network resources, this work considers the following variables:

- The *beamforming vectors* $\boldsymbol{w}_{b,k}^\varphi \in \mathbb{C}^{L_c}$ and $\boldsymbol{v}_{e,k}^\varphi \in \mathbb{C}^{L_e}$ for transmitting user k's signals at BS b and at UAV $e \in \mathcal{E}$, respectively.
- The *computation vector* $\boldsymbol{f} \in \mathbb{N}^K$ capturing the allocated computation cycles to process the users data, where each entry f_k is given in cycles/s, $\forall k \in \mathcal{K}$.
- The *rate vector* $\boldsymbol{r} \in \mathbb{R}^K$ containing the achievable user rates. Under RS-CMD, let $r_k = r_k^p + r_k^c$.
- The *data-split vector* $\boldsymbol{\nu} \in \mathbb{R}^K$ denoting each user's common and private message data-split.

As users are associated with either the CC or one of the ECs, the aggregate beamforming vector $\boldsymbol{a}_k^\varphi$ is group-sparse by design. For each CC user, we may find $\boldsymbol{w}_{b,k}^\varphi = \boldsymbol{0}^{L_c}$ for some BSs b, since not all BS serve user k. A similar statement holds for the EC users, i.e., $\boldsymbol{v}_{e,k}^\varphi = \boldsymbol{0}^{L_e}$ for some UAVs e, which do not serve user k.

In what follows, we describe the metrics of this paper interest, mainly, the private and common data rates, the

power consumption at the BSs and at the UAVs, the users' transmission delays, and the computation capacity.

a) Achievable Rate: In (1), we observe each user receiving a mixture of signals, i.e., the common signals to be decoded, its own intended private signal, and all other signals plus noise, which pose interference. Decoding each desired signal, we next express the different formulations of the signal to interference plus noise ratio (SINR). Specifically, user k first decodes all common messages of $\mathcal{I}_k$, and so the SINR of user i's common message at user k becomes

$$\Gamma^c_{i,k} = \frac{\left|\boldsymbol{h}_k^\dagger \boldsymbol{a}_i^c\right|^2}{\sum\limits_{j\in\mathcal{K}} \left|\boldsymbol{h}_k^\dagger \boldsymbol{a}_j^p\right|^2 + \sum\limits_{l\in\mathcal{K}\setminus\mathcal{I}_k\cup\mathcal{I}'_{i,k}} \left|\boldsymbol{h}_k^\dagger \boldsymbol{a}_l^c\right|^2 + \sigma^2}, \quad (2)$$

where σ^2 is the additive white Gaussian noise variance. In (2), the common signal of i is interfered by all network private messages and all common messages, which are not decoded at k or to be decoded in a later SIC step. At last, user k decodes its own private signal with SINR

$$\Gamma^p_k = \frac{\left|\boldsymbol{h}_k^\dagger \boldsymbol{a}_k^p\right|^2}{\sum\limits_{j\in\mathcal{K}\setminus\{k\}} \left|\boldsymbol{h}_k^\dagger \boldsymbol{a}_j^p\right|^2 + \sum\limits_{l\in\mathcal{K}\setminus\mathcal{I}_k} \left|\boldsymbol{h}_k^\dagger \boldsymbol{a}_l^c\right|^2 + \sigma^2}. \quad (3)$$

The achievable rates of user k's private and common message are then expressed as

$$r^c_k \le \tau \log_2(1+\Gamma^c_{i,k}), \quad (4)$$

$$r^p_k \le \tau \log_2(1+\Gamma^p_k), \quad (5)$$

respectively, where τ is the transmission bandwidth. Additionally, for all CC users, keeping all signals to be transmitted by a BS b within the limits of the finite fronthaul capacity[1] $R_b^{\max}$, we write the fronthaul capacity constraint as

$$\sum_{k\in\mathcal{K}_c} \left\|\left\|\boldsymbol{w}^p_{b,k}\right\|_2^2\right\|_0 r^p_k + \left\|\left\|\boldsymbol{w}^c_{b,k}\right\|_2^2\right\|_0 r^c_k \le R_b^{\max},\ \forall b\in\mathcal{B}, \quad (6)$$

where the ℓ_0-norm in (6) is a non-linear function that determines the number of non-zero elements. Thus, (6) ensures that if BS b does not assign any power to serve user k, the rate of k does not contribute to BS b's fronthaul link.

b) Power Consumption: This paper considers two power consumption metrics, P_b^{cc} and P_e^{ec}, i.e., BS b's and EC e's power consumption, respectively:

$$P_b^{\mathrm{cc}} = \sum\nolimits_{k\in\mathcal{K}_c} \left\|\boldsymbol{w}^p_{b,k}\right\|_2^2 + \left\|\boldsymbol{w}^c_{b,k}\right\|_2^2, \quad (7)$$

$$P_e^{\mathrm{ec}} = \underbrace{\sum\nolimits_{k\in\mathcal{K}_e} \left\|\boldsymbol{v}^p_{e,k}\right\|_2^2 + \left\|\boldsymbol{v}^c_{e,k}\right\|_2^2}_{\text{Transmission}} + \underbrace{s_e\left(\sum\nolimits_{k\in\mathcal{K}_e} f_k\right)^{\mu_e}}_{\text{Computation}} + \underbrace{Q_e}_{\text{Operation}}, \quad (8)$$

where s_e and μ_e are constants that depend on the CPU model [32]. While BSs are solely restricted in their communication transmit power, the mobile ECs are subject to more elaborate power constraints. More specifically, (8) consists of transmit, computation, and operational power, see [32], whereas the first two are directly coupled to the optimization variables, and the operational power Q_e is fixed, accounting for mechanical and operational power.

c) Delay Considerations: In this paper, a user's overall delay is a function of the following components:

$$\Theta_k = \underbrace{F_k/f_k}_{\text{Computation Delay}} + \underbrace{\Lambda_k}_{\text{Fronthaul Delay}} + \underbrace{D_k/r_k}_{\text{Transmission Delay}}, \quad (9)$$

where F_k, Λ_k, and D_k denote the total computation cycles required for user k's task, the worst-case fronthaul delay of all BSs serving k, and the data size for k, respectively. More specifically, equation (9) consists of a computation delay, i.e., the time it takes to process k's requested task, a fronthaul delay, i.e., the time-loss on the fronthaul link, and a transmission delay, i.e., the latency during the wireless data transfer. Note that for the delay of user k served by EC e, i.e., if k is an edge user, the fronthaul delay is rather negligible, and is ignored in the context of our paper.

Under RS-CMD, the delay considerations change significantly, as each user k decodes common messages of all users in $\mathcal{M}_k$ (including its own common message) in addition to its own private message. Hence, user k's delay is coupled with the users whose common messages are to be decoded at user k. Specifically, under the RS-CMD setup, equation (9) becomes

$$\Theta_k^{\mathrm{cc}} = \max\left\{ \frac{F_k}{f_k} + \Lambda_k + \frac{\nu_k D_k}{r^p_k},\ \max_{k'\in\mathcal{M}_k}\left\{ \frac{F_{k'}}{f_{k'}} + \Lambda_{k'} + \frac{(1-\nu_{k'})D_{k'}}{r^c_{k'}} \right\} \right\}. \quad (10)$$

To explain the above delay consideration, we note that (10) consists of the same aspects as (9), namely, computation, fronthaul, and communication delay, for private and common messages. The first term of the max-operator in (10) denotes the delay of user k's private part. The second term of the max-operator in (10) originates from the RS-CMD paradigm. It refers to the maximum delay of all users, whose common messages are decoded by k. In other words, since k decodes messages from users $k'\in\mathcal{M}_k$, k's delay is dependent on the maximum delay these messages experience. In this system model, we consider parallel computing of user data at the computing platform (i.e., CC or ECs), e.g., by virtual machines. Similarly, data on the fronthaul link and in the wireless medium are transmitted in parallel, which is the reason for using the max-operators in (10).

Please note that we denote the delay formulation at the ECs as Θ_k^{ec}, which is similar to (10) but without the fronthaul delays Λ_k. In this work, the fronthaul delay is expressed as

$$\Lambda_k = \max_{b\in\mathcal{B}_k}\left\{ \frac{1}{R_b^{\max}} \sum_{i\in\mathcal{K}_c} \left\|\left\|\boldsymbol{w}^p_{b,i}\right\|_2^2\right\|_0 \nu_i D_i + \left\|\left\|\boldsymbol{w}^c_{b,i}\right\|_2^2\right\|_0 (1-\nu_i) D_i \right\}, \quad (11)$$

where $\mathcal{B}_k$ is the set of BSs serving user k. The rationale behind (11) is that each BS's fronthaul link is occupied by all the data it transmits to its associated users. The ℓ_0-norm of the beamforming vectors thus determines whether common (or private) data parts are forwarded over the fronthaul link.

[1]Compared to sharing the user signals over the fronthaul links, the rate to share the beamforming coefficients is negligible [17].

d) Computation Capacity: In this paper, each computation platform is subject to a specific maximum computation capacity constraint as follows:

$$\sum\nolimits_{k\in\mathcal{K}_c} f_k \le f_c^{\max}; \quad \sum\nolimits_{k\in\mathcal{K}_e} f_k \le f_e^{\max},\ \forall e\in\mathcal{E}, \quad (12)$$

where $f_c^{\max}$ and $f_e^{\max}$ denote the CC's and EC e's capacity, respectively.

A. Problem Formulation

Given the above considerations, the paper then aims at maximizing a weighted sum-rate by managing the beamforming vectors, allocated rates, computation allocation variables, and data-split factors, given by

$$\begin{aligned}&\boldsymbol{a}=[(\boldsymbol{a}_1^p)^T,\cdots,(\boldsymbol{a}_K^p)^T,(\boldsymbol{a}_1^c)^T,\cdots,(\boldsymbol{a}_K^c)^T]^T,\\ &\boldsymbol{r}=[r_1^p,\cdots,r_K^p,r_1^c,\cdots,r_K^c]^T,\boldsymbol{f}=[f_1,\cdots,f_K]^T,\\ &\boldsymbol{\nu}=[\nu_1,\cdots,\nu_K],\end{aligned}$$

respectively. The mathematical formulation of the considered problem can then be written as follows:

$$\begin{aligned}\max_{\boldsymbol{a},\boldsymbol{r},\boldsymbol{f},\boldsymbol{\nu}}\quad & \sum\nolimits_{k\in\mathcal{K}}\alpha_k(r_k^p+r_k^c) && (13)\\ \text{s.t.}\quad & (6),(12),\\ & r_i^c\le\tau\log_2(1+\Gamma_{i,k}^c), && \forall i\in\mathcal{I}_k,\forall k\in\mathcal{K}, \quad (13a)\\ & r_k^p\le\tau\log_2(1+\Gamma_k^p), && \forall k\in\mathcal{K}, \quad (13b)\\ & P_b^{\rm cc}(\boldsymbol{a},\boldsymbol{f})\le P_b^{\max}, && \forall b\in\mathcal{B}, \quad (13c)\\ & P_e^{\rm ec}(\boldsymbol{a},\boldsymbol{f})\le P_e^{\max}, && \forall e\in\mathcal{E}, \quad (13d)\\ & \Theta_k^{\rm cc}(\boldsymbol{r},\boldsymbol{f})\le t_k, && \forall k\in\mathcal{K}_c, \quad (13e)\\ & \Theta_k^{\rm ec}(\boldsymbol{r},\boldsymbol{f})\le t_k, && \forall k\in\mathcal{K}_e,\forall e\in\mathcal{E}, \quad (13f)\\ & 0\le\nu_k\le 1, && \forall k\in\mathcal{K}, \quad (13g)\end{aligned}$$

where α_k is the fixed weight associated with user k's rate. Problem (13)'s feasible set is defined by the fronthaul capacity constraint per BS (6) (non-convex, mixed-integer), the maximum computation capacity per EC or CC (12) (convex), the achievable rate per each user's common and private message (13a) and (13b), respectively (non-convex, fractional), the maximum transmit power constraint per BS (13c) and per UAV (13d) (convex), the delay constraint per CC's users (13e) and per ECs' users (13f) (non-convex, fractional), where t_k denotes the maximum tolerable delay per user, and the data-split constraint (13g) (convex). Due to the nature of the ℓ_0-norm, and the non-convexity of the feasible set, problem (13) is a mixed-integer non-convex optimization problem. As such problems are generally difficult to solve, our paper next proposes a series of problem reformulations so as to devise a numerically practical algorithm to solve problem (13).

III. Proposed Solutions

Given the numerical intricacies of problem (13), the paper now proposes a practically feasible algorithm, the highlight of which is its ability of being implemented in a distributed fashion across the CC and the ECs. Initially, the mixed-integer nature of constraints (6) and (13e) are tackled using a heuristic ℓ_0-norm approximation. Subsequently, SCA finds a convex upper-bound to tackle the non-convex nature of the relaxed fronthaul constraint, and we give an upper bound on the fronthaul delay (11). After introducing an auxiliary variable, FP decouples the nominator and denominator in (13a), (13b), (13e), and (13f). Each step introduces auxiliary variables, which either contribute to the set of optimization variables, or are updated in an outer loop. We next present each of the above optimization formulations in details.

A. User-to-BS Clustering

To tackle the discrete structure of the ℓ_0-norm in (6), we utilize a heuristic ℓ_1-norm approximation with fixed weights in each iteration as

$$\begin{aligned}\sum\nolimits_{k\in\mathcal{K}}&\beta_{b,k}^p\big\|\big\|\boldsymbol{w}_{b,k}^p\big\|_2^2\big\|_1 r_k^p\\ &+\beta_{b,k}^c\big\|\big\|\boldsymbol{w}_{b,k}^c\big\|_2^2\big\|_1 r_k^c\le R_b^{\max}, \qquad \forall b\in\mathcal{B}, \quad (14)\end{aligned}$$

where the weights are defined as $\beta_{b,k}^{\varphi}=\big(\delta+\big\|\tilde{\boldsymbol{w}}_{b,k}^{\varphi}\big\|_2^2\big)^{-1}$, with $\delta>0$ and $\tilde{\boldsymbol{w}}_{b,k}^{\varphi}$ denoting the beamforming vector from the last iteration [17]. Due to the inverse relationship between $\beta_{b,k}^{\varphi}$ and $\tilde{\boldsymbol{w}}_{b,k}^{\varphi}$, BS-user-links with low allocated power are likely to be deactivated, as high weights pose a burden on the limited-capacity fronthaul link. Naturally, only BS-user-links with reasonable allocated power remain active at the end of the algorithm. Such formulation enables the opportunity to introduce an auxiliary variable $q_{b,k}^{\varphi}=\beta_{b,k}^{\varphi}\big\|\big\|\boldsymbol{w}_{b,k}^{\varphi}\big\|_2^2\big\|_1$, so as to obtain a bilinear formulation of (14)'s left hand side as $q_{b,k}^{\varphi}r_k^{\varphi}$. Define the aggregate vector $\boldsymbol{q}^{\varphi}$ as $\boldsymbol{q}^{\varphi}=[q_{1,1}^{\varphi},\cdots,q_{1,K_c}^{\varphi},q_{2,1}^{\varphi},\cdots,q_{B,K_c}^{\varphi}]^T$. To be able to apply the SCA framework, e.g., see [15], write $q_{b,k}^{\varphi}r_k^{\varphi}$ as $q_{b,k}^{\varphi}r_k^{\varphi}\triangleq\frac{1}{4}((q_{b,k}^{\varphi}+r_k^{\varphi})^2-(q_{b,k}^{\varphi}-r_k^{\varphi})^2)$. Using the SCA approach enables us to replace its concave part, i.e., $-(q_{b,k}^{\varphi}-r_k^{\varphi})^2$, using the linear first-order Taylor expansion. The reformulated fronthaul capacity constraint (6) becomes

$$\begin{aligned}\sum_{k\in\mathcal{K}}&\big((q_{b,k}^p+r_k^p)^2-2(\tilde{q}_{b,k}^p-\tilde{r}_k^p)(q_{b,k}^p-r_k^p)+(\tilde{q}_{b,k}^p-\tilde{r}_k^p)^2\big)\\ &+\big((q_{b,k}^c+r_k^c)^2-2(\tilde{q}_{b,k}^c-\tilde{r}_k^c)(q_{b,k}^c-r_k^c)+(\tilde{q}_{b,k}^c-\tilde{r}_k^c)^2\big)\\ &\le 4R_b^{\max}, \qquad \forall b\in\mathcal{B}. \quad (15)\end{aligned}$$

Here $\tilde{q}_{b,k}^{\varphi}$ and $\tilde{r}_k^{\varphi}$ denote feasible fixed values, i.e., the optimal optimization variables of the previous iteration. Similar to $\beta_{b,k}^{\varphi}$, $\tilde{q}_{b,k}^{\varphi}$ and $\tilde{r}_k^{\varphi}$ are updated after each iteration.

Under the above mentioned ℓ_1-norm relaxation and SCA approach, we may reformulate the fronthaul delay (11), which is part of the total CC users' delay (13e), by

$$\Lambda_k=\max_{b\in\mathcal{B}_k}\left\{\frac{1}{R_b^{\max}}\sum_{i\in\mathcal{K}}\tilde{q}_{b,i}^p\nu_i D_i+\tilde{q}_{b,i}^c(1-\nu_i)D_i\right\}. \quad (16)$$

Note that (16) denotes the worst-case fronthaul delay, as it depends on the last iteration's clustering variables, which might include more users in a BS clustering set than the current iteration's solution. As an example, initially, all data is transmitted over every fronthaul link, posing a strict constraint, which is relieved with more iterations.

At this point, (15) is a convex constraint, (16) is a convex formulation, and problem (13)'s non-convexity stems from

the achievable rate constraints (13a) and (13b) and the delay constraints (13e) and (13f).

B. Auxiliary Variables

The highly coupled and fractional SINR expressions (2) and (3) within the concave logarithm in constraints (13a) and (13b) prevent from finding solutions to problem (13) efficiently. We, therefore, introduce an auxiliary variable for reformulating the complicated constraints (13a) and (13b) as $\boldsymbol{\gamma} = [\gamma_1^p, \ldots, \gamma_K^p, \gamma_{1,1}^c, \gamma_{1,2}^c, \ldots, \gamma_{K,K}^c]^T$. Note that $\gamma_{i,k}^c = 0$ $\forall i \notin \mathcal{I}_k, k \in \mathcal{K}$ as only a part of all users decode common message i. The reformulated problem, by including the auxiliary variable $\boldsymbol{\gamma}$ and the relaxed fronthaul constraint (15), is mathematically formulated as

$$\begin{aligned}
\max_{\boldsymbol{a},\boldsymbol{r},\boldsymbol{f},\boldsymbol{\nu},\boldsymbol{q},\boldsymbol{\gamma}} \quad & \sum\nolimits_{k\in\mathcal{K}} \alpha_k (r_k^p + r_k^c) && (17)\\
\text{s.t.} \quad & (12), (13c) - (13g), (15), \\
& \gamma_{i,k}^c \le \Gamma_{i,k}^c, && \forall i \in \mathcal{I}_k, \forall k \in \mathcal{K}, \ (17a)\\
& \gamma_k^p \le \Gamma_k^p, && \forall k \in \mathcal{K}, \ (17b)\\
& r_i^c \le \tau \log_2(1+\gamma_{i,k}^c), && \forall i \in \mathcal{I}_k, \forall k \in \mathcal{K}, \ (17c)\\
& r_k^p \le \tau \log_2(1+\gamma_k^p), && \forall k \in \mathcal{K}, \ (17d)\\
& \beta_{b,k}^p \left\| \left\| \boldsymbol{w}_{b,k}^p \right\|_2^2 \right\|_1 \le q_{b,k}^p, && \forall b \in \mathcal{B}, \forall k \in \mathcal{K}_c, \ (17e)\\
& \beta_{b,k}^c \left\| \left\| \boldsymbol{w}_{b,k}^c \right\|_2^2 \right\|_1 \le q_{b,k}^c, && \forall b \in \mathcal{B}, \forall k \in \mathcal{K}_c. \ (17f)
\end{aligned}$$

Note that (17c)-(17f) are convex constraints, and so constraints (13e), (13f), (17a) and (17b) remain the main reason for the non-convexity of problem (17), and are therefore tackled next in the text.

C. Fractional Programming

Based on the general FP framework developed in [46, Theorem 2], we apply the quadratic transform in multidimensional and complex form to reformulate (17a) and (17b) as

$$\begin{aligned}
g_{i,k}^c(\boldsymbol{a},\boldsymbol{\gamma}) &= \gamma_{i,k}^c - 2\mathrm{Re}\left\{ \left(u_{i,k}^c\right)^\dagger \left(\boldsymbol{a}_i^c\right)^\dagger \boldsymbol{h}_k \right\} && (18)\\
&+ |u_{i,k}^c|^2 \left[\sigma^2 + \sum_{j\in\mathcal{K}} \left| \boldsymbol{h}_k^\dagger \boldsymbol{a}_j^p \right|^2 + \sum_{l\in\mathcal{K}\setminus\mathcal{I}_k\cup\mathcal{I}_{i,k}'} \left| \boldsymbol{h}_k^\dagger \boldsymbol{a}_l^c \right|^2 \right], \\
g_k^p(\boldsymbol{a},\boldsymbol{\gamma}) &= \gamma_k^p - 2\mathrm{Re}\left\{ \left(u_k^p\right)^\dagger \left(\boldsymbol{a}_k^p\right)^\dagger \boldsymbol{h}_k \right\} && (19)\\
&+ |u_k^p|^2 \left[\sigma^2 + \sum_{j\in\mathcal{K}\setminus\{k\}} \left| \boldsymbol{h}_k^\dagger \boldsymbol{a}_j^p \right|^2 + \sum_{l\in\mathcal{K}\setminus\mathcal{I}_k} \left| \boldsymbol{h}_k^\dagger \boldsymbol{a}_l^c \right|^2 \right],
\end{aligned}$$

where $\boldsymbol{u} = [u_1^p, \cdots, u_K^p, u_{1,1}^c, u_{1,2}^c, \ldots, u_{K,K}^c]^T$ is a vector of complex-valued auxiliary variables. Note that (18) and (19) are convex functions of the beamforming vectors, in case $\boldsymbol{u}$ is fixed. The optimal $u_{i,k}^c$ and u_k^p for fixed $\boldsymbol{a}$ can be written as

$$(u_{i,k}^c)^* = \frac{\left(\boldsymbol{a}_i^c\right)^\dagger \boldsymbol{h}_k}{\sigma^2 + \sum\limits_{j\in\mathcal{K}} \left| \boldsymbol{h}_k^\dagger \boldsymbol{a}_j^p \right|^2 + \sum\limits_{l\in\mathcal{K}\setminus\mathcal{I}_k\cup\mathcal{I}_{i,k}'} \left| \boldsymbol{h}_k^\dagger \boldsymbol{a}_l^c \right|^2}, \quad (20)$$

$$(u_k^p)^* = \frac{\left(\boldsymbol{a}_k^p\right)^\dagger \boldsymbol{h}_k}{\sigma^2 + \sum\limits_{j\in\mathcal{K}\setminus\{k\}} \left| \boldsymbol{h}_k^\dagger \boldsymbol{a}_j^p \right|^2 + \sum\limits_{l\in\mathcal{K}\setminus\mathcal{I}_k} \left| \boldsymbol{h}_k^\dagger \boldsymbol{a}_l^c \right|^2}, \quad (21)$$

respectively. Such results are obtained by setting the partial derivative of $g_{i,k}^c(\boldsymbol{a},\boldsymbol{\gamma})$ and $g_k^p(\boldsymbol{a},\boldsymbol{\gamma})$ with respect to u_k^c and u_k^p to zero and then solving for u_k^c and u_k^p, respectively.

In a similar manner, utilizing the quadratic transform from [46, Theorem 1] constraint (13e) can be formulated as

$$\begin{aligned}
\tilde{\Theta}_k^{\mathrm{cc}}(\boldsymbol{f},\boldsymbol{r},\boldsymbol{\nu}) = \max\Bigg\{ & \frac{F_k}{f_k} + \Lambda_k + 2\theta_k^p \sqrt{\nu_k D_k} - (\theta_k^p)^2 r_k^p \ , \\
& \max_{k'\in\mathcal{M}_k} \left\{ \frac{F_{k'}}{f_{k'}} + \Lambda_{k'} + \frac{D_{k'}}{r_{k'}^c} - 2\theta_{k'}^c \sqrt{\nu_{k'} D_{k'}} + (\theta_{k'}^c)^2 r_{k'}^c \right\} \Bigg\}, \quad (22)
\end{aligned}$$

where $\boldsymbol{\theta} = [\theta_1^p, \cdots, \theta_K^p, \theta_1^c, \ldots, \theta_K^c]^T$ is a vector of complex-valued auxiliary variables. Note that the same transformation in (22) holds for the edge users' delay constraint (13f), except for the fronthaul delay terms Λ_k, which are set to zero for $\tilde{\Theta}_k^{\mathrm{ec}}(\boldsymbol{f},\boldsymbol{r},\boldsymbol{\nu})$. Without loss of generality, we let $\hat{\nu}_k = \sqrt{\nu_k}$ and the new optimization variable is then $\hat{\boldsymbol{\nu}} = [\hat{\nu}_1, \cdots, \hat{\nu}_K]$. Thereby, constraint (22) becomes a convex function of the optimization variables, when we fix $\boldsymbol{\theta}$. Optimal values of these auxiliary variables are

$$(\theta_k^\varphi)^* = \frac{\hat{\nu}_k \sqrt{D_k}}{r_k^\varphi}, \quad (23)$$

for $\varphi \in \{p, c\}$.

D. Problem Convexification

1) Problem Reformulation: Under all previously mentioned reformulations and manipulations, the original problem (13) is approximated by the following computationally tractable optimization problem

$$\begin{aligned}
\max_{\boldsymbol{a},\boldsymbol{r},\boldsymbol{f},\hat{\boldsymbol{\nu}},\boldsymbol{q},\boldsymbol{\gamma}} \quad & \sum\nolimits_{k\in\mathcal{K}} \alpha_k (r_k^p + r_k^c) && (24)\\
\text{s.t.} \quad & (12), (13c), (13d), (15), (17c) - (17f), \\
& g_{i,k}^c(\boldsymbol{a},\boldsymbol{\gamma}) \le 0, && \forall i \in \mathcal{I}_k, \forall k \in \mathcal{K}, \ (24a)\\
& g_k^p(\boldsymbol{a},\boldsymbol{\gamma}) \le 0, && \forall k \in \mathcal{K}. \ (24b)\\
& \tilde{\Theta}_k^{\mathrm{cc}}(\boldsymbol{f},\boldsymbol{r},\boldsymbol{\nu}) \le t_k, && \forall k \in \mathcal{K}_c, \ (24c)\\
& \tilde{\Theta}_k^{\mathrm{ec}}(\boldsymbol{f},\boldsymbol{r},\boldsymbol{\nu}) \le t_k, && \forall k \in \mathcal{K}_e, \forall e \in \mathcal{E}, \ (24d)\\
& 0 \le \hat{\nu}_k^2 \le 1, && \forall k \in \mathcal{K}. \ (24e)
\end{aligned}$$

As the objective (24) is linear and the constraints define a convex set, problem (24) is a convex optimization problem, which can be solved efficiently [47]. More specifically, the solution is now computed in an iterative manner, via alternating between solving problem (24) and updating the auxiliary variables $\boldsymbol{u}$, $\boldsymbol{\theta}$, the weights $\beta_{b,k}^\varphi$, and the feasible fixed values $(q_{b,k}^\varphi)'$ and $(r_k^\varphi)'$, $\varphi \in \{p, c\}$, $\forall b \in \mathcal{B}$, $\forall e \in \mathcal{E}$, $\forall k \in \mathcal{K}$. The detailed steps of the fully centralized protocol (FCP) for resource management are presented in Algorithm 1 below.

Algorithm 1 Centralized Protocol for Resource Management

1: Initialize $\boldsymbol{a}$ and $\hat{\boldsymbol{\nu}}$ to feasible values
Repeat: until convergence
2: Update $\beta_{b,k}^\varphi$, $\tilde{q}_{b,k}^\varphi$, and $\tilde{r}_k^\varphi$; update $(u_{i,k}^c)^*$ and $(u_k^p)^*$ using (20) and (21); update $(\theta_k^\varphi)^*$ using (23)
3: Solve convex optimization problem (24)
4: **End**

Protocol	Central cloud	Edge cloud
FCP	$\mathcal{O}(V_{\max}(2K(4+EL_e+BL_c)+2BK_c)^{3.5})$	
PDP	$\mathcal{O}(V_{\max}^{\text{pdp}}(2K_c(4+B+BL_c))^{3.5})$	$\mathcal{O}(V_{\max}^{\text{pdp}}(2K_e(4+L_e))^{3.5})$

TABLE I
COMPUTATIONAL COMPLEXITIES.

2) Complexity of the Fully Centralized Solution: The above FCP's complexity depends on (a) the convergence rate of Algorithm 1, i.e., the maximum number of iterations, and (b) the complexity of problem (24). The convex problem (24) can be solved using an interior-point method, where the total number of variables is $\xi_{\text{FCP}} = 2K(4+EL_e+BL_c)+2BK_c$ [48]. Hence, the upper-bound computational complexity of the FCP is $\mathcal{O}(V_{\max}(\xi_{\text{FCP}})^{3.5})$, where $V_{\max}$ is the worst-case number of iterations. Note that, for fixed clustering, high-quality beamforming vectors and computation capacities can be found with reduced computational complexity.

E. Decentralized Resource Management

The paper now illustrates how the proposed algorithm above is amenable for distributed implementation in the considered hybrid CC/MEC network, with a reasonable amount of information exchange. Note that initially, the CSIT has to be known at the CC and all ECs. This is reasonable, as, in practice, the UAVs are employed by the network operator under exchanging control information during UAV-operation. Also, in this work, the RS-CMD operations, e.g., common message decoding sets, are done on a per-CC or per-EC basis, enabling the decentralized operation. The distribution of optimization variables towards their respective management entity, i.e., CC or ECs, becomes then feasible, except for constraint (24a) and (24b), which require the CC and ECs to exchange the following terms iteratively

$$\sum_{j\in\mathcal{K}}\left|\boldsymbol{h}_k^\dagger\boldsymbol{a}_j^p\right|^2+\sum_{l\in\mathcal{K}\setminus\mathcal{I}_k\cup\mathcal{I}'_{i,k}}\left|\boldsymbol{h}_k^\dagger\boldsymbol{a}_l^c\right|^2, \tag{25}$$

$$\sum_{j\in\mathcal{K}\setminus\{k\}}\left|\boldsymbol{h}_k^\dagger\boldsymbol{a}_j^p\right|^2+\sum_{l\in\mathcal{K}\setminus\mathcal{I}_k}\left|\boldsymbol{h}_k^\dagger\boldsymbol{a}_l^c\right|^2, \tag{26}$$

respectively. As we assume full CSIT, the CC and ECs are able to efficiently compute (25) and (26) and then forward a single value to all other entities. A procedure for decentralized resource management builds upon Algorithm 1, which is implemented at the CC and each EC and adds an additional step between step 3 and step 4: *Exchange* (25) *and* (26) *between CC and ECs*.

1) Complexity of the Partially Decentralized Protocol: Using the partially decentralized protocol (PDP), the computational complexity is substantially reduced and distributed among CC and ECs. The computation complexity at the CC becomes $\mathcal{O}(V_{\max}^{\text{pdp}}(\xi_{\text{PDP}})^{3.5})$, and at each EC $\mathcal{O}(V_{\max}^{\text{pdp}}(2K_e(4+L_e))^{3.5})$, where $\xi_{\text{PDP}} = 2K_c(4+B+BL_c)$ and $V_{\max}^{\text{pdp}}$ is the worst-case number of iterations. For ease of presentation, Table I summarizes the two protocols' complexities.

IV. SIMULATIONS

To assess the performance of the proposed algorithms, we consider a cellular network of B BSs, each equipped with 2 antennas, and with 400m inter-BS distance. At the network edge, we consider E UAVs, each equipped with two antennas, and placed at 100m altitude. The CC users are placed randomly within the network, while $2\cdot E$ users are placed at the cell edge in groups of two, so each UAV would serve two users simultaneously, with minimum distance 75m and maximum distance 200m. For illustration, we set the noise power-spectral density to -124dBm/Hz, the fronthaul capacity to $R_b^{\max} = 40$Mbps, $Q_e = 100$W, $F_k = 10^8$cycles, $D_k = 10^5$bits, $f_c^{\max} = 5\cdot 10^{10}$cycles/s, $f_e^{\max} = 10^9$cycles/s, $P_b^{\max} = 24$dBm, $P_e^{\max} = 20$dBm, $\tau = 10$MHz, $t_k = 750$ms, $\alpha_k = 1$, $D = 2$, and $\eta = 0.4$, $\forall k\in\mathcal{K}, b\in\mathcal{B}$, and $e\in\mathcal{E}$, unless mentioned otherwise. The beamforming vectors are initialized as maximum ratio transmitters (MRT) and the initial data-split is $\nu_k = 0.8$, $\forall k\in\mathcal{K}$. In this work, we utilize the 3GPP specified pathloss $p_{k,n}(\text{dB}) = 128.1+37.6\log_{10}(d_{k,n})$, with $d_{k,n}$ being the distance between user and BS in km, and the UAV-user channel pathloss modeled in [49], with the pathloss component $\eta_{\text{los}} = 5$, and $\eta_{\text{nlos}} = 35$. The adopted channel model consists of Rayleigh fading ($\mathcal{CN}(0,1)$). Similar to [38], we set $s_e = 10^{-28}$ and $\mu_e = 3$, utilizing a specific CPU model.

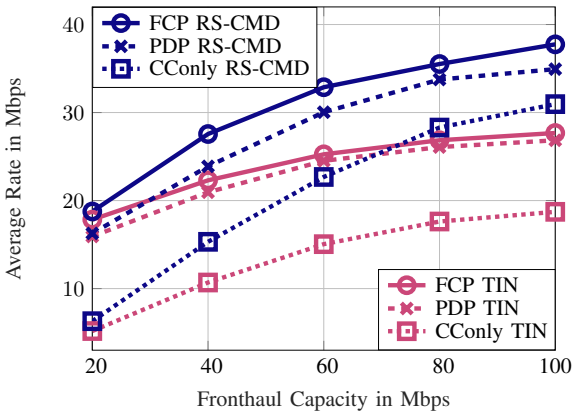


Fig. 3. Average rate over the fronthaul capacity for the FCP and PDP RS-CMD and TIN, additionally we consider the CConly ($E=0$) special case.

In this work, we propose specific methods, which we refer to as FCP RS-CMD and PDP RS-CMD, denoting the centralized and distributed implementation of Algorithm 1 respectively. Additionally, we consider the TIN strategy as the benchmarking scheme [50], whenever only private messages are considered. We note that solving our problem under TIN strategy is indeed a specific special case of Algorithm 1, where all common message decoding capabilities and the RS are deactivated. From a receiver complexity perspective, our TIN version is less complex than RS-CMD, i.e., compared to Fig. 2, TIN has no SIC steps.

A. Impact of the Fronthaul Capacity

In the first set of simulations, we provide insights into the behavior of the proposed algorithm by plotting the average rate as a function of the fronthaul capacity $R_b^{\max}$. We let $B = 4$, $K = 14$, and $E = 2$ and show the average user rate in Fig. 3. While considering FCP and PDP for RS-CMD and TIN in the adopted hybrid CC/MEC scheme, in these simulations, we also consider both RS-CMD and TIN transmission schemes in

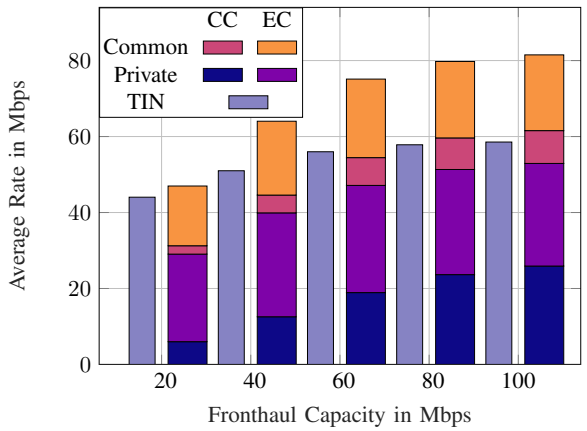


Fig. 4. Sum-rate of the PDP RS-CMD scheme partitioned into the sum CC users' and the sum of all EC users' private and common messages.

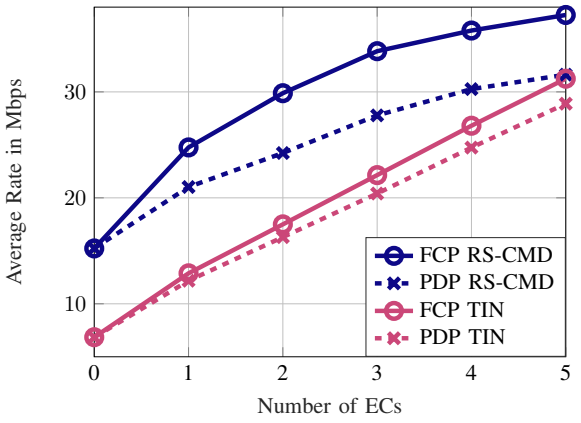


Fig. 5. Average rate over the number of ECs for the FCP and PDP under RS-CMD and TIN.

the presence of a central cloud only, which are referred to in Fig. 3 as central cloud only RS-CMD (CConly RS-CMD) and central cloud only TIN (CConly TIN), respectively.

Fig. 3 particularly shows how our RS-CMD proposed scheme outperforms the TIN scheme for all values of the fronthaul capacity. The figure also shows that the rates naturally increase as the fronthaul capacity increases, due to the higher throughput possibilities on the BSs' fronthaul links. However, we observe a saturation of the average rate for all schemes. This is especially due to the increased levels of interference originating from the CC at high fronthaul capacities. Such saturation is, however, most distinctive for the TIN schemes, which provide inferior interference mitigation capabilities as compared to RS-CMD. That is, RS-CMD can more efficiently manage the interference even when $R_b^{\max}$ is large, and thus outperforms TIN significantly.

Another observation stemming from Fig. 3 relates to the comparison of the proposed schemes and the schemes without MEC, i.e., CConly RS-CMD and CConly TIN. It is obvious that the joint CC and EC setup yields enhanced rates; however, at higher values of $R_b^{\max}$, CConly RS-CMD performs close to both RS-CMD schemes. This is reasonable in view of the strong impact of the CC at high fronthaul capacities. As a note regarding the decentralized implementation of Algorithm 1, it is observed that PDP performs very close to FCP in this set of simulations. Given the reduced complexity of PDP, the results in Fig. 3 further illustrate our proposed decentralized algorithm's practicality for providing reasonable performance as compared to FCP, which makes it amenable for future practical implementation.

We now shift the focus to the PDP schemes and highlight further details on the rate performance of this proposed scheme. More specifically, we show the average rate per user by breaking the rates down into the CC users' private rate, all ECs users' private rate, and the respective common counterparts versus the fronthaul capacity in Fig. 4 both for the PDP RS-CMD and the PDP TIN. Fig. 4 particularly shows that both the private and common rates increase jointly as the fronthaul capacity increases. The RS-CMD scheme is particularly shown in Fig. 4 to outperform the TIN scheme, especially at high fronthaul levels. As illustrated earlier, this is due to the fact that the high-fronthaul region necessitates particularly effective interference mitigation techniques in order to provide such high rates, which makes our proposed RS-CMD scheme especially suited for such scenarios given its strong capabilities at interference mitigation. As a last observation from Fig. 4, we note a less obvious impact of the parameter $R_b^{\max}$ on the ECs rate. To better illustrate such impact, consider the following rates: For $R_b^{\max} = 20$Mbps, the average private rate of an EC user is 23.05Mbps and the average common rate is 15.72Mbps. In contrast, at 60Mbps fronthaul capacity, these values increase to 28.21Mbps and 20.67Mbps for private and common rate, respectively. An explanation for such behavior is that in the low fronthaul-region, single BS-user links are clustered more prominently. To achieve the same SINR, such single BS-user link needs more transmit power compared to multiple BS-one user links. Thus, low fronthaul results in more transmit power and more interference at the network edge. As a result, even if the effect has only a minor impact under the current parameter set, the fronthaul capacity parameter has an impact on the edge users, which shows that the CC and EC are indeed coupled.

With increased rates at high fronthaul capacities, sophisticated interference management in terms of common message decoding becomes more important. The results in both Fig. 3 and Fig. 4 indeed validate the emphasized interference mitigation capabilities of the RS-CMD scheme, the close performance of PDP to FCP, and the coupling of cloud and edge common and private rates.

B. Impact of the Number of ECs

In the next set of simulations, we set $K = 16$, $B = 3$, and vary the number of ECs from $E = 0$ to $E = 5$, where for $E = 0$, all the 16 users are set to be CC users, regardless of their location. We consider RS-CMD and TIN for the FCP and the PDP. Fig. 5 shows the average rate of all active users in Mbps over the number of ECs.

As a first observation from Fig. 5, we note that for each scheme the average user rate increases with E. Compared to a network without MEC, the FCP with RS-CMD and $E = 5$ shows 144.7% gain in terms of average rates, and the FCP with TIN has even 356.7% rate improvement. That is, the proposed hybrid CC/MEC scheme is able to achieve enormous gains in terms of average rate compared to a non-MEC setup based on the CC only. Such gain is in essence due to the additional

communication and computation resources in the network, and due to the enhanced availability of service conditions, i.e., edge users can be served more efficiently and more reliably. Secondly, both the FCP and PDP under RS-CMD outperform the TIN schemes significantly. Especially for low number of ECs, FCP with RS-CMD achieves average rate more than double the TIN performance. But even at $K = 5$, a rate difference of about 7Mbps is observed. In fact, the enhanced interference management opportunities of RS-CMD yield the higher rate gains under the considered network setup. Interestingly, in Fig. 5, the rate growth of RS-CMD lessens with more ECs, while such behavior can hardly be observed for the TIN schemes. This is reasonable, as the proposed system model considers a common message decoding separation of CC and all ECs. That is, RS-CMD may unleash its full potential of interference mitigation in fully centralized cloud setups, whereas the performance degrades when less opportunities of interference management are available. Nevertheless, RS-CMD is still able to provide substantially higher average rates even in decentralized setups. Fig. 5 further shows that for both TIN and RS-CMD, FCP outperforms PDP in each point after $E = 0$. While this observation emphasizes the need for centralized resource management for optimal performance, we note that reasonable rates may as well be achieved by the decentralized protocol, which is much more practical in terms of implementation, complexity, and computation time, as also illustrated later in Section IV-E.

To conclude the above, we note that Fig. 5 underlines the auspicious performance gains of the joint CC/MEC system model, especially with increasing number of ECs, emphasizes the eminent advantages of RS-CMD over TIN in terms of average rate, and shows that the FCP yields superior rates, yet the PDP provides reasonable performance, given its superior practicality.

C. Available Power at the UAVs

The system model considered in this work strongly depends on the MEC capabilities provided by UAVs, which are specifically limited in their power consumption. Therefore, in the next set of simulations, we consider different available power levels at the UAVs for $E = 3$, $B = 3$, and $K = 14$. We show the average rate of all active users in Mbps versus the available transmit and computation power at the UAVs, i.e., $P_e^{\max} - Q_e$ in dBm, in Fig. 6.

Fig. 6 particularly shows that the parameter $P_e^{\max}$ has a significant impact on the user rates. For the FCP, rate improvements of 47.5% for RS-CMD and 35.8% for TIN can be observed when increasing $P_e^{\max} - Q_e$ from 16dBm to 24dBm. Thus, RS-CMD, particularly, benefits from the increased UAV power as both computation and transmission variables become less constrained in terms of power limits. More transmission power especially aids the RS-CMD as such enhancement can significantly benefit the interference mitigation capabilities, e.g., more power may be assigned to common messages.

Similar to previous results, in Fig. 6, RS-CMD outperforms TIN regardless of the implemented protocol, i.e., FCP and PDP. Such a result emphasizes the gains on the RS paradigm

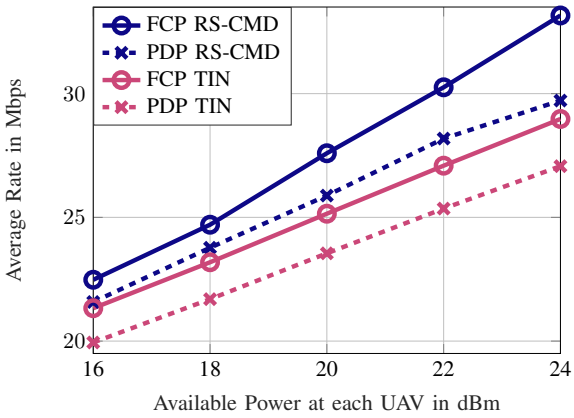


Fig. 6. Average rate over the available transmit and computation power at the UAVs, i.e., $P_e^{\max} - Q_e$.

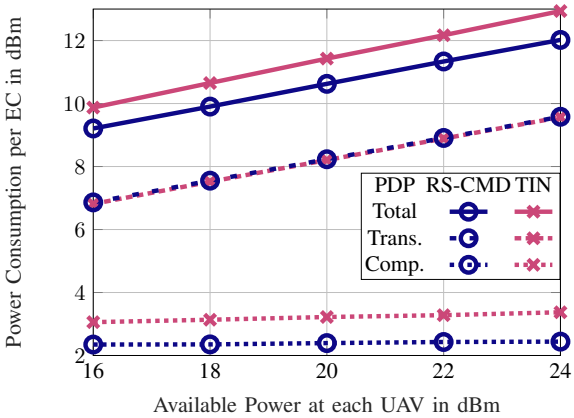
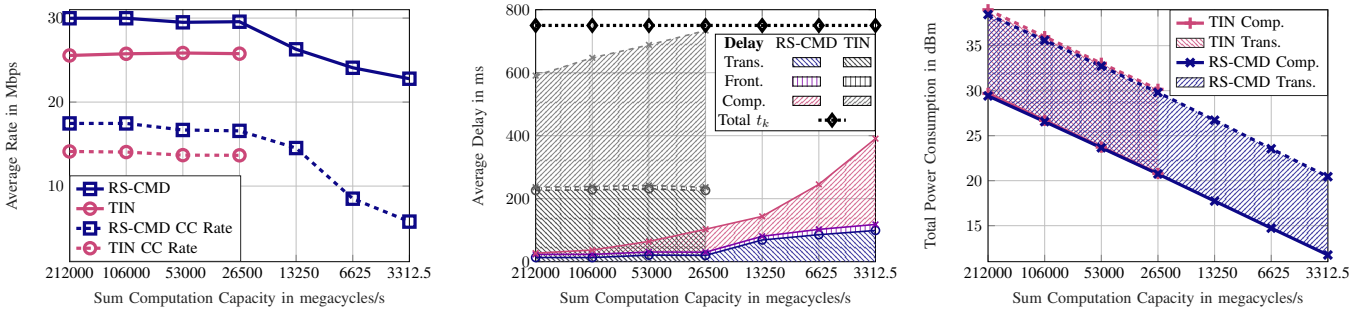


Fig. 7. Power consumption (total, computation, and transmission) over the available transmit and computation power at the UAVs, i.e., $P_e^{\max} - Q_e$.

for achieving higher data rates and utilizing the available computation and communication power resources. Comparing the FCP and the PDP, Fig. 6 outlines a reasonable gap in terms of the average rate. Especially when the UAVs have low powers, PDP RS-CMD has a marginal rate loss against its centralized counterpart, which highlights the favorable performance of the proposed scheme. In other words, with reduced complexity and runtime, PDP RS-CMD is able to achieve close to the FCP in terms of average data rate in practical setups.

Next, we illustrate the numerical dynamics of the EC power metric expression P_e^{ec} (8), especially those related to the computation and communication power consumption, while implementing the PDP RS-CMD. Note that we do not show the operational fixed powers Q_e, as they remain constant. Fig. 7 plots the power consumption per EC versus $P_e^{\max} - Q_e$ in dBm, i.e., the available transmit and computation power at the UAVs. In Fig. 7, it is obvious that the total (net operational) power consists of transmit and computation power, whereas the transmission power yields higher values than the computation power. Given the objective function (24), i.e., a weighted sum-rate, such a result is reasonable, as the algorithm prefers allocating higher data rates, rather than allocating more computational resources. Such behavior becomes even more clear through the change of UAV power in Fig. 7, i.e., the total power consumption increases jointly with the transmission power consumption for larger UAV power.



(a) Average rate. (b) Average delay. (c) Total power consumption.

Fig. 8. Average rate, average delay, and total power consumption over the sum computation capacity for the FCP RS-CMD and FCP TIN schemes.

To compare RS-CMD to TIN, we note that in Fig. 7, RS-CMD achieves lower total power consumption as compared to TIN. While the transmit power levels are almost equal (RS-CMD has a marginally increased power consumption), the main difference originates from the computational power. RS-CMD is able to allocate less computational resources, similar transmit power resources, and in total less power as compared to TIN, all while providing higher data rates. This is especially interesting from the perspective that RS-CMD employs double the number of streams as TIN.

The results presented in the above simulation set emphasize the superior performance of RS-CMD both in terms of average rates and power consumption in a wide range of UAV power levels. In particular, at practical low power levels, the PDP implementation of RS-CMD performs close to the centralized solution, all whilst having a lower complexity and runtime. This underlines the suitability of the PDP and RS-CMD for practical B5G systems.

D. Impact of Computation Capacities

For $K = 14$, $E = 3$, and $B = 3$, we now vary the computation capacities in constraint (12). In more detail, initially, we start with $f_c^{\max} = 20 \cdot 10^{10}$cycles/s, $f_e^{\max} = 4 \cdot 10^9$cycles/s and then for each point on the x-axis divide the previous values for $f_c^{\max}$ and $f_e^{\max}$ by two. In Fig. 8a, the average rate is shown as a function of the sum computation capacity. Note that the FCP TIN scheme becomes infeasible for values greater than 26500megacycles/s, therefore these values are omitted. Fig. 8a shows that low computation capabilities severely deteriorate the average rate of each user. This effect is, in particular, pronounced when observing the average rates of CC users. Such values fall to almost 5Mbps for 3312.5megacycles/s. Thus, based on the paper objective, the parameters $f_c^{\max}$ and $f_e^{\max}$ greatly affect the performance of RS-CMD. On the same dataset, Fig. 8b depicts the average delay per user in ms over the computation capacity. Specifically, in Fig. 8b, the reason for FCP TIN's infeasibility becomes visible. At the first four points on the x-axis, TIN's average delay increases severely to almost the total tolerated delay t_k. Further decreasing the computation capacity, hence, leads to the infeasibility of constraint (10). On the other hand, FCP RS-CMD is able to keep a comparably low average delay per user in Fig. 8b. Even at the lowest value of the illustrated computation capacity, the average delay is slightly above half the tolerated delay t_k. Interestingly, with decreasing computation capacities, both the transmission delay and the computation delay increase. While the growth of computation delay seems obvious while decreasing the computation capabilities at both the CC and the ECs, the transmission delay growth requires further elaboration. As the computation capabilities decrease, the delays of all users naturally increase as a consequence. Due to the coupling of the delays among users, i.e., by the common message decoding, such delay increases may be problematic at strict t_k values. Hence, the algorithm shifts towards a more equal distribution of all resources so as to meet each user's delay requirement. As one consequence, each user might be allocated more transmission resources, e.g., transmit power, but the average transmission delay rises, especially with strict delay constraints. Fig. 8c further shows the total power consumption versus the computation capacities. While RS-CMD achieves higher average rates with slightly less power consumption that TIN, we note that the power consumption of both schemes decreases jointly with the computation capacity. Fig. 8c illustrates that the transmission powers remain almost constant, but the computation power consumption decreases significantly. While at 53000megacycles/s, the computation power makes up a large portion of the total power consumption. Fig. 8c also shows that at 3312.5megacycles/s, the total power consumption is almost equally partitioned into transmission and computation power.

Fig. 8a-8c illustrate the significant impact of the computation capacity network parameter on the overall system performance in terms of average rate, average delay, and power consumption. The results show the appreciable feasibility of FPC RS-CMD over TIN and the enhanced numerical performance prospects of RS-CMD. RS-CMD is shown to be well suited for a wide range of computation capabilities, which makes it an amenable scheme for practical hybrid CC/MEC networks.

E. Computation Time and Scalability Analysis

In this subsection, we investigate the runtime values of different schemes and the scalability, which are both particularly relevant to practical B5G resource management schemes. Firstly, over the number of ECs, we provide normalized runtime values for the FCP and PDP RS-CMD schemes and

Number of ECs E	1	2	3	4	5
FCP RS-CMD	1	1.02	1.04	1.06	1.10
PDP RS-CMD	0.90	0.78	0.73	0.60	0.50
PDP TIN	0.58	0.53	0.45	0.39	0.31

TABLE II
NORMALIZED RUNTIME VALUES OVER E.

Number of users K	8	10	12	14	16	18
RS-CMD ($\eta = 0.65$)	1	1.73	2.18	2.80	3.48	4.42
RS-CMD ($\eta = 0.35$)	1	1.66	2.08	2.53	3.18	4.27
TIN	1	1.45	1.92	2.30	2.75	3.05

TABLE III
SCALABILITY AS NORMALIZED RUNTIME VALUES OVER K FOR THE PDP.

the PDP TIN scheme in Table II. All runtime values are normalized to the FCP RS-CMD with 1 EC enabled, and parameters are chosen according to Section IV-B. For the FCP RS-CMD, we observe a reasonable increase in runtime as the number of ECs increases. While more UAVs introduce more opportunities for optimization, more variables, and more interference, which needs to be taken care of carefully, it is reasonable that the runtime increases for the centralized scheme. Contrary to this, both the PDP RS-CMD and the PDP TIN start with lower runtime values at $E = 1$ to begin with, and both experience decreased runtime values with increasing number of ECs. Such behavior is particularly relying on the decentralized nature of the PDP, which distributes the network variables to the CC and all ECs. As the number of ECs increases and all other parameters are kept fixed, the computational burden of the CC is reduced by each added EC. Therefore, while the FCP increases its runtime jointly with the parameter E, the PDP RS-CMD has only 50% runtime at $E = 5$ as compared to its centralized counterpart at $E = 1$. Comparing RS-CMD and TIN, it is noteable that TIN achieves lower runtime than the RS-CMD scheme. However, RS-CMD decreases its relative runtime by 40%, while TIN by 27%, from $K = 1$ to $K = 5$. The relief of the computational complexity by distributing the optimization variables is thus more effective for the RS-CMD, as it naturally includes a larger set of variables.

Secondly, the scalability versus the number of users, i.e., the runtime values normalized to each scheme's runtime at $K = 8$, for the PDP RS-CMD and TIN is depicted in Table III. We consider additionally the common message decoding parameter η, and adopt the parameters from Section IV-C. The results show that TIN yields better scalability as compared to both RS-CMD schemes. Specifically, when doubling the number of users, TIN has a relative runtime of 2.75, while RS-CMD ($\eta = 0.35$) and RS-CMD ($\eta = 0.65$) have 3.18 and 3.48, respectively. Doubling the number of users thus almost triples the TIN's runtime, and more than triples the RS-CMD runtime values. However, since TIN employs only K streams, while RS-CMD employs $2{\cdot}K$ streams, which are even partially decoded at multiple users, such scalability increases are put into perspective. Thus, we note that the proposed RS-CMD scheme for hybrid CC/MEC networks yields reasonable scalability for increasing numbers of users. Another aspect of Table III is the reduced runtime of RS-CMD ($\eta = 0.35$) as compared to the reference scheme with $\eta = 0.65$. It can be concluded that carefully choosing the parameter η is especially important in this paper system model.

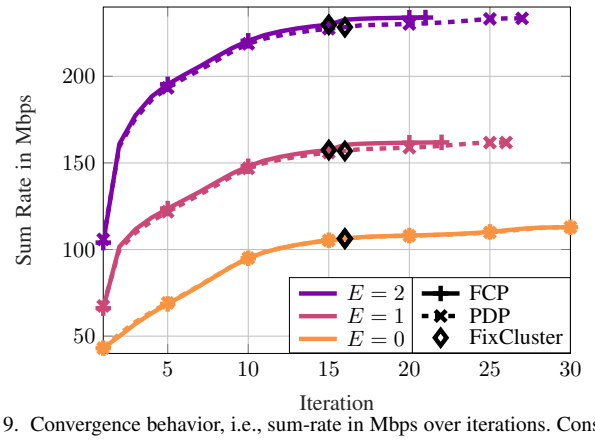


Fig. 9. Convergence behavior, i.e., sum-rate in Mbps over iterations. Consider RS-CMD for different number of ECs, where FixCluster refers to the iteration, wherefrom the clustering is kept fixed.

F. Convergence Behavior

Lastly, with the same parameters as in Section IV-B, we numerically validate the convergence behavior of the proposed algorithms, i.e., the centralized and decentralized implementation of Algorithm 1. Fig. 9 shows the sum-rate over the iterations for the FCP and the PDP for different number of ECs. Thereby, Fig. 9 depicts the rapid convergence behavior of the proposed solutions. Comparing different numbers of ECs, we note that, similar to previous observations, schemes with more ECs converge faster due to the distribution of optimization variables. The decentralized nature of the PDP has only minor impact on the convergence behavior, but yields slightly more number of iterations. However, as per the last subsection discussions, the PDP iterations reduce the runtime values due to the decentralized implementation manner, which highlights the numerical prospects of the proposed algorithm in the context of RS-CMD applications to future hybrid CC/MEC networks.

V. CONCLUSION

Facing a vast amount of connections, huge performance demands, and the need for reliable connectivity, future wireless networks are envisioned to implement disruptive technologies that jointly spur connectivity, performance, and reliability. To this end, this paper proposes, and evaluates the benefit of, one particular hybrid CC/MEC platform, especially introduced to balance the network resources required for joint computation and communication. The transmission process is further empowered by splitting the users' messages into private and common parts, specifically introduced to mitigate the interference. This work particularly focuses on maximizing the weighted sum-rate subject to fronthaul capacity, maximum computation capacity, achievable rates, power consumption, delay, and data-split constraints, so as to determine the optimal allocated rates, private and common beamforming vectors, computational capacities, and data-split factors. Thanks to ℓ_0-norm relaxation, SCA, and FP, different algorithms are proposed to solve the intricate optimization problem. The impact of network parameters on the physical resources is then illustrated in the simulation results, which highlight the prospects of the proposed algorithms for enabling efficient joint communication and computation, especially the

appreciable improvements in throughputs, data processing, transmission delay, and power consumption as compared to conventional system strategies. At strong interference levels, our proposed distributed algorithm, i.e., PDP RS-CMD, particularly achieves reasonable gains and superior runtime advantages, emphasizing its applicability in the context of RS-CMD applications to future hybrid CC/MEC networks.

APPENDIX A
DETERMINING COMMON MESSAGE SETS

Finding the global optimal common message sets for each user to decode is a challenging task, especially as an exhaustive search over all possible combinations would be needed. In this paper, based on results from [6], [51], we instead propose a numerically practical, yet heuristic, algorithm to determine the common messages to be decoded by a network user. Our approach is inspired by information theoretical works on the optimally of treating interference as noise, e.g., [52]–[54]. Such theoretical findings suggest that the best scheduling strategy for wireless links sharing the same resources is as follows: The links are divided into independent groups and only one group of links is active in a specific communication slot. These groups are determined in such a way that each link meets a specified TIN criterion. Links which do not meet such criteria are usually scheduled to different resource blocks. However, in this work, instead of scheduling wireless links among which TIN is not optimal to other resource blocks, we apply RS-CMD to this group of interfering links. Hence, Algorithm (2) aims at determining two groups of users associated with each user k. More specifically, for each user $k \in \mathcal{K}$, the first group receiving interference from user k does not decode k's common message as the interference is below a certain threshold. The second group of users which suffer from strong interference of user k are set to decode k's common message.

In more details, similar to [6], [51], the adopted algorithm to determine the set of common messages utilizes two parameters, the maximum number of SIC steps allowed per user D and the parameter η, which determines the percentage of users being chosen for common message decoding. In practice, D may be chosen by the network operator and the available user hardware, but is likely chosen to be a small integer. The intuition behind such a choice is as follows: When a user l decodes the common message of user k, user l enjoys less level of interference as it decodes part of it; however, the common rate of user k is subject to additional constraints and thus may negatively impact the overall sum-rate. Hence, to balance these two effects, we use the design parameters η and D. The detailed steps for determining the common message sets $\mathcal{I}_k$ for each user are given in Algorithm 2. The procedure utilizes the available CSIT and computes MRT beamforming vectors and the achievable rates, e.g., see (5) for TIN. Initially, each user is set to decode its own common message. Then, for each computing instance, i.e., for either the CC or each EC, Algorithm 2 checks which users are the "weak" users according to their achievable rate. In more details, further common messages to decode are only assigned

Algorithm 2 Set To-Be-Decoded Common Messages $\{\mathcal{I}_k\}_{k=1}^{K}$

Obtain CSIT; intialize MRT beamforming vectors and achievable rates
Let $\mathcal{I}_k = k \ \forall k \in \mathcal{K}$
For: $i \in \{c, \mathcal{E}\}$ **do**
  For: $\{k \in \mathcal{K}_i | R_k$ within the η-percentile of rates$\}$ **do**
    $\hat{\mathcal{K}} = \mathcal{K}_i \backslash \{k\}$
    Determine interference power $\{\Xi_{k,j}\}_{j \in \hat{\mathcal{K}}}$
    While: $|\mathcal{I}_k| < D$ and $\hat{\mathcal{K}} \neq \emptyset$ **do**
      $\mathcal{I}_k = \mathcal{I}_k \cup \{\underset{j \in \hat{\mathcal{K}}}{\text{argmax}}\ \Xi_{k,j}\}$
      $\hat{\mathcal{K}} = \hat{\mathcal{K}} \backslash \{\underset{j \in \hat{\mathcal{K}}}{\text{argmax}}\ \Xi_{k,j}\}$
    End while
  End for
End for

to users within the η-th percentile of the users' rates connected to this instance. For each of these "weak" users, the procedure determines the interference levels induced by the candidate users, i.e., all other users connected to the current computing instance. As long as the maximum number of SIC steps is not yet reached and the candidate user set is not empty, each "weak" user is assigned to decode its strongest interferer. This interferer is then removed from the candidate user set. Given Algorithm 2's output, i.e., $\mathcal{I}_k$, $\forall k \in \mathcal{K}$, the set $\mathcal{M}_k$ can be determined for each user, whereby the decoding order is set according to the channel strengths. We lastly note that Algorithm 2 is heuristic as the interference threshold is solely determined based on the effective channel response and does not involve beamforming optimization. The numerical results, however, show the effectiveness and superiority of our design in the context of RS-CMD applicability to our hybrid CC/MEC system model.

REFERENCES

[1] "Ericsson mobility report june 2022," Ericson, Tech. Rep., Jun. 2022. [Online]. Available: https://www.ericsson.com/en/reports-and-papers/mobility-report/reports/june-2022

[2] P. Popovski, K. F. Trillingsgaard, O. Simeone, and G. Durisi, "5G wireless network slicing for eMBB, URLLC, and mMTC: A communication-theoretic view," *IEEE Access*, vol. 6, pp. 55 765–55 779, 2018.

[3] Y. Mao, O. Dizdar, B. Clerckx, R. Schober, P. Popovski, and H. V. Poor, "Rate-splitting multiple access: Fundamentals, survey, and future research trends," *IEEE Commun. Surv. Tutor.*, pp. 1–1, 2022.

[4] A. Carleial, "Interference channels," *IEEE Trans. Inf. Theory*, vol. 24, no. 1, pp. 60–70, 1978.

[5] T. Han and K. Kobayashi, "A new achievable rate region for the interference channel," *IEEE Trans. Inf. Theory*, vol. 27, no. 1, pp. 49–60, 1981.

[6] A. A. Ahmad, H. Dahrouj, A. Chaaban, A. Sezgin, and M.-S. Alouini, "Interference mitigation via rate-splitting and common message decoding in cloud radio access networks," *IEEE Access*, vol. 7, pp. 80 350–80 365, 2019.

[7] H. Dahrouj and W. Yu, "Multicell interference mitigation with joint beamforming and common message decoding," *IEEE Trans. Commun.*, vol. 59, no. 8, pp. 2264–2273, 2011.

[8] Y. Mao, B. Clerckx, and V. O. Li, "Rate-splitting multiple access for downlink communication systems: bridging, generalizing, and outperforming SDMA and NOMA," *EURASIP J Wirel. Commun. Netw.*, 2018.

[9] F. Zhou, R. Q. Hu, Z. Li, and Y. Wang, "Mobile edge computing in unmanned aerial vehicle networks," *IEEE Wirel. Commun.*, vol. 27, no. 1, pp. 140–146, 2020.

[10] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Commun. Surv. Tutor.*, vol. 19, no. 4, pp. 2322–2358, 2017.

[11] H. Joudeh and B. Clerckx, "Sum-rate maximization for linearly precoded downlink multiuser MISO systems with partial CSIT: A rate-splitting approach," *IEEE Trans. Commun.*, vol. 64, no. 11, pp. 4847–4861, 2016.

[12] ——, "Robust transmission in downlink multiuser MISO systems: A rate-splitting approach," *IEEE Trans. Signal Process.*, vol. 64, no. 23, pp. 6227–6242, 2016.

[13] E. Piovano, H. Joudeh, and B. Clerckx, "Generalized degrees of freedom of the symmetric cache-aided MISO broadcast channel with partial CSIT," *IEEE Trans. Inf. Theory*, vol. 65, no. 9, pp. 5799–5815, 2019.

[14] T. Q. Quek, M. Peng, O. Simeone, and W. Yu, *Cloud radio access networks: Principles, technologies, and applications*. Cambridge University Press, 2017.

[15] R.-J. Reifert, A. A. Ahmad, H. Dahrouj, A. Chaaban, A. Sezgin, T. Y. Al-Naffouri, and M.-S. Alouini, "Distributed resource management in downlink cache-enabled multi-cloud radio access networks," *IEEE Trans. Veh. Technol.*, 2022.

[16] L. Liu and W. Yu, "Cross-layer design for downlink multihop cloud radio access networks with network coding," *IEEE Trans. Signal Process.*, vol. 65, no. 7, pp. 1728–1740, Apr. 2017.

[17] B. Dai and W. Yu, "Sparse beamforming and user-centric clustering for downlink cloud radio access network," *IEEE Access*, vol. 2, pp. 1326–1339, 2014.

[18] N. Ha, W. Shin, M. Vaezi, and H. V. Poor, "Coordinated rate splitting multiple access for multi-cell downlink networks," in *54th ACSSC*, 2020, pp. 996–1001.

[19] A. Mishra, Y. Mao, L. Sanguinetti, and B. Clerckx, "Rate-splitting assisted massive machine-type communications in cell-free massive MIMO," *IEEE Commun. Lett.*, vol. 26, no. 6, pp. 1358–1362, 2022.

[20] Z. Lin, M. Lin, T. de Cola, J.-B. Wang, W.-P. Zhu, and J. Cheng, "Supporting IoT with rate-splitting multiple access in satellite and aerial-integrated networks," *IEEE Internet Things J.*, vol. 8, no. 14, pp. 11 123–11 134, 2021.

[21] D. Yu, J. Kim, and S.-H. Park, "An efficient rate-splitting multiple access scheme for the downlink of C-RAN systems," *IEEE Wirel. Commun. Lett.*, vol. 8, no. 6, pp. 1555–1558, 2019.

[22] A. A. Ahmad, Y. Mao, A. Sezgin, and B. Clerckx, "Rate splitting multiple access in C-RAN: A scalable and robust design," *IEEE Trans. Commun.*, vol. 69, no. 9, pp. 5727–5743, 2021.

[23] R.-J. Reifert, A. A. Ahmad, Y. Mao, A. Sezgin, and B. Clerckx, "Rate-splitting multiple access in cache-aided cloud-radio access networks," *Front. Comms. Net.*, vol. 2, 2021.

[24] A. A. Ahmad, H. Dahrouj, A. Chaaban, T. Y. Al-Naffouri, A. Sezgin, J. S. Shamma, and M.-S. Alouini, "Power minimization using rate splitting with statistical CSI in cloud-radio access networks," *Front. Comms. Net.*, vol. 2, 2021.

[25] R.-J. Reifert, S. Roth, A. A. Ahmad, and A. Sezgin, "Energy efficiency in rate-splitting multiple access with mixed criticality," in *IEEE ICC Workshops*, 2022, pp. 681–686.

[26] Y. Mao, B. Clerckx, and V. O. K. Li, "Rate-splitting for multi-antenna non-orthogonal unicast and multicast transmission: Spectral and energy efficiency analysis," *IEEE Trans. Commun.*, vol. 67, no. 12, pp. 8754–8770, 2019.

[27] A. A. Ahmad, B. Matthiesen, A. Sezgin, and E. Jorswieck, "Energy efficiency in C-RAN using rate splitting and common message decoding," in *IEEE ICC Workshops*, 2020, pp. 1–6.

[28] G. Arora and A. Jaiswal, "Zero SIC based rate splitting multiple access technique," *IEEE Commun. Lett.*, pp. 1–1, 2022.

[29] Y. S. Reddy, G. Chopra, A. Dubey, A. Kumar, T. Panigrahi, and L. Reddy Cenkeramaddi, "Rate-splitting random access mechanism for massive machine type communications in 5G cellular internet-of-things," in *IEEE 32nd PIMRC*, 2021, pp. 648–653.

[30] N. Abbas, Y. Zhang, A. Taherkordi, and T. Skeie, "Mobile edge computing: A survey," *IEEE Internet Things J.*, vol. 5, no. 1, pp. 450–465, 2018.

[31] X. Diao, J. Zheng, Y. Wu, Y. Cai, and A. Anpalagan, "Joint trajectory design, task data, and computing resource allocations for NOMA-based and UAV-assisted mobile edge computing," *IEEE Access*, vol. 7, pp. 117 448–117 459, 2019.

[32] F. Zhou, Y. Wu, R. Q. Hu, and Y. Qian, "Computation rate maximization in UAV-enabled wireless-powered mobile-edge computing systems," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 9, pp. 1927–1941, 2018.

[33] Q. Zhang, J. Chen, L. Ji, Z. Feng, Z. Han, and Z. Chen, "Response delay optimization in mobile edge computing enabled UAV swarm," *IEEE Trans. Veh. Technol.*, vol. 69, no. 3, pp. 3280–3295, 2020.

[34] T. Zhang, Y. Xu, J. Loo, D. Yang, and L. Xiao, "Joint computation and communication design for UAV-assisted mobile edge computing in IoT," *IEEE Trans. Ind. Inform.*, vol. 16, no. 8, pp. 5505–5516, 2020.

[35] H. Liu, Y. Ye, Z. Bai, K. J. Kim, and T. A. Tsiftsis, "Rate splitting multiple access aided mobile edge computing in cognitive radio networks," in *IEEE ICC Workshops*, 2022, pp. 598–603.

[36] R. Han, Y. Wen, L. Bai, J. Liu, and J. Choi, "Rate splitting on mobile edge computing for UAV-aided IoT systems," *IEEE Trans. Cogn. Commun. Netw.*, vol. 6, no. 4, pp. 1193–1203, 2020.

[37] T. P. Truong, N.-N. Dao, and S. Cho, "HAMEC-RSMA: Enhanced aerial computing systems with rate splitting multiple access," *IEEE Access*, vol. 10, pp. 52 398–52 409, 2022.

[38] Z. Yang, C. Pan, K. Wang, and M. Shikh-Bahaei, "Energy efficient resource allocation in UAV-enabled mobile edge computing networks," *IEEE Trans. Wirel. Commun.*, vol. 18, no. 9, pp. 4576–4589, 2019.

[39] W. Feng, J. Tang, N. Zhao, X. Zhang, X. Wang, K.-K. Wong, and J. A. Chambers, "Hybrid beamforming design and resource allocation for UAV-aided wireless-powered mobile edge computing networks with NOMA," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 11, pp. 3271–3286, 2021.

[40] A. A. Ahmad, J. Kakar, R.-J. Reifert, and A. Sezgin, "UAV-assisted C-RAN with rate splitting under base station breakdown scenarios," in *IEEE ICC Workshops*, 2019, pp. 1–6.

[41] J. Kakar, A. Chaaban, V. Marojevic, and A. Sezgin, "UAV-aided multi-way communications," in *IEEE PIMRC*, 2018, pp. 1169–1173.

[42] X. Cao, J. Xu, and R. Zhang, "Mobile edge computing for cellular-connected UAV: Computation offloading and trajectory optimization," in *IEEE SPAWC*, 2018, pp. 1–5.

[43] W. Jaafar, S. Naser, S. Muhaidat, P. C. Sofotasios, and H. Yanikomeroglu, "On the downlink performance of RSMA-based UAV communications," *IEEE Trans. Veh. Technol.*, vol. 69, no. 12, pp. 16 258–16 263, 2020.

[44] S. K. Singh, K. Agrawal, K. Singh, and C.-P. Li, "Outage probability and throughput analysis of UAV-assisted rate-splitting multiple access," *IEEE Wirel. Commun. Lett.*, vol. 10, no. 11, pp. 2528–2532, 2021.

[45] J. Xu, K. Ota, M. Dong, and H. Zhou, "MCTS-enhanced hybrid offloading for aerial multi-access edge computing," *IEEE Wirel. Commun.*, vol. 28, no. 5, pp. 82–87, 2021.

[46] K. Shen and W. Yu, "Fractional programming for communication systems part I: Power control and beamforming," *IEEE Trans. Signal Process.*, vol. 66, no. 10, pp. 2616–2630, May 2018.

[47] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 2.1," 2014. [Online]. Available: http://cvxr.com/cvx

[48] M. Lobo, L. Vandenberghe, S. P. Boyd, and H. Lebret, "Applications of second-order cone programming," *Linear Algebra and its Applications*, vol. 284, pp. 193–228, 1998.

[49] A. Al-Hourani, S. Kandeepan, and S. Lardner, "Optimal LAP altitude for maximum coverage," *IEEE Wireless Commun. Lett.*, vol. 3, no. 6, pp. 569–572, Dec 2014.

[50] B. Bandemer, A. Sezgin, and A. Paulraj, "On the noisy interference regime of the MISO gaussian interference channel," in *42nd ACSSC*, 2008, pp. 1098–1102.

[51] A. A. Ahmad, "Resource and interference management in cloud radio access networks," doctoral thesis, Ruhr-Universität Bochum, Universitätsbibliothek, 2021. [Online]. Available: https://doi.org/10.13154/294-8313

[52] N. Naderializadeh and A. S. Avestimehr, "ITLinQ: A new approach for spectrum sharing in device-to-device communication systems," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1139–1151, 2014.

[53] X. Yi and G. Caire, "Optimality of treating interference as noise: A combinatorial perspective," *IEEE Trans. Inf. Theory*, vol. 62, no. 8, pp. 4654–4673, 2016.

[54] K. Shen and W. Yu, "FPLinQ: A cooperative spectrum sharing strategy for device-to-device communications," in *IEEE ISIT*, 2017, pp. 2323–2327.