Response to Reviews of IEEE Journal on Selected Areas in Communications #1570803135

Joint Communication and Computation Offloading for Ultra-Reliable and Low-Latency with Multi-tier Computing

Addressed Comments for Publication as An Original Paper on Special Issue "Multi-Tier Computing for Next Generation Wireless Networks"

Dang Van Huynh, Van-Dinh Nguyen, Symeon Chatzinotas,
Saeed R. Khosravirad, H. Vincent Poor, and Trung Q. Duong

August 26, 2022

August 26, 2022

Dear Guest Editors,

Thank you very much for handling the review process of our paper. We would also like to thank the Anonymous Reviewers for their constructive comments on the previous manuscript version, which have been very helpful in improving the quality of our manuscript.

To address the Reviewers' comments, we have carefully revised the paper to enhance the quality of its content and the clarity of its exposition. In the following, we provide a point-by-point response and detail our revisions to address these comments. Unless stated otherwise, all the numbered items (figures, equations, references, etc.) in this response letter refer to those in the revised manuscript.

We hope that the paper is now suitable for publication.

Yours sincerely,

Dang Van Huynh, Van-Dinh Nguyen, Symeon Chatzinotas, Saeed R. Khosravirad, H. Vincent Poor, and Trung Q. Duong

Note: To help legibility of the remainder of this response letter, all the Reviewers' comments and questions are typeset in *italic font.* Our responses and remarks are written in plain font. The updated texts in the revised manuscript are typeset in blue.

IEEE Journal on Selected Areas in Communications
Paper No. #1570803135
Authors' Responses to Reviewer 1's Comments

We would like to thank the reviewer for valuable and constructive comments and suggestions. We have revised the paper in line with the Reviewer's comments, thereby improving the contributions and the clarity of the paper accordingly.

General Comments:

This paper considers the joint communications and computing design with ultra-reliability and low-latency communications (URLLC). Specifically, the users can share their task for processing with edge server and cloud server to reduce the computing load on the users. The communications protocol is considered as URLLC for satisfying some critical constraints in QoS and makes the system more suitable to mission-critical services which is strong point. The computing model is multier computing, i.e., users to edge servers to cloud, where the latency of computation for each steps is taken into account.

The optimization problem is to minimize the worse-case user latency with respect to several variables including the offloading paramters, the users association to edge servers, and the power allocation. The proposed algorithm for the optimisation problem has been shown to work well through the numerical results with some nice illustration on the performance. Over all, the paper falls into the scope of the special, quite nicely written and organise (although there are rooms for improving), and has some interesting results. The Reviewer has several comments in the following sections that require some revisions.

Overall the paper has no major problems, perhaps some minor weaknesses that can be addressed to improve the quality of the paper. Some points should be clearer. Check the comments for detailed improvement for the improvement . The authors are strongly encouraged to apply the changes, especially focusing on the interpretation of the results. I think it is important to explain the physical meaning of the results what and why?

Response:

We thank the Reviewer for the positive comment on the contribution of our paper.

Comment 1:

Please check the Abstract, make it more tractable, for example, "intractable problem" should be "intractable"

Response:

Thank you for the comment. We have corrected it.

Comment 2:

The Introduction should be revised. Some parts are not consistent, "low-latency transmis-

sion", "low latency communications", which one?
"the fifth generation mobile network (5G)" →"the fifth generation (5G) networks"
"edge devices"? What is this "device" at the edge? Do you mean "edge server" Strongly recommend the authors to rewrite the literature review in passive form. It is quite irritating to read "Xiao et al." or "The authors in [8]". For a technical paper (engineering paper), try to avoid writing personal thing
I saw several inconsistency as well, for example, "delay-sensitive" "delay-efficiency"?

Response:
Thank you very much for the constructive comment. We have corrected all these typos. In addition, we also have updated some parts of the literature review in the passive form.

> Page 2-3, Section I.A
> In particular, a novel offload forwarding scheme was proposed in [8], where fog servers (FSs) cooperate with each other to tackle their heterogeneousness in terms of computation capacity and resources, improving the efficiency of power usage. In [9], a reformulation-linearization-technique-based Branch-and-Bound (BnB) method was developed to minimize the energy consumption of end devices by jointly optimizing the offloading selection, radio and computation resource allocation. The results in [10] showed that joint transmission energy allocation and task allocation design can significantly reduce the total energy consumption.

Comment 3:
The numerical results section should be improved. There are some no-meaning interpreted sentences. For example, "The e2e latency ... increases to 30 gigacycles", or "increasing from 30 to 34 gigacycles", or "decreases from 0.34 s to around 0.32s". These "absolute values" do not tell you anything. Unless it is higher or lower than something.
Response:
Thank you very much for the constructive comment. We fully agree with the reviewer that the interpretation in these sentences do not provide the added value and we have removed all of them in the revised manuscript.

IEEE Journal on Selected Areas in Communications
Paper No. #1570803135
Authors' Responses to Reviewer 2's Comments

We would like to thank the reviewer for valuable and constructive comments and suggestions. We have revised the paper in line with the reviewer's comments, thereby improving the contributions and the clarity of the paper accordingly.

General comment:

This paper studied a joint optimization of communication and computation resources to minimize the end-to-end latency of computational tasks among multiple IoT devices in hierarchical edge-cloud systems with ultra-reliable and low latency communications. A novel approach based on the alternating optimization and inner approximation framework is developed to solve the formulated nonconvex problem. In addition, suboptimal solutions are also proposed to reduce the computational complexity. Extensive numerical results are provided to verify the effectiveness of the proposed approaches in terms of the convergence speed and the fairness e2e latency.

This paper formulated a very general optimization problem which comprehensively takes into account affects of communication and computation, such as offloading probabilities, processing rates, user association policies and power control.

A simple yet efficient iterative algorithm by leveraging the alternating optimization (AO) approach and inner approximation (IA) framework is developed to effectively solve the challenging optimization problem in an iterative manner. More interestingly, a novel penalty function is introduced to parameterize the JCCO problem, which helps improve the convergence speed of the AO-IA algorithm.

The authors also proposed two sub-optimal designs, such as best channel selection and random user association, which can achieve comparable performance with much lower complexity.

The Reviewer did not see significant negatives of the work, but there are some points should be clarified as follows:
1. To motivate the work, the authors mentioned that the straggler effect is a major bottleneck in implementing computation offload in hierarchical edge cloud systems. The actual meaning of the "straggler effect" should be further elaborated in this context.
2. There is missing a detailed explanation of equation (2). A toy example (for example with $\pi_{mk} = 0, 1$) could help.
3. The overall e2e latency in (11) includes 5 different latencies. Which one causes the

major latency?

Response:

We thank the reviewer for carefully reading our manuscript and summarising the major contributions of our paper. We also have carefully revised the manuscript as following responses of detailed comments.

Comment 1:

To motivate the work, the authors mentioned that the straggler effect is a major bottleneck in implementing computation offload in hierarchical edge cloud systems. The actual meaning of the "straggler effect" should be further elaborated in this context.

Response:

Thank you very much for the constructive comment. As discussed in the system model, there is a significant heterogeneity of the offloaded portions, transmission time, and ESs' computing capacity to execute a task, creating straggler effect in the obtained e2e latency. Therefore, to reduce this effect and improve the performance, we apply the maximum operator, i.e., max(.) in the the ESs' processing latency and fronthaul transmission latency equations.

Comment 2:

There is missing a detailed explanation of equation (2). A toy example (for example with $\pi_{mk} = 0, 1$*) could help.*

Response:

Thank you very much for this helpful comment. The detailed explanation of user association variables ($\boldsymbol{\pi}$) is already given at the description of "User layer" in Section II-A. In particular, $\pi_{mk} = 1$ means the k-th ES admits tasks from the m-th UE; and $\pi_{mk} = 0$, otherwise. We assume that the tasks of a UE is only offloaded to one ES, i.e., $\sum_{k \in \mathcal{K}} \pi_{mk} = 1, \forall m$.

Comment 3:

The overall e2e latency in (11) includes 5 different latencies. Which one causes the major latency?

Response:

Thank you very much for this constructive comment. These five different latencies can be categorised into two types, such as the communication latency and computation latency. We have numerically observed that the wireless transmission latency is the major source of the overall e2e latency. This reflects practical scenarios, where the wireless transmission is affected by many factors, e.g. channel conditions, transmit power and locations of devices, while the computation capacity of UEs, ESs and CS are typically large enough to execute tasks rapidly.

To address this comment, we have added a paragraph after eq. (11) to discuss the overall e2e latency.

Page 10, Section II.C:

There are five parts of the overall e2e latency that can be classified into two categories, including communication latency and computation latency. Typically, the wireless transmission latency is the major source of the overall e2e latency. This reflects practical scenarios, where the wireless transmission is affected by many factors, e.g., channel conditions, transmit power and locations of devices, while the computation capacity of UEs, ESs and CS are large enough to execute tasks rapidly.

Comment 4:

Algorithm 1 requires an initial feasible point to start the iterative algorithm. The authors are suggested to discuss an effective way to general such initial feasible point.

Response:

Thank you very much for this constructive comment. We have added a new paragraph to describe the key step to generate initial points of Algorithm 1.

Page 19, Section III-C.1.

Algorithm 1 requires initial feasible points to start at the first iteration. The initial feasible points of $\mathcal{S}_1^{(0)}$ and $\mathcal{S}_2^{(0)}$ are generated as follows. Firstly, the transmission power and the processing rate are randomly generated with respect to constraint (13g). Secondly, the offloading portions and user association variables are initiated equally, i.e., $\alpha_m = 0.5$ and $\pi_{mk} = 0.5, \forall m, k$. Finally, we implement a validating function to guarantee that all constraints in (14) are satisfied before running the optimisation algorithm.

Comment 5:

The reviewer wonders how Algorithm 2 with a penalty function can guarantee an exact binary solution for π. Is there an exact binary solution or at least nearly exact binary solution at convergence?

Response:

Thank you very much for this constructive comment. The penalty function is introduced to not only speed up the convergence but also force the relaxed values of $\boldsymbol{\pi}$ to close to binary ones at convergence. From Fig. 3, it is clear that Algorithm 2 converges faster than Algorithm 1, and the latency gap of Algorithm 2 at the "recovery binary step" is much smaller than Algorithm 1 due to the use of penalty function.

IEEE Journal on Selected Areas in Communications
Paper No. #1570803135

Authors' Responses to Reviewer 3's Comments

We would like to thank the reviewer for valuable and constructive comments and suggestions. We have revised the paper in line with the reviewer's comments, thereby improving the contributions and the clarity of the paper accordingly.

General comment:
The authors have studied joint communication and computation offloading (JCCO) for hierarchical edge-cloud systems with ultra-reliable and low latency communications (URLLC). They aimed at minimizing the E2E latency of computational tasks among multiple industrial Internet of Things (IIoT) devices by jointly optimizing offloading probabilities, processing rates, user association policies and power control subject to their service delay and energy consumption requirements as well as queueing stability conditions.

The authors have studied joint communication and computation offloading for hierarchical edge-cloud systems with ultra-reliable and low latency communications. They aimed at minimizing the E2E latency of computational tasks among multiple industrial Internet of Things devices by jointly optimizing offloading probabilities, processing rates, user association policies and power control subject to their service delay and energy consumption requirements as well as queueing stability conditions. The paper is well written and technically solid. The studied problem is timely and important.

The authors have studied joint communication and computation offloading for hierarchical edge-cloud systems with ultra-reliable and low latency communications. They aimed at minimizing the E2E latency of computational tasks among multiple industrial Internet of Things devices by jointly optimizing offloading probabilities, processing rates, user association policies and power control subject to their service delay and energy consumption requirements as well as queueing stability conditions. The paper is well written and technically solid. The studied problem is timely and important. The novelties and differences with existing studies can be further clarified. And some technical contents can be explain in more details.

Response:
We thank the Reviewer for carefully reading our manuscript and nicely summarising the major contributions of our paper.

Comment 1:
The joint communication and computing resource maximization problem is very timely and

important. However, considering other existing works also dealing with similar research issue, it would be better if the authors can clarify the unique novelties, and and explain more on the difference with existing studies in terms of the studied research problem, for example, with one recent work "Joint RAN Slicing and Computation Offloading for Autonomous Vehicular Networks: A Learning-Assisted Hierarchical Approach", IEEE Open Journal of Vehicular Technology, 2021.

Response:

Thank you so much for this constructive comment. The novelty and difference of our work over existing studies are clearly discussed in Section I-A and I-B. In particular, to fully exploit benefits offered by a hierarchical edge-cloud system, there are still several formidable challenges that need to be tackled, including communication costs, heterogeneous computational capabilities of ESs as well as limited radio resources. Although ESs and FSs are often equipped with more powerful computing capability than end-users, they are quite limited compared to large-scale cloud data centers (CDCs) at the cloud server. The straggler effect is a major bottleneck in implementing computation offload in hierarchical edge-cloud systems. For example, an ES with limited computation capability admitting tasks from many users may increase processing latency, leading to higher e2e latency. All these challenges have not yet been fully addressed in the aforementioned works. In addition, a comprehensive analysis for the e2e latency model considering all factors of communication (including URLLC) and computation is not presented in [2], [23]. In constrast, this work proposes a novel joint communication and computation for URLLC-enabled hierarchical edge-cloud system, taking into account all the above issues.

Comment 2:

To solve the formulated MINLP, the authors propose to use the AO approach to solve the problem. From the Reviewer's point of view, AO may not give the optimal solutions (very likely suboptimal). The authors may explain how to evaluate how good the suboptimal solutions would be, in other words, the optimality gap.

Response:

Thank you very much for the insightful comment. As discussed in Section III-A, the problem JCCO (13) is a mixed-integer non-convex program, and the objective function is nonconcave and nonsmooth while constraints are non-convex. Thus, it is not practical to apply a direct application of the well-known Brute-force search (BFS) method by searching over all possible association policies, even for networks of small-to-medium size. In addition, the improved branch-and-bound algorithm (IBBA) method presented in [2] is also inapplicable as the relaxed problem of JCCO is still non-convex due the strong coupling between continuous and binary variables. To overcome this challenge, we first transform the original problem (13) into a more computational tractable form by bypassing the non-smooth objective function. Then, by exploiting the unique structure of the underlying problem, we decompose it into two subproblems. Finally, we leverage the combination of

AO method and IA framework to solve subproblems efficiently, which converge to at least a local optimal solution.

Comment 3:

The authors may further proofread the manuscript. For example, e2e should be E2E, etc.

Response:

Thank you for this kind suggestion. We have carefully proofread the manuscript and corrected all typos and grammatical errors.

Comment 4:

A figure showing the whole system model under consideration is preferred.

Response:

Thank you for this suggestion. We have added Fig. 1 to show the whole system model.

Page 7, Section II.A

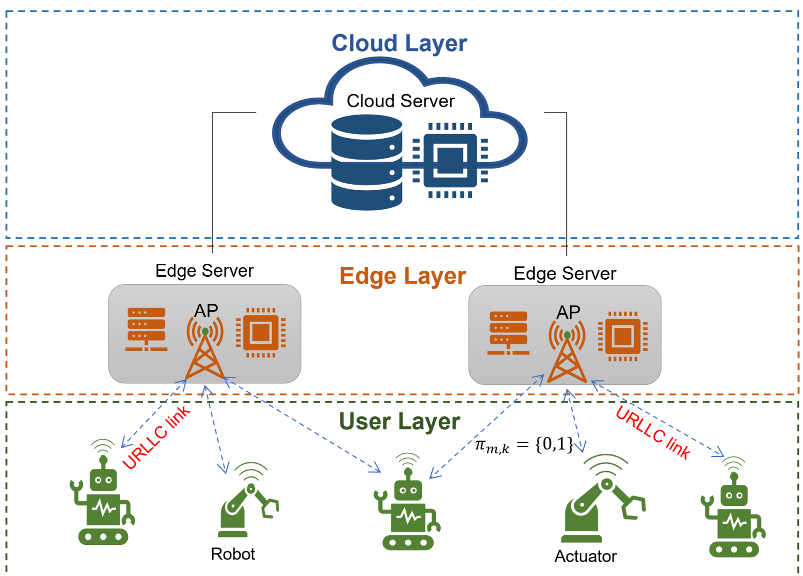


Figure 1: An URLLC-enabled hierarchical edge-cloud system model.

IEEE Journal on Selected Areas in Communications Paper No. #1570803135

Authors' Responses to Reviewer 4's Comments

We would like to thank the reviewer for valuable and constructive comments and suggestions. We have revised the paper in line with the reviewer's comments, thereby improving the contributions and the clarity of the paper accordingly.

General comment:

This paper considers a joint communication and computation resources optimisation for edge-cloud systems with URLLC. The addressed problem minimizes the end-to-end latency by jointly optimizing offloading portions, user associations, the processing rate of UEs and ESs, and transmission power of UEs subject to the queuing stability, resource budget of the system, and energy budget of UEs. The problem is solved by applying the alternating optimisation with convex approximations. The penalty objective function is additionally introduced to speed up user association and two sub-optimal designs are provided as benchmarks. Extensive numerical results demonstrate the effectiveness of the proposed solutions in minimizing the e2e latency.

The addressed optimization problem is well formulated which considers the critical communication technology (i.e., URLLC) in the multi-tier computing systems. This is practically attractive in the next generation of wireless applications. The proposed solutions are mathematically solid and the provided simulation results effectively validate the proposed solutions in various scenarios. Overall, the positives highly dominate the negatives in this paper.

There is no noticeable negative aspect of this paper. There are some minor comments that should be addressed to improve the clarity and presentation of current version. Full details have been provided in the comments below. However, they are minor compared with the positives and the main contributions of the paper.

Response:

We thank the Reviewer for carefully reading our manuscript and nicely summarising the major contributions of our paper.

Comment 1:

The motivation can be improved to emphasise the contributions in terms of investigating the queuing-aware multi-tier computing system.

Response:

Thank you so much for this constructive comment. We have carefully revised Section I-B to further emphasise the main contributions of the work, especially the queuing-aware latency.

> Page 4, Section I-B
>
> Although ESs and FSs are often equipped with more powerful computing capability than end users, they are still limited compared to large-scale cloud data centers (CDCs) at the cloud server. The straggler effect is a major bottleneck in implementing computation offload in hierarchical edge-cloud systems. For example, an ES with limited computation capability admitting tasks from many users with high arrival rate may increase processing latency, leading to higher e2e latency. All these challenges have not yet been fully addressed in the aforementioned works. In addition, a comprehensive analysis for the e2e latency model considering all factors of mission-critical communications (including URLLC) and queuing-aware computation is not presented in [2], [23].

Comment 2:

It is better to provide a figure which illustrates the system model with three layers of the end-edge-cloud architecture.

Response:

Thank you for this suggestion. In the reivised manuscriot, we have provided Fig. 1 to show the whole system model.

Comment 3:

In the communication model, the authors should further discuss the impact of the MF-SIC technique on the performance of the optimized solution. In addition, references of MF-SIC should be provided in this subsection.

Response:

Thank you very much for this constructive comment. We have provided a new reference [26] and discuss the impact of MF-SIC technique on the performance of the considered system. This technique has been widely used in the literature to improve the throughput of users with poorer channel conditions by SIC.

> Page 8, Section II-B
>
> To reduce information exchange between APs and the cloud server via fronthaul links, we adopt the matched filtering ($i.e., \hat{\mathbf{h}}_{mk}^H$) and successive interference cancellation, called the MF-SIC receiver for signal detection in the uplink [26], which only requires the local CSI of all UEs at each AP. To guarantee fairness among all UEs, we assume that the decoding order follows UEs' index by arranging the channel vectors as $\|\hat{\mathbf{h}}_{1k}\|^2 \geq \|\hat{\mathbf{h}}_{2k}\|^2 \geq, \cdots, \geq \|\hat{\mathbf{h}}_{Mk}\|^2, \forall k$. In

other words, AP k decodes the signals of UEs with better channel condition first and then remove them before decoding the signals of UEs with poorer channel conditions.

[26] L. Fang and L. Milstein, "Performance of successive interference cancellation in convolutionally coded multicarrier DS/CDMA systems," *IEEE Trans. Commun.*, vol. 49, no. 12, pp. 2062–2067, 2001.

Comment 4:
Why do we need (1-ω) in the uplink data rate equation (eq. 5)? Can the authors elaborate this expression?
Response:
Thank you very much for the constructive comment. As discussed in the uplink channel estimation part, the number of pilot symbols for uplink channel estimation should be at least equal to the number of UEs. In Eq. (5), we calculate $\omega = M/N$, where N is the blocklength. Therefore, the term (1-ω) indicates that the transmission rate given by Eq. (5) is used for the data rate of the m-th UE, where the rate of uplink training (i.e., channel estimation) is not considered.

Comment 5:
The authors consider a partial task offloading scheme for the system model, which is practically inefficient in real-world deployments. Can the author provide related work to support this idea?
Response:
Thank you very much for this helpful comment. We agree with the reviewer that the arbitrary partition for offloading may be challenging for practical deployments. However, according to [R1], the partial offloading is more suitable for the applications with more stringent latency requirement by taking advantages of parallel processing. There are many existing studies that have considered the partial task offloading for the edge computing, such as [R1]-[R3]. For instance, in [R2], the partial task offloading is applied for the offloading policy. The offload packet is offloaded to MEC server with probability $x \in [0, 1]$ and executed on the local device with probability $(1 - x)$. Furthermore, as mentioned in [R1], the partial offloading scheme can be modelled by offloading portion variables (i.e. $0 \leq \alpha \leq 1$). The full granularity is assumed in task partitioning. As a result, a task could be partitioned into sub-tasks with the appropriate sizes depending on service types and the network resource budgets.

[R1] Y. Wang, M. Sheng, X. Wang, L. Wang and J. Li, "Mobile-Edge Computing: Partial Computation Offloading Using Dynamic Voltage Scaling," in *IEEE Trans. Commun.*, vol. 64, no. 10, pp. 4268-4282, Oct. 2016.

[R2] R. Dong, C. She, W. Hardjawana, Y. Li, and B. Vucetic, "Deep learning for hybrid 5G services in mobile edge computing systems: Learn from a digital twin," *IEEE Trans. Wireless Commun.*, vol. 18, no. 10, pp. 4692–4707, Oct. 2019.

[R3] M.-H. Chen, M. Dong, and B. Liang, "Resource sharing of a computing access point for multi-user mobile cloud offloading with delay constraints," *IEEE Trans. Mobile Comput.*, vol. 17, no. 12, pp. 2868–2881, Dec. 2018.

Comment 6:
The authors provided the per-iteration complexity of the proposed algorithms. Can the authors further discuss the overall complexity of the algorithms?
Response:
Thank you very much for the constructive comment. In the revised manuscript, we have provided the analysis of the overall complexity as follows.

> Page 19, Section III-C:
>
> In term of the complexity analysis, in each iteration of Algorithm 1, the computational complexity of solving SP-2 dominates that of SP-1, especially in large-scale scenarios. Therefore, for a given number of I iterations that guarantee the convergence of Algorithm 1, its worst-case complexity is given as $\mathcal{O}\big(I\sqrt{9M+4MK+3K}(3MK+3M+4)^2\big)$.

Comment 7:
Since the solving process is quite complicated, it is better to add a flow diagram to illustrate the solving procedure.
Response:
Thank you very much for the constructive comment. We have added Fig.2 to better illustrate the optimisation procedure of the proposed algorithms.

> Page 23, Section IV:

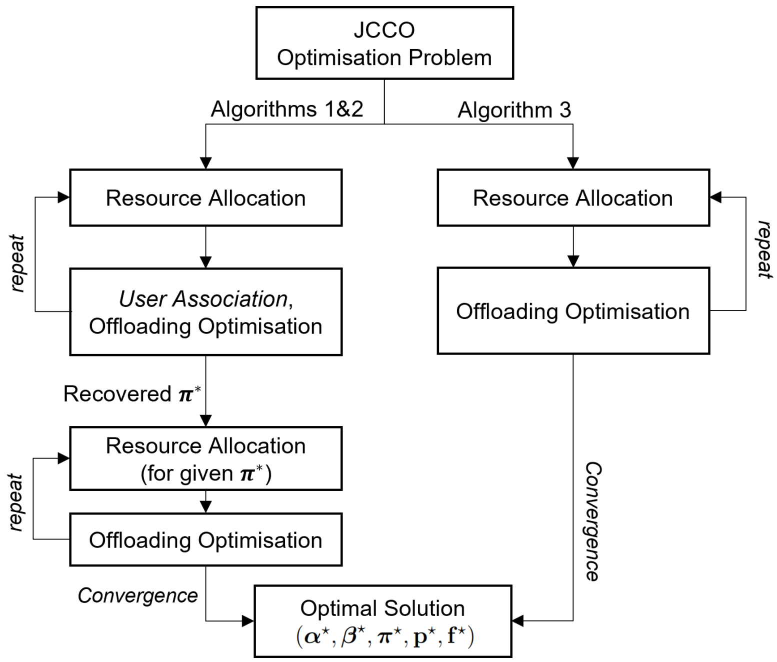


Figure 2: The optimisation procedure for solving the JCCO problem.

Comment 8:
In Fig. 1, it is more interesting to discuss why the difference of the e2e latency between Alg. 1 and Alg. 2 is so large after the recovering step (i.e., why is the increase of the latency in Alg. 1 after the recovering step so large compared to that in Alg. 2).
Response:
Thank you very much for this constructive comment. The penalty function is introduced to not only speed up the convergence but also force the relaxed values of $\boldsymbol{\pi}$ to close to binary ones at convergence. From Fig. 3, it is clear that Algorithm 2 converges faster than Algorithm 1, and the latency gap of Algorithm 2 at the "recovery binary step" is much smaller than Algorithm 1 due to the use of penalty function.

Comment 9:
In Fig. 2, why does the scheme "Fixed Freq." maintain the same level of the latency over the range of Emax?
Response:
As presented in Eq. (12), the processing rate of UEs impacts on the total energy consumption of UEs by contributing to the energy of computation. In the "Fixed Freq." scheme, the processing rate variables are fixed and set to be equal to initial values during

the optimising process. The initial values of the processing rate have to be satisfied with the lowest value of $E^{\max}$ in these simulations. Unsurprisingly, it maintains the same level of the optimised latency when $E^{\max}$ increases and the values of processing rate variables are unchanged.

Comment 10:

In terms of writing, there are still some errors in the paper (e.g., on page 6, $\forall m,.$", improper comma and full stop at the end of eq. (41) on page 16 and eq. (47b) on page 17, respectively). The authors should carefully proofread to correct these errors.

Response:

Thank you very much for the constructive comments. We have carefully proofread the manuscript and corrected all typos and grammatical errors.

Joint Communication and Computation Offloading for Ultra-Reliable and Low-Latency Multi-tier Computing

Dang Van Huynh, Van-Dinh Nguyen, Symeon Chatzinotas, Saeed R. Khosravirad, H. Vincent Poor, and Trung Q. Duong

Abstract

In this paper, we study joint communication and computation offloading (JCCO) for hierarchical edge-cloud systems with ultra-reliable and low latency communications (URLLC). We aim to minimize the end-to-end (e2e) latency of computational tasks among multiple industrial Internet of Things (IIoT) devices by jointly optimizing offloading probabilities, processing rates, user association policies and power control subject to their service delay and energy consumption requirements as well as queueing stability conditions. The formulated JCCO problem belongs to a difficult class of mixed-integer non-convex optimization problem, making it computationally intractable. In addition, a strong coupling between binary and continuous variables and the large size of hierarchical edge-cloud systems make the problem even more challenging to solve optimally. To address these challenges, we first decompose the original problem into two subproblems based on the unique structure of the underlying problem and leverage the alternating optimization (AO) approach to solve them in an iterative fashion by developing newly convex approximate functions. To speed up optimal user association searching, we incorporate a penalty function into the objective function to resolve uncertainties of a binary nature. Two sub-optimal designs for given user association policies based on channel conditions and random user associations are also investigated to serve as state-of-the-art benchmarks. Numerical results are provided to demonstrate the effectiveness of the proposed algorithms in terms of the e2e latency and convergence speed.

Index Terms

Alternating optimization, multi-tier computing, ultra-reliable and low latency communications, URLLC.

D. V. Huynh and T. Q. Duong are with Queen's University Belfast, UK. (e-mail: {dhuynh01,trung.q.duong}@qub.ac.uk). V.-D. Nguyen and S. Chatzinotas are with University of Luxembourg (e-mail: {dinh.nguyen, symeon.chatzinotas}@uni.lu). S. R. Khosravirad is with Nokia Bell Labs (e-mail: saeed.khosravirad@nokia-bell-labs.com). H. V. Poor is with Princeton University (e-mail: poor@princeton.edu).

This paper has been accepted in part for presentation at IEEE International Conference on Communications (ICC), Seoul, South Korea, May 2022.

Corresponding author is Trung Q. Duong (e-mail: trung.q.duong@qub.ac.uk).

I. INTRODUCTION

Recent advances in wireless communications and powerful computing platforms open new opportunities to enable various emerging and delay-sensitive applications that require low latency and energy consumption. Real-time monitoring and control is considered to be one of core pillars specified for the fifth generation (5G) networks, which allows implementing holographic communications, tactile Internet applications and telehealth applications. In order to guarantee real-time control and manage complex systems of autonomous devices in the industrial Internet of Things (IIoT), low latency communications and processing plays a vital role for holistic facilitation of such wireless networked systems. Ultra-reliable and low latency communications (URLLC) is defined in 3GPP Release 15, where the reliability requirement for transmitting a packet is $1 - 10^{-5}$ for 32 bytes with the user plane latency of 1 ms [1]. Recently, mobile-edge computing (MEC) with computing resources deployed at the network edge is considered as a promising solution to provide powerful computational ability and improved energy efficiency for battery-powered mobile devices [2].

Task offloading is the key enabler in hierarchical edge-cloud systems, allowing computation tasks to be partially executed at both IIoT devices (or user equipments (UEs) for short) and edge servers (ESs), thus minimizing the overall execution time [3]. In addition, resource-intensive computational tasks of edge servers can be offloaded and processed at the edge and remote cloud servers (CSs) with the more powerful computing capability. However, IIoT devices with emerging applications and services generate a very large amount of data at the network edge, which creates serious delay bottlenecks in sending data between users and edge/cloud servers. The limited radio spectrum may also create an unstable and intermittent network connectivity to offload data from a massive number of UEs to ESs, resulting high end-to-end (e2e) latency communication and potentially high energy consumption. To overcome these challenges, an intelligent joint design of task offloading and resource allocation decisions is required to reap full advantage of edge and cloud computing to ultimately attain the optimal e2e latency while meeting URLLC requirements and other system constraints such as low energy consumption at edge devices [4]–[6].

A. Review of Related Literature

Task offloading designs for MEC have been widely investigated in the literature (see [7] and the references therein). In most existing works, energy-efficiency and delay-efficiency are considered as major figure-of-merit in designing task offloading schemes for MEC systems [8]–[14]. In particular, a novel offload forwarding scheme was proposed in [8], where fog servers (FSs) cooperate with each other to tackle their heterogeneousness in terms of computation capacity and resources, improving the efficiency of power usage. In [9], a reformulation-linearization-technique-based Branch-and-Bound (BnB) method was developed to minimize the energy consumption of end devices by jointly optimizing the offloading selection, radio and computation resource allocation. The results in [10] showed that joint transmission energy allocation and task allocation design can significantly reduce the total energy consumption. The authors in [11] developed novel BnB and heuristic algorithms to solve the mixed-integer non-convex problems. Focusing on delay-efficiency, a distributed BnB approach to minimize the long-term average of the response time delay was proposed in [12]. From the economic point of view, Duong *et al.* [13] developed a new market-based framework to optimize heterogeneous network resources at the edge by dynamically pricing distributed MEC servers. In [14], the authors proposed a distributed task offloading scheme to maximize the expected offloading rates, where the impacts of queueing dynamics and wireless network interference are taken into account.

To guarantee low-latency wireless communication, short packets to convey a small amount of data must be used [1]. This however will pose several challenges to design and optimize the performance of short packet-enabled networks since it demands for more resources (e.g., parity, redundancy) and ultrahigh reliability. In addition, the performance analysis of throughput and decoding error probability under the short packet communication is more complex than the traditional Shannon capacity under the long block-length regime. Fortunately, the approximated achievable rate in the short block-length regime was derived in [15], which is a simple function of the traditional Shannon channel capacity, channel dispersion and complementary Gaussian cumulative distribution function for a given blocklength and error probability. Since then, resource allocation in the URLLC-based short block-length regime has recently studied to reduce the required bandwidth, the packet dropping [16] and maximize the energy efficiency (EE) [17]. Focusing on designing URLLC-aware optimization for task offloading, Zhou *et al.* [18] proposed the exponential-weight algorithm to balance URLLC constraints and energy consumption through online learning. The authors in [4] proposed a user-server association policy to reduce users' power consumption while trading off the resource allocations for local computation and task offloading.

The current literature on resource allocation in hybrid edge- and fog-cloud computing systems is still sparse and isolated. For example, the authors in [19] developed an efficient offloading scheme to minimize the average task duration. However, the radio resource allocation to support task offloading was not considered in this work. Wang *et al.* [20] proposed a modified BnB approach to solve the problem of power control and task allocation, aiming to minimize the total delay, where the energy consumption and delay requirements for each user are taken into consideration. The work in [21] jointly optimized the task assignment and throughput to minimize the computation latency for a single user. A non-orthogonal multiple-access (NOMA)-aided cooperative computing scheme was proposed in [22] that allows a single user can simultaneously offload computation tasks to a helper and a base station. The collaboration amongst fog/edge servers and cloud to achieve the energy and delay trade-off was studied in [23]. In these works, a network with small size (i.e., a single user or single MEC server) is considered or the impact of radio resource allocation and URLLC is not jointly analyzed and optimized.

B. Motivation and Main Contributions

To fully exploit the potential benefits offered by a hierarchical edge-cloud system, there are still several formidable challenges that need to be tackled, including communication costs, heterogeneous computational capabilities of ESs as well as limited radio resources. Although ESs and FSs are often equipped with more powerful computing capability than end users, they are still limited compared to large-scale cloud data centers at the cloud server. The straggler effect is a major bottleneck in implementing computation offload in hierarchical edge-cloud systems. For example, an ES with limited computation capability admitting tasks from many users with high arrival rate may increase processing latency, leading to higher e2e latency. All these challenges have not yet been fully addressed in the aforementioned works. In addition, a comprehensive analysis for the e2e latency model considering all factors of mission-critical communications (including URLLC) and queuing-aware computation is not presented in [2], [23].

Moving beyond the above background, this work proposes a novel joint communication and computation for URLLC-enabled hierarchical edge-cloud system, taking into account all the above issues. The main goal is to minimize the e2e latency of computational tasks among multiple UEs (IIoT devices). Our main contributions are summarized as follows:

- We first develop efficient offloading decisions amongst three layers of user, edge, and cloud by introducing new binary variables to establish UE-ES association policies. This design helps decide which ES is most suitable to handle computation tasks from UEs under available computation and communication resources. To mitigate the straggler effect caused by URLLC-aided uplink transmission, we adopt the matched filtering and successive interference cancellation (MF-SIC) receiver at ESs, and the rigorous closed-form expression of the total e2e latency is then provided. We formulate a generalized minimization problem for the e2e latency by incorporating various aspects of joint communication and computation offloading (called JCCO), such as offloading probabilities, processing rates, user association policies and power control subject to their service delay and energy consumption requirements, which is a mixed-integer non-convex optimization problem.

- We propose a simple yet efficient iterative algorithm by leveraging the alternating optimization (AO) approach and inner approximation (IA) framework [24], which solves the JCCO problem sub-optimally. To develop this algorithm, we first decompose the original problem into two subproblems to bypass the strong coupling among the optimization variables. For each subproblem, we provide newly approximate convex functions to convexify non-convex parts, and the AO-based iterative algorithm is then developed. To speed up the convergence of the proposed AO algorithm, we penalize relaxed binary variables by introducing a parameterized relaxed JCCO problem while still guaranteeing the satisfaction of binary nature.
- Towards appealing applications, two sub-optimal designs based on given user association policies are proposed, *namely* best channel selection (BCS) and random user association (RUA). The BCS scheme selects the strongest wireless link between a UE and ESs, while the RUA randomly assigns a UE to an ES. The corresponding problems are special cases of the JCCO problem that can be easily solved by the AO-based iterative algorithm after some slight modifications.
- Extensive numerical results are provided to evaluate the effectiveness of the proposed algorithms in terms of the convergence speed, e2e latency, and offloading portion, compared with existing benchmark schemes. They also reveal the excellent performance gain achieved by joint optimization of offloading probabilities, processing rates, user association policies and power control in a hierarchical edge-cloud system.

C. Paper Structure and Notations

The rest of this paper is organised as follows. Section II describes the system model and problem formulation. In Section III, we provide the AO-based iterative algorithms for solving the

JCCO and parameterized JCCO problems. Two sub-optimal designs are presented in Section IV. Numerical results are provided in Sections V, while Section VI concludes the paper.

Notation: Throughout the paper, numbers and vectors are denoted by lower-case and bold-face lower-case letters, respectively. $(\cdot)^T$ and $(\cdot)^H$ indicate the transpose and conjugate transpose of a matrix or vector, respectively. $|\cdot|$ and $\|\cdot\|_2$ denote the absolute value of a scalar and the $l2$-norm operator of a vector, respectively. $\mathbb{E}[\cdot]$ represents the expectation operation. $\mathcal{CN}(\mu,\sigma^2)$ is circularly symmetric complex Gaussian random variable with mean μ and variance σ^2.

II. System Model and Problem Formulation

A. System Model

In this paper, we consider a hierarchical edge-cloud system illustrated in Fig. 1. There are the set $\mathcal{M}=\{1,2,\cdots M\}$ of M UEs, i.e., IIoT devices including actuators, robots and sensors randomly distributed in a factory automation scenario, and the set $\mathcal{K} = \{1,2,\cdots,K\}$ of K ESs at the edge layer. Each ES is co-located with an access point (AP) to communicate with UEs over URLLC wireless links. The ESs connect with the cloud server via wired fronthaul links. The description of three-layer of the edge-cloud computing system is given as:

- ***User layer*** includes multiple UEs, which can be robots, actuators or sensors, etc. A task of UE $m\in\mathcal{M}$ can be executed locally with processing rate f_m^{lo} (cycles/s) or offloaded to a ES with probability of $\alpha_m\in[0,1]$. We use the indicator vector $\boldsymbol{\pi}\triangleq[\pi_{mk}]_{\forall m,k}$ to denote the association between UEs and ESs. In particular, $\pi_{mk}=1$ means ES $k\in\mathcal{K}$ admits tasks from UE $m\in\mathcal{M}$; otherwise, $\pi_{mk}=0$. We assume that the tasks of a UE is only offloaded a portion of α_m to one ES, i.e., $\sum_{k\in\mathcal{K}}\pi_{mk}=1,\forall m$.
- ***Edge layer*** consists of K ESs placed close to UEs, where the processing rate of ES k is denoted as f_k^{es} (cycles/s). To minimize the processing latency while admitting computation tasks from multiple UEs, ES k can offload a portion of $\beta_{mk}\in[0,1]$ of tasks of UE m to the cloud server through the fronthaul link.
- ***Cloud layer*** contains large-scale cloud data centers equipped with powerful processing units, which can process complex computational tasks with very high processing rate f^{cs} (cycles/s).

Suppose a task of UE m is characterised by a tuple $I_m\triangleq(D_m,C_m,T_m^{\max})$, in which D_m, C_m and $T_m^{\max}$ are the input task size, the required computation resource (number of CPU cycles) and the maximum delay requirement of this task, respectively. For transmission in URLLC-based

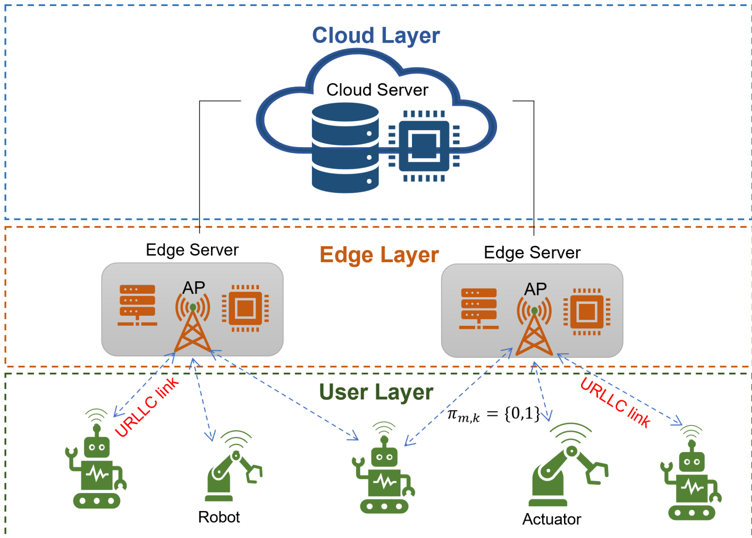


Fig. 1: An URLLC-enabled hierarchical edge-cloud system model.

links, the tasks can be split into multiple short packets to guarantee low-latency communications. The mean task arrival rate of UE m is denoted as λ_m^{lo} (tasks/s).

B. Communication and Computation Models

1) Communication Model: Each AP is equipped with $L > 1$ antennas while each UE has single antenna. The channel vector between UE m and AP k, denoted by $\mathbf{h}_{mk} \in \mathbb{C}^{L\times 1}$, can be modeled as $\mathbf{h}_{mk} = \sqrt{g_{mk}}\bar{\mathbf{h}}_{mk}$, where g_{mk} is the large-scale channel coefficient including the pathloss and shadowing which is normalized by the noise power, and $\bar{\mathbf{h}}_{mk}$ is the small-scale fading following the Rayleigh fading model as $\bar{\mathbf{h}}_{mk} \sim \mathcal{CN}(0, \mathbf{I}_L)$. Under a shared wireless medium, the $L\times 1$ received signal vector at AP k can be expressed as $\mathbf{y}_k = \sum_{m\in\mathcal{M}} \mathbf{h}_{mk}\sqrt{p_m}s_m + \mathbf{z}_k$, where p_m and s_m are the transmit power and unit-power data symbol of UE m, respectively; $\mathbf{z}_k \sim \mathcal{CN}(\mathbf{0}, \mathbf{I}_L)$ is the additive white Gaussian noise (AWGN) with zero mean and unit variance.

Uplink Channel Estimation: We adopt the time-division duplex (TDD) operation, where all the UEs send their pilot sequences to APs to perform channel estimation [25]. The number of pilot symbols for uplink channel estimation should be at least equal to the number of UEs. We consider that each coherence interval of all UEs is divided into two main phases, including uplink training with n_p symbols and n_d symbols for data transmission. The time duration for channel estimation and data transmission in one coherence interval can be expressed as $t_p = n_p/B$ and $t_d = n_d/B$, respectively, where B is the system bandwidth.

We assume that all the pilot sequences are mutually orthogonal. The MMSE channel estimate of $\mathbf{h}_{mk}$ is given by [25]:

$$\hat{\mathbf{h}}_{mk} = \frac{g_{mk} M p_m^p}{g_{mk} M p_m^p + 1} \mathbf{y}_{mk}^p \tag{1}$$

which follows the distribution of $\mathcal{CN}\left(\mathbf{0}, \sigma_{mk}^2 \mathbf{I}\right)$, where σ_{mk}^2 is given as $\sigma_{mk}^2 = g_{mk}^2 M p_m^p / (g_{mk} M p_m^p + 1)$ and p_m^p is the pilot transmit power of UE m. According to the minimum mean square error (MMSE) estimation property, the channel estimation error $\tilde{\mathbf{h}}_{mk} = \mathbf{h}_{mk} - \hat{\mathbf{h}}_{mk}$ is independent of $\hat{\mathbf{h}}_{mk}$ that follows the distribution of $\mathcal{CN}\left(\mathbf{0}, \delta_{mk}^2 \mathbf{I}_L\right)$, where δ_{mk}^2 is given by $\delta_{mk}^2 = g_{mk} / (g_{mk} M p_m^p + 1)$.

URLLC Uplink Transmission Rate: all M UEs simultaneously send their data to APs. To reduce information exchange between APs and the cloud server via fronthaul links, we adopt the matched filtering $(i.e., \hat{\mathbf{h}}_{mk}^H)$ and successive interference cancellation, called the MF-SIC receiver for signal detection in the uplink [26], which only requires the local CSI of all UEs at each AP. To guarantee fairness among all UEs, we assume that the decoding order follows UEs' index by arranging the channel vectors as $\|\hat{\mathbf{h}}_{1k}\|^2 \geq \|\hat{\mathbf{h}}_{2k}\|^2 \geq, \cdots, \geq \|\hat{\mathbf{h}}_{Mk}\|^2, \forall k$. In other words, AP k decodes the signals of UEs with better channel condition first and then remove them before decoding the signals of UEs with poorer channel conditions. Under MF-SIC receiver and definition of $\boldsymbol{\pi}$ in Section II-A, the instantaneous signal-to-interference-plus-noise (SINR) of UE m can be expressed as

$$\gamma_m(\mathbf{p}, \boldsymbol{\pi}_m) = \frac{\sum\limits_{k \in \mathcal{K}} \pi_{mk} p_m \|\hat{\mathbf{h}}_{mk}\|^4}{\Phi_m(\mathbf{p}, \boldsymbol{\pi}_m)} \tag{2}$$

where

$$\begin{aligned}\Phi_m(\mathbf{p}, \boldsymbol{\pi}_m) \triangleq & \sum_{i > m} \sum_{k \in \mathcal{K}} \pi_{mk} p_i |\hat{\mathbf{h}}_{mk}^H \hat{\mathbf{h}}_{ik}|^2 + \sum_{i \in \mathcal{M}} \sum_{k \in \mathcal{K}} (1 - \pi_{mk}) p_i |\hat{\mathbf{h}}_{mk}^H \hat{\mathbf{h}}_{ik}|^2 \\ & + \sum_{i \in \mathcal{M}} \sum_{k \in \mathcal{K}} \pi_{mk} p_i |\hat{\mathbf{h}}_{mk}^H \tilde{\mathbf{h}}_{ik}|^2 + \|\hat{\mathbf{h}}_{mk}\|^2 \end{aligned} \tag{3}$$

with $\mathbf{p} = \{p_m\}_{\forall m}$ and $\boldsymbol{\pi}_m = \{\pi_{mk}\}_{\forall k}$. In this paper, we focus the ergodic achievable rate of UE m, where $\bar{\gamma}_m(\mathbf{p}, \boldsymbol{\pi}_m) = \mathbb{E}\{\gamma_m(\mathbf{p}, \boldsymbol{\pi}_m)\}$ can be approximated by

$$\bar{\gamma}_m(\mathbf{p}, \boldsymbol{\pi}_m) = \frac{\sum\limits_{k \in \mathcal{K}} \pi_{mk} \left(L - 1\right) p_m \sigma_{mk}^2}{\bar{\Phi}_m(\mathbf{p}, \boldsymbol{\pi}_m)} \tag{4}$$

with $\bar{\Phi}_m(\mathbf{p}, \boldsymbol{\pi}_m) \triangleq \sum\limits_{i > m} \sum\limits_{k \in \mathcal{K}} \pi_{mk} p_i \sigma_{i,k}^2 + \sum\limits_{i \in \mathcal{M} \setminus m} \sum\limits_{k \in \mathcal{K}} \left(1 - \pi_{mk}\right) p_i \sigma_{i,k}^2 + \sum\limits_{i \in \mathcal{M}} \sum\limits_{k \in \mathcal{K}} \pi_{mk} p_i \delta_{i,k}^2 + 1$, whose derivation is given in Appendix A.

The uplink achievable data rate of UE m (in bits/s) under URLLC finite blocklength can be approximated as [15], [27]:

$$R_m(\mathbf{p},\boldsymbol{\pi}_m) = (1-\omega) B \log_2\left[1+\bar{\gamma}_m(\mathbf{p},\boldsymbol{\pi}_m)\right] - B\sqrt{\frac{(1-\omega) V_m(\mathbf{p},\boldsymbol{\pi}_m)}{N}}\frac{Q^{-1}(\epsilon_m)}{\ln 2} \quad (5)$$

where $N = \Delta_t B$ denotes the blocklength with Δ_t being the transmission time interval and $\omega \triangleq M/N$; ϵ_m is the decoding error probability, $Q^{-1}(\cdot)$ is the inverse function defined by $Q(x) = \frac{1}{\sqrt{2\pi}}\int_x^\infty \exp^{\left(\frac{-t^2}{2}\right)} dt$, and $V(\mathbf{p},\boldsymbol{\pi}_m)$ is the channel dispersion given by $V_m(\mathbf{p},\boldsymbol{\pi}_m) = 1-\left[1+\bar{\gamma}_m(\mathbf{p},\boldsymbol{\pi}_m)\right]^{-2}$. When the blocklength N goes to infinity, the data rate will approach to $R_m(\mathbf{p},\boldsymbol{\pi}_m) \rightarrow (1-\omega) B \log_2\left[1+\bar{\gamma}_m(\mathbf{p},\boldsymbol{\pi}_m)\right]$, which is the traditional Shannon rate function.

2) Computation Model: We now model the overall e2e latency of the considered hierarchical edge-cloud system, including the latency of the local processing, uplink wireless transmission through URLLC links, ESs' processing, fronthaul transmission and cloud processing. We note that the data size of the computation results is typically very small (e.g. control packets) while the APs can transmit with high power, and therefore, the downlink transmission latency can be ignored [28].

Local Processing Latency: UE m can partially offload with the portion α_m of its task to the ES. The latency to process the remaining task at UE m with the processing rate f_m^{lo} is given as

$$t_m^{\mathsf{lo}}(\alpha_m, f_m^{\mathsf{lo}}) = \frac{(1-\alpha_m) C_m}{f_m^{\mathsf{lo}}}. \quad (6)$$

Wireless transmission latency: Given the uplink data rate in (5), the latency to transmit the portion α_m of UE m's task is caculated as

$$t_m^{\mathsf{co}}(\alpha_m, \mathbf{p}, \boldsymbol{\pi}_m) = \frac{\alpha_m D_m}{R_m(\mathbf{p},\boldsymbol{\pi}_m)}. \quad (7)$$

ES Processing Latency: Let λ_k^{es} and λ_m^{lo} be the mean arrival rates of tasks at ES k and UE m, respectively. We have $\lambda_k^{\mathsf{es}} = \sum_{m\in\mathcal{M}} \pi_{mk}\alpha_m\lambda_m^{\mathsf{lo}}$ [29]. We denote by $\beta_{mk} \in [0,1]$ the offloading portion of task m from ES k to CS. As a result, the task processing rate to execute the remaining tasks offloaded from all UEs at ES k can be computed as $\mu_k = f_k^{\mathsf{es}} / \sum_{m\in\mathcal{M}} \pi_{mk}\alpha_m(1-\beta_{mk}) C_m$. By following the standard queuing model M/M/1 [29], we can compute the worst-case processing latency among ESs as

$$t^{\mathsf{es}}(\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\pi}) = \max_{\forall k\in\mathcal{K}}\left\{\frac{1}{\mu_k - \lambda_k^{\mathsf{es}}}\right\} = \max_{\forall k\in\mathcal{K}}\left\{\frac{1}{\frac{f_k^{\mathsf{es}}}{\sum_{m\in\mathcal{M}}\pi_{mk}\alpha_m(1-\beta_{mk})C_m} - \sum_{m\in\mathcal{M}}\pi_{mk}\alpha_m\lambda_m^{\mathsf{lo}}}\right\} \quad (8)$$

where $\boldsymbol{\alpha} \triangleq \{\alpha_m\}_{\forall m}$ and $\boldsymbol{\beta} \triangleq \{\beta_{mk}\}_{\forall m,k}$.

Fronthaul Transmission Latency: Each ES k transmits the portion β_{mk} of the offloaded task $\pi_{mk}\alpha_m$ of all UEs $m \in \mathcal{M}$ to CS for further processing. The worst-case transmission latency to offload tasks from ESs to CS via fronthaul links can be expressed as

$$t^{\mathsf{fh}}\left(\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\pi}\right)=\max_{\forall k\in\mathcal{K}}\left\{\sum_{m\in\mathcal{M}}\pi_{mk}\alpha_m\beta_{mk}\frac{D_m}{R_k^{\mathsf{fh}}}\right\} \tag{9}$$

where R_k^{fh} is the fronthaul capacity between ES k and CS.

Cloud Processing Latency: Given the processing rate f^{cs}, the latency for the CS to process offloaded tasks ESs can be expressed as

$$t^{\mathsf{cs}}\left(\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\pi}\right)=\frac{1}{\dfrac{f^{\mathsf{cs}}}{\sum\limits_{m\in\mathcal{M}}\sum\limits_{k\in\mathcal{K}}\pi_{mk}\alpha_m\beta_{mk}C_m}-\sum\limits_{m\in\mathcal{M}}\sum\limits_{k\in\mathcal{K}}\pi_{mk}\alpha_m\beta_{mk}\lambda_m^{\mathsf{lo}}} \tag{10}$$

where $f^{\mathsf{cs}}/\sum\limits_{m\in\mathcal{M}}\sum\limits_{k\in\mathcal{K}}\pi_{mk}\alpha_m\beta_{mk}C_m$ and $\sum\limits_{m\in\mathcal{M}}\sum\limits_{k\in\mathcal{K}}\pi_{mk}\alpha_m\beta_{mk}\lambda_m^{\mathsf{lo}}$ are considered as the task processing rate and the mean task arrival rate at CS, respectively.

C. Problem Formulation

From (6)–(10), the overall e2e latency of UE m including processing latency of UEs, ESs, CS, transmission latency from UEs to ESs and ESs to CS is given by

$$\begin{aligned} t_m(f_m^{\mathsf{lo}},\mathbf{p},\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\pi}) &= t_m^{\mathsf{lo}}(\alpha_m,f_m^{\mathsf{lo}})+t_m^{\mathsf{co}}\left(\alpha_m,\mathbf{p},\boldsymbol{\pi}_m\right)+t^{\mathsf{es}}\left(\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\pi}\right)\\ &\quad+t^{\mathsf{fh}}\left(\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\pi}\right)+t^{\mathsf{cs}}\left(\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\pi}\right). \end{aligned} \tag{11}$$

There are five parts of the overall e2e latency that can be classified into two categories, including communication latency and computation latency. Typically, the wireless transmission latency is the major source of the overall e2e latency. This reflects practical scenarios, where the wireless transmission is affected by many factors, e.g., channel conditions, transmit power and locations of devices, while the computation capacity of UEs, ESs and CS are large enough to execute tasks rapidly.

The total energy of UE m consumed for the local processing and uplink transmission can be computed as [3], [30]

$$E_m(\alpha_m,f_m^{\mathsf{lo}},\mathbf{p},\boldsymbol{\pi})=(1-\alpha_m)\frac{\theta_m}{2}C_m(f_m^{\mathsf{lo}})^2+p_m\frac{\alpha_mD_m}{R_m\left(\mathbf{p},\boldsymbol{\pi}_m\right)} \tag{12}$$

where the constant $\theta_m/2$ denotes the average switched capacitance and the average activity factor of UE m [4].

In this paper, we address a JCCO problem that aims to minimize the worst-case e2e latency among computational tasks under their service delay and energy consumption requirements. The

JCCO problem is mathematically formulated as follows

$$\mathbf{JCCO}: \underset{\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\pi},\mathbf{p},\mathbf{f}}{\text{minimize}} \ \max_{\forall m\in\mathcal{M}} \left\{ t_m\left(f_m^{\mathsf{lo}}, \mathbf{p}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\pi}\right)\right\} \tag{13a}$$

$$\text{s.t. } t_m\left(f_m^{\mathsf{lo}}, \mathbf{p}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\pi}\right) \leq T_m^{\max}, \forall m \tag{13b}$$

$$E_m(\alpha_m, f_m^{\mathsf{lo}}, \mathbf{p}, \boldsymbol{\pi}) \leq E_m^{\max}, \forall m \tag{13c}$$

$$R_m\left(\mathbf{p}, \boldsymbol{\pi}_m\right) \geq \sum_{k\in\mathcal{K}} \pi_{mk} R_m^{\min}, \forall m \tag{13d}$$

$$\sum_{m\in\mathcal{M}} \pi_{mk}\alpha_m\lambda_m^{\mathsf{lo}} \leq \frac{f_k^{\mathsf{es}}}{\sum_{m\in\mathcal{M}} \pi_{mk}\alpha_m\left(1-\beta_{mk}\right)C_m}, \forall k \tag{13e}$$

$$\sum_{m\in\mathcal{M}}\sum_{k\in\mathcal{K}} \pi_{mk}\alpha_m\beta_{mk}\lambda_m^{\mathsf{lo}} \leq \frac{f^{\mathsf{cs}}}{\sum_{m\in\mathcal{M}}\sum_{k\in\mathcal{K}} \pi_{mk}\alpha_m\beta_{mk}C_m} \tag{13f}$$

$$\boldsymbol{\alpha}, \boldsymbol{\beta} \in \mathscr{D}, \boldsymbol{\pi} \in \Pi, \mathbf{p} \in \mathscr{P}, \mathbf{f} \in \mathscr{F} \tag{13g}$$

where $\mathscr{D} \triangleq \{\alpha_m, \beta_{mk}, \forall m,k | 0 \leq \alpha_m \leq 1, 0 \leq \beta_{mk} \leq 1, \forall m,k\}$, $\mathscr{P} \triangleq \{p_m, \forall m | 0 \leq p_m \leq P_m^{\max}, \forall m\}$, $\mathscr{F} \triangleq \{f_m^{\mathsf{lo}}, \forall m | 0 \leq f_m^{\mathsf{lo}} \leq F_m^{\max}, \forall m\}$, and $\Pi \triangleq \{\pi_{mk}, \forall m,k | \pi_{mk} \in \{0,1\} \,\&\, \sum_{k\in\mathcal{K}} \pi_{mk} = 1, \forall m,k\}$ are the set constraints of offloading decisions, uplink transmission power, processing rates and association policies, respectively; Herein, $P_m^{\max}$ and $F_m^{\max}$ are the maximum power budget and processing rate of UE $m \in \mathcal{M}$, respectively. Constraints (13b) and (13c) are imposed to ensure that the overall e2e latency and energy consumption of UE m are limited by the predetermined thresholds $T_m^{\max}$ and $E_m^{\max}$, respectively. Constraint (13d) guarantees the minimum rate requirement $R_m^{\min}$ for all UEs. Finally, constraints (13e) and (13f) are added to ensure the queue stability at ESs and CS, respectively.

III. Proposed AO-based Algorithms for Solving Problem **JCCO**

A. *Challenges of Solving Problem* ***JCCO*** (13)

We can see that the objective (13a) is nonconcave and nonsmooth, and the feasible set is also non-convex due to the non-convexity of constraints (13b)–(13f). Due to binary variables $\boldsymbol{\pi}$, the JCCO problem (13) is a mixed-integer non-convex optimization program, which is generally NP-hard. The main barrier in solving problem (13) is due to the binary constraint $\boldsymbol{\pi} \in \Pi$ in (13g). We note that it is not practical to apply a direct application of the well-known brute-force search (BFS) method by searching over all possible association policies, even for networks of small-to-medium size. In addition, the improved branch-and-bound algorithm (IBBA) method presented in [2] is also inapplicable as the relaxed problem of JCCO is still non-convex due the strong coupling between continuous variables and binary variables.

In what follows, we first transform the original problem (13) into a more computational tractable form by bypassing the nonsmooth of the objective function. By exploiting the unique structure of the underlying problem, we decompose it into two subproblems, which are optimized in an iterative fashion over $(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\pi})$ and $(\mathbf{p}, \mathbf{f})$. We then leverage the combination of AO method and IA framework to solve subproblems efficiently, which converge to at least a local optimal solution.

B. Approximate Convex Problems

Let us start by rewriting the JCCO problem (13) equivalently as

$$\underset{\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\pi},\mathbf{p},\mathbf{f},\boldsymbol{\tau}}{\text{minimize}} \ \max_{\forall m\in\mathcal{M}} \left\{ t_m(f_m^{\mathsf{lo}}, \boldsymbol{\tau}) \right\} \tag{14a}$$

$$\text{s.t.} \quad (13\text{c}), (13\text{d}), (13\text{e}), (13\text{f}), (13\text{g}) \tag{14b}$$

$$t_m\big(f_m^{\mathsf{lo}}, \boldsymbol{\tau}\big) \leq T_m^{\max}, \ \forall m \tag{14c}$$

$$\tau^{\mathsf{co}} \geq t_m^{\mathsf{co}}\left(\alpha_m, \mathbf{p}, \boldsymbol{\pi}_m\right), \ \forall m \tag{14d}$$

$$\tau^{\mathsf{es}} \geq t^{\mathsf{es}}\left(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\pi}\right) \tag{14e}$$

$$\tau^{\mathsf{fh}} \geq t^{\mathsf{fh}}\left(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\pi}\right) \tag{14f}$$

$$\tau^{\mathsf{cs}} \geq t^{\mathsf{cs}}\left(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\pi}\right) \tag{14g}$$

where $t_m(f_m^{\mathsf{lo}}, \boldsymbol{\tau}) \triangleq \dfrac{(1-\alpha_m)C_m}{f_m^{\mathsf{lo}}} + \tau^{\mathsf{co}} + \tau^{\mathsf{es}} + \tau^{\mathsf{fh}} + \tau^{\mathsf{cs}}$, and $\boldsymbol{\tau} \triangleq \{\tau^{\mathsf{co}}, \tau^{\mathsf{es}}, \tau^{\mathsf{fh}}, \tau^{\mathsf{cs}}\}$ are newly introduced variables to simplify the objective function. Constraint (14c) is derived from (13b). We introduce the following lemma to verify the equivalence between problems (13) and (14).

Lemma 1. *There exists a set* $(\boldsymbol{\alpha}^\star, \boldsymbol{\beta}^\star, \boldsymbol{\pi}^\star, \mathbf{p}^\star, \mathbf{f}^\star)$ *which is the optimal solution to both problems* (13) *and* (14)*, resulting in the same objective value. In other words, if* $(\boldsymbol{\alpha}^\star, \boldsymbol{\beta}^\star, \boldsymbol{\pi}^\star, \mathbf{p}^\star, \mathbf{f}^\star, \boldsymbol{\tau}^\star)$ *is the optimal solution to problem* (13)*, then* $(\boldsymbol{\alpha}^\star, \boldsymbol{\beta}^\star, \boldsymbol{\pi}^\star, \mathbf{p}^\star, \mathbf{f}^\star)$ *is also the optimal solution to problem* (14) *and vice versa.*

Proof. The proof is straightforward by showing that constraints (14d)-(14g) must hold with equality at optimum. We now prove for constraint (14d) and other ones follow immediately. Assume that the equality of (14d) does not hold at the optimum for some m, i.e., $\tau^{\mathsf{co},\star} > t_m^{\mathsf{co}}\left(\alpha_m^\star, \mathbf{p}^\star, \boldsymbol{\pi}_m^\star\right)$. There exists a positive constant $\Delta\tau^{\mathsf{co}} > 0$ which guarantees $\tau^{\mathsf{co},\star} - \Delta\tau^{\mathsf{co}} =$

$t_m^{\mathsf{co}}\left(\alpha_m^\star, \mathbf{p}^\star, \boldsymbol{\pi}_m^\star\right)$. As a result, $\tau^{\mathsf{co},\star} - \Delta\tau^{\mathsf{co}}$ is also feasible to problem (14), but leading to a strictly lower e2e latency. This contracdicts with the original assumption that the set $(\boldsymbol{\alpha}^\star, \boldsymbol{\beta}^\star, \boldsymbol{\pi}^\star, \mathbf{p}^\star, \mathbf{f}^\star)$ is the optimal solution to problem (14). $\square$

It is clear that the objective (14a) is a convex function in $(f_m^{\mathsf{lo}}, \boldsymbol{\tau})$. We also note that a direct application of IA method is still inapplicable due to strong coupling between variables. Considering the fact that the decision variables $(\mathbf{p}, \mathbf{f})$ and $(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\pi})$ can be executed from UEs' and ESs' sides, respectively. Let us denote by $x^{(i)}$ the feasible point of x at the i-th iteration of the proposed iterative algorithm, which is a constant. By leveraging AO method, at iteration i, we decompose problem (14) into two subproblems (SPs) as follows:

$$\mathsf{SP\text{-}1:} \underset{\mathbf{p},\mathbf{f},\boldsymbol{\tau}|\boldsymbol{\alpha}^{(i)},\boldsymbol{\beta}^{(i)},\boldsymbol{\pi}^{(i)}}{\text{minimize}} \quad \max_{\forall m\in\mathcal{M}} \left\{ t_m(f_m^{\mathsf{lo}}, \boldsymbol{\tau}) \right\} \tag{15a}$$

$$\text{s.t. } (13\text{c}), (13\text{d}), (14\text{c}), (14\text{d}) \tag{15b}$$

$$\mathbf{p} \in \mathscr{P}, \mathbf{f} \in \mathscr{F} \tag{15c}$$

and

$$\mathsf{SP\text{-}2:} \underset{\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\pi},\boldsymbol{\tau}|\mathbf{p}^{(i+1)},\mathbf{f}^{(i+1)}}{\text{minimize}} \quad \max_{\forall m\in\mathcal{M}} \left\{ t_m(f_m^{\mathsf{lo},(i+1)}, \boldsymbol{\tau}) \right\} \tag{16a}$$

$$\text{s.t. } (13\text{c}), (13\text{d}), (13\text{e}), (13\text{f}), (14\text{c}), (14\text{d}), (14\text{e}), (14\text{f}), (14\text{g}) \tag{16b}$$

$$\boldsymbol{\alpha}, \boldsymbol{\beta} \in \mathscr{D}, \boldsymbol{\pi} \in \Pi. \tag{16c}$$

In an AO-based iterative algorithm, we first solve SP-1 for given $(\boldsymbol{\alpha}^{(i)}, \boldsymbol{\beta}^{(i)}, \boldsymbol{\pi}^{(i)})$ to generate the next optimal point of $(\mathbf{p}^{(i+1)}, \mathbf{f}^{(i+1)})$ and then solve SP-2 for updated value of $(\mathbf{p}^{(i+1)}, \mathbf{f}^{(i+1)})$ to generate the next feasible point $(\boldsymbol{\alpha}^{(i+1)}, \boldsymbol{\beta}^{(i+1)}, \boldsymbol{\pi}^{(i+1)})$. This procedure is repeated until convergence. In what follows, we apply IA framework to convexify non-convex parts of two subproblems. To facilitate the approximation of non-convex parts, we provide some fundamental inequalities in Appendix B which satisfy the IA properties [31], [32].

1) Approximate Convex Program for SP-1: In problem (15), non-convex parts include (13c), (13d) and (14d). Let us handle constraint (13d) first by rewriting $R_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)})$ as

$$R_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}) = \frac{(1-\omega)\,B}{\ln 2}\left[G_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}) - \kappa_m V_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)})\right] \tag{17}$$

where $\kappa_m = Q^{-1}(\epsilon_m)/\sqrt{(1-\omega)\,N}$ and $G_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}) = \ln(1 + \bar{\gamma}_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}))$. To convexify constraint (13d), we need to devise a lower bounding concave function of $R_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)})$, which is provided in Lemma 2 of which the derivation is given Appendix C.

Lemma 2. *For $q_m(\mathbf{p},\boldsymbol{\pi}_m^{(i)}) \triangleq \bar{\Phi}_m(\mathbf{p},\boldsymbol{\pi}_m^{(i)}) / \sum_{k\in\mathcal{K}} \pi_{mk}^{(i)}\,(L-1)\,\sigma_{mk}^2$, the lower bounding concave function of $R_m(\mathbf{p},\boldsymbol{\pi}_m^{(i)})$ at the feasible point $\mathbf{p}^{(i)}$ is given as*

$$R_m(\mathbf{p},\boldsymbol{\pi}_m^{(i)}) \geq \mathcal{R}_m^{(i)}(\mathbf{p},\boldsymbol{\pi}_m^{(i)}) = \frac{(1-\omega)\,B}{\ln 2}\left[\mathcal{G}_m^{(i)}(\mathbf{p},\boldsymbol{\pi}_m^{(i)}) - \kappa_m \mathcal{V}_m^{(i)}(\mathbf{p},\boldsymbol{\pi}_m^{(i)})\right] \tag{18}$$

under the trusted regions

$$q_m(\mathbf{p},\boldsymbol{\pi}_m^{(i)}) + p_m \leq 2(q_m(\mathbf{p}^{(i)},\boldsymbol{\pi}_m^{(i)}) + p_m^{(i)}),\ \forall m \tag{19}$$

$$\frac{q_m(\mathbf{p},\boldsymbol{\pi}_m^{(i)}) + p_m}{q_m(\mathbf{p}^{(i)},\boldsymbol{\pi}_m^{(i)}) + p_m^{(i)}} \leq 2\frac{q_m(\mathbf{p},\boldsymbol{\pi}_m^{(i)})}{q_m(\mathbf{p}^{(i)},\boldsymbol{\pi}_m^{(i)})},\ \forall m \tag{20}$$

where $\mathcal{G}_m^{(i)}(\mathbf{p},\boldsymbol{\pi}_m^{(i)}) \triangleq a_m^{(i)} - \frac{b_m^{(i)}}{p_m} - c_m^{(i)} q_m(\mathbf{p},\boldsymbol{\pi}_m^{(i)})$ and $\mathcal{V}_m^{(i)}(\mathbf{p},\boldsymbol{\pi}_m^{(i)}) \triangleq d_m^{(i)} - \frac{2e_m^{(i)}}{q_m(\mathbf{p}^{(i)},\boldsymbol{\pi}_m^{(i)}) + p_m^{(i)}}$ $\big(2f_m^{(i)} q_m(\mathbf{p},\boldsymbol{\pi}_m^{(i)}) - (f_m^{(i)})^2\big(q_m(\mathbf{p},\boldsymbol{\pi}_m^{(i)}) + p_m\big)\big) + \frac{(f_m^{(i)})^2}{q_m^2(\mathbf{p}^{(i)},\boldsymbol{\pi}_m^{(i)})} q_m^2(\mathbf{p},\boldsymbol{\pi}_m^{(i)})$, and the constants $a_m^{(i)}, b_m^{(i)}, c_m^{(i)}$, $d_m^{(i)}, e_m^{(i)}, f_m^{(i)}$ are defined in Appendix C. Herein, $\mathcal{G}_m^{(i)}(\mathbf{p},\boldsymbol{\pi}_m^{(i)})$ and $\mathcal{V}_m^{(i)}(\mathbf{p},\boldsymbol{\pi}_m^{(i)})$ are the lower bounding concave function of $G_m(\mathbf{p},\boldsymbol{\pi}_m^{(i)})$ and the upper bounding convex function of $V_m(\mathbf{p},\boldsymbol{\pi}_m^{(i)})$, respectively, which satisfy $\mathcal{G}_m^{(i)}(\mathbf{p}^{(i)},\boldsymbol{\pi}_m^{(i)}) = G_m(\mathbf{p}^{(i)},\boldsymbol{\pi}_m^{(i)})$ and $\mathcal{V}_m^{(i)}(\mathbf{p}^{(i)},\boldsymbol{\pi}_m^{(i)}) = V_m(\mathbf{p}^{(i)},\boldsymbol{\pi}_m^{(i)})$.

As a result, constraint (13d) is iteratively replaced by the following convex constraint

$$\mathcal{R}_m^{(i)}(\mathbf{p},\boldsymbol{\pi}_m^{(i)}) \geq \sum_{k\in\mathcal{K}} \pi_{mk}^{(i)} R_m^{\min},\ \forall m \tag{21}$$

under the regions in (19) and (20).

Next, we introduce new variables $\mathbf{r} \triangleq \{r_m\}_{\forall m}$ to express constraint (13c) equivalently as

$$\begin{cases} (1-\alpha_m^{(i)})\dfrac{\theta_m}{2} C_m (f_m^{\mathsf{lo}})^2 + \alpha_m^{(i)} D_m p_m r_m \leq E_m^{\max}, \forall m & \text{(22a)} \\ \dfrac{1}{r_m} \leq \mathcal{R}_m^{(i)}(\mathbf{p},\boldsymbol{\pi}_m^{(i)}), \forall m & \text{(22b)} \end{cases}$$

where constraint (22a) is non-convex due to the product of $p_m r_m$. We note that $p_m r_m$ is a concave function which can be innerly approximated by (B.5) for $x = p_m,\ y = r_m,\ \bar{x} = p_m^{(i)},\ \bar{y} = r_m^{(i)}$, yielding

$$(1-\alpha_m^{(i)})\frac{\theta_m}{2} C_m (f_m^{\mathsf{lo}})^2 + \frac{1}{2}\alpha_m^{(i)} D_m \Big(\frac{r_m^{(i)}}{p_m^{(i)}} p_m^2 + \frac{p_m^{(i)}}{r_m^{(i)}} r_m^2\Big) \leq E_m^{\max}, \forall m. \tag{23}$$

Lastly, by (22b), constraint (14d) is iteratively replaced by the following linear constraint

$$\tau^{\mathsf{co}} \geq \alpha_m^{(i)} D_m r_m,\ \forall m. \tag{24}$$

As a result, we obtain the following approximate convex program of SP-1 (15) solved at iteration i:

$$\text{SP-1 Convex: } \underset{\mathbf{p},\mathbf{f},\boldsymbol{\tau},\mathbf{r}|\boldsymbol{\alpha}^{(i)},\boldsymbol{\beta}^{(i)},\boldsymbol{\pi}^{(i)}}{\text{minimize}} \quad \max_{\forall m\in\mathcal{M}} \big\{t_m(f_m^{\mathsf{lo}},\boldsymbol{\tau})\big\} \tag{25a}$$

$$\text{s.t. } (14c),(15c),(19),(20),(21),(22b),(23),(24). \tag{25b}$$

The complexity of solving the convex program (25) is only polynomial in the numbers of optimization variables and constraints. In particular, problem (25) involves $3M+4$ scalar decision variables and $9M$ linear and quadratic constraints, resulting in the per-iteration computational complexity of $\mathcal{O}\big(3\sqrt{M}(3M+4)^2\big)$ [33, Chapter 6].

2) Approximate Convex Program for SP-2: For given $(\mathbf{p}^{(i+1)},\mathbf{f}^{(i+1)})$ obtained by solving (25), we are now in position to convexify (16). To bypass the binary nature of (16), we first relax $\boldsymbol{\pi}$ to be continuous, i.e., $\boldsymbol{\pi}\in\tilde{\Pi}\triangleq\{\pi_{mk},\forall m,k|0\leq\pi_{mk}\leq 1 \ \& \ \sum_{k\in\mathcal{K}}\pi_{mk}=1,\forall m,k\}$ and rewrite it as

$$\text{SP-2: } \underset{\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\pi},\boldsymbol{\tau}|\mathbf{p}^{(i+1)},\mathbf{f}^{(i+1)}}{\text{minimize}} \quad \max_{\forall m\in\mathcal{M}} \big\{t_m(f_m^{\mathsf{lo},(i+1)},\boldsymbol{\tau})\big\} \tag{26a}$$

$$\text{s.t. } (13c),(13d),(13e),(13f),(14c),(14d),(14e),(14f),(14g) \tag{26b}$$

$$\boldsymbol{\alpha},\boldsymbol{\beta}\in\mathscr{D},\boldsymbol{\pi}\in\tilde{\Pi}. \tag{26c}$$

Constraints (14c) and (26c) are linear while others are non-convex.

Convexity of (13d) *and* (13c)*:* We rewrite $\bar{\gamma}_m(\mathbf{p}^{(i+1)},\boldsymbol{\pi}_m)=\frac{\sum_{k\in\mathcal{K}}\pi_{mk}\sigma_{mk}^2}{\tilde{q}_m\left(\mathbf{p}^{(i+1)},\boldsymbol{\pi}_m\right)}$ where $\tilde{q}_m(\mathbf{p}^{(i+1)},\boldsymbol{\pi}_m)$ is defined as

$$\tilde{q}_m(\mathbf{p}^{(i+1)},\boldsymbol{\pi}_m)\triangleq\frac{\sum\limits_{i>m}\sum\limits_{k\in\mathcal{K}}\pi_{mk}p_i^{(i+1)}\sigma_{i,k}^2+\sum\limits_{i\in\mathcal{M}\backslash m}\sum\limits_{k\in\mathcal{K}}(1-\pi_{mk})\,p_i^{(i+1)}\sigma_{i,k}^2+\sum\limits_{i\in\mathcal{M}}\sum\limits_{k\in\mathcal{K}}\pi_{ik}p_i^{(i+1)}\delta_{i,k}^2+1}{p_m^{(i+1)}\,(L-1)}.$$

It follows from Lemma 2 that

$$R_m(\mathbf{p}^{(i+1)},\boldsymbol{\pi}_m)\geq\tilde{\mathcal{R}}_m^{(i)}(\mathbf{p}^{(i+1)},\boldsymbol{\pi}_m)=\frac{(1-\omega)\,B}{\ln 2}\left[\tilde{\mathcal{G}}_m^{(i)}(\mathbf{p}^{(i+1)},\boldsymbol{\pi}_m)-\kappa_m\tilde{\mathcal{V}}_m^{(i)}(\mathbf{p}^{(i+1)},\boldsymbol{\pi}_m)\right] \tag{27}$$

under the trusted regions

$$\tilde{q}_m(\mathbf{p}^{(i+1)},\boldsymbol{\pi}_m)+\sum_{k\in\mathcal{K}}\pi_{mk}\sigma_{mk}^2\leq 2\Big(\tilde{q}_m(\mathbf{p}^{(i+1)},\boldsymbol{\pi}_m^{(i)})+\sum_{k\in\mathcal{K}}\pi_{mk}^{(i)}\sigma_{mk}^2\Big),\ \forall m \tag{28}$$

$$\frac{\tilde{q}_m(\mathbf{p}^{(i+1)},\boldsymbol{\pi}_m)+\sum\limits_{k\in\mathcal{K}}\pi_{mk}\sigma_{mk}^2}{\tilde{q}_m(\mathbf{p}^{(i+1)},\boldsymbol{\pi}_m^{(i)})+\sum\limits_{k\in\mathcal{K}}\pi_{mk}^{(i)}\sigma_{mk}^2}\leq 2\frac{\tilde{q}_m(\mathbf{p}^{(i+1)},\boldsymbol{\pi}_m)}{\tilde{q}_m(\mathbf{p}^{(i+1)},\boldsymbol{\pi}_m^{(i)})},\ \forall m \tag{29}$$

where

$$\tilde{\mathcal{G}}_m^{(i)}(\mathbf{p}^{(i+1)}, \boldsymbol{\pi}_m) \triangleq \tilde{a}_m^{(i)} - \frac{\tilde{b}_m^{(i)}}{\sum_{k\in\mathcal{K}} \pi_{mk}\sigma_{mk}^2} - \tilde{c}_m^{(i)}\tilde{q}_m(\mathbf{p}^{(i+1)}, \boldsymbol{\pi}_m)$$

$$\begin{aligned}\tilde{\mathcal{V}}_m^{(i)}(\mathbf{p}^{(i+1)}, \boldsymbol{\pi}_m) \triangleq \tilde{d}_m^{(i)} - &\frac{2\tilde{e}_m^{(i)}}{\tilde{q}_m(\mathbf{p}^{(i+1)}, \boldsymbol{\pi}_m^{(i)}) + \sum_{k\in\mathcal{K}} \pi_{mk}^{(i)}\sigma_{mk}^2}\big(2\tilde{f}_m^{(i)}\tilde{q}_m(\mathbf{p}^{(i+1)}, \boldsymbol{\pi}_m) \\ &- (\tilde{f}_m^{(i)})^2\big(\tilde{q}_m(\mathbf{p}^{(i+1)}, \boldsymbol{\pi}_m) + \sum_{k\in\mathcal{K}} \pi_{mk}\sigma_{mk}^2\big)\big) + \frac{(\tilde{f}_m^{(i)})^2}{\tilde{q}_m^2(\mathbf{p}^{(i+1)}, \boldsymbol{\pi}_m^{(i)})}\tilde{q}_m^2(\mathbf{p}^{(i+1)}, \boldsymbol{\pi}_m)\end{aligned}$$

and constants $\tilde{a}_m^{(i)}, \tilde{b}_m^{(i)}, \tilde{c}_m^{(i)}, \tilde{d}_m^{(i)}, \tilde{e}_m^{(i)}$ and $\tilde{f}_m^{(i)}$ are defined similarly as in Appendix C. As a result, we innerly approximate constraint (13d) as

$$\tilde{\mathcal{R}}_m^{(i)}(\mathbf{p}^{(i+1)}, \boldsymbol{\pi}_m) \geq \sum_{k\in\mathcal{K}} \pi_{mk}R_m^{\min},\ \forall m. \tag{30}$$

The constraint (13c) is equivalent to

$$\begin{cases} \dfrac{1}{\tilde{\mathcal{R}}_m^{(i)}\left(\mathbf{p}^{(i+1)}, \boldsymbol{\pi}_m\right)} \leq r_m,\ \forall m & \text{(31a)} \\ (1-\alpha_m)\,\dfrac{\theta_m}{2}C_m(f_m^{\mathsf{lo},(i+1)})^2 + p_m^{(i+1)}D_m\alpha_m r_m \leq E_m^{\max},\ \forall m & \text{(31b)} \end{cases}$$

where $\mathbf{r} \triangleq \{r_m\}_{\forall m}$ were defined (22). We use (B.5) to approximate $\alpha_m r_m$ in (31b) as

$$(1-\alpha_m)\,\frac{\theta_m}{2}C_m(f_m^{(i+1)})^2 + \frac{1}{2}p_m^{(i+1)}D_m\Big(\frac{r_m^{(i+1)}}{\alpha_m^{(i)}}\alpha_m^2 + \frac{\alpha_m^{(i)}}{r_m^{(i+1)}}r_m^2\Big) \leq E_m^{\max},\ \forall m. \tag{32}$$

Convexity of (13e) *and* (13f)*:* By introducing new variables $\check{\boldsymbol{\phi}} \triangleq \{\check{\phi}_{mk}\}_{\forall m,k}$, constraint (13e) is equivalently expressed as

$$\begin{cases} \sum_{m\in\mathcal{M}} \lambda_m^{\mathsf{lo}}\pi_{mk}\alpha_m \leq \dfrac{f_k^{\mathsf{es}}}{\sum_{m\in\mathcal{M}} C_m\check{\phi}_{mk}^2},\ \forall k & \text{(33a)} \\ \check{\phi}_{mk}^2 \geq \pi_{mk}\alpha_m(1-\beta_{mk}),\ \forall m,k. & \text{(33b)} \end{cases}$$

In (33a), the right-hand side (RHS) is a convex function which can be addressed by (B.2) while the product $\pi_{mk}\alpha_m$ in the left-hand side (LHS) can be approximated by (B.5). We iteratively replace (33a) by

$$\sum_{m\in\mathcal{M}} \frac{1}{2}\lambda_m^{\mathsf{lo}}\left(\frac{\alpha_m^{(i)}}{\pi_{mk}^{(i)}}\pi_{mk}^2 + \frac{\pi_{mk}^{(i)}}{\alpha_m^{(i)}}\alpha_m^2\right) \leq f_k^{\mathsf{es}}\left(\frac{2}{\sum_{m\in\mathcal{M}} C_m(\check{\phi}_{mk}^{(i)})^2} - \frac{\sum_{m\in\mathcal{M}} C_m\check{\phi}_{mk}^2}{\big(\sum_{m\in\mathcal{M}} C_m(\check{\phi}_{mk}^{(i)})^2\big)^2}\right),\ \forall k \tag{34}$$

which is a convex constraint. To handle constraint (33b), we first rewrite as $\dfrac{\check{\phi}_{mk}^2}{1-\beta_{mk}} \geq \pi_{mk}\alpha_m$, and apply inequalities (B.3) and (B.5) to approximate both sides as

$$\frac{2\check{\phi}_{mk}^{(i)}\check{\phi}_{mk}}{1-\beta_{mk}^{(i)}} - \frac{(\check{\phi}_{mk}^{(i)})^2(1-\beta_{mk})}{(1-\beta_{mk}^{(i)})^2} \geq \frac{1}{2}\left(\frac{\alpha_m^{(i)}}{\pi_{mk}^{(i)}}\pi_{mk}^2 + \frac{\pi_{mk}^{(i)}}{\alpha_m^{(i)}}\alpha_m^2\right),\ \forall m,k. \tag{35}$$

Similarly, by introducing new variables $\hat{\boldsymbol{\phi}} \triangleq \{\hat{\phi}_m\}_{\forall m}$, (13f) is equivalent to

$$\begin{cases} \sum_{m\in\mathcal{M}} \lambda_m^{\mathsf{lo}} \hat{\phi}_m^2 \le \frac{f^{\mathsf{cs}}}{\sum_{m\in\mathcal{M}} C_m \hat{\phi}_m^2} & \text{(36a)} \\ \frac{\hat{\phi}_m^2}{\alpha_m} \ge \sum_{k\in\mathcal{K}} \pi_{mk}\beta_{mk},\ \forall m & \text{(36b)} \end{cases}$$

which are approximated using (B.2), (B.3) and (B.5) as

$$\begin{cases} \sum_{m\in\mathcal{M}} \lambda_m^{\mathsf{lo}} \hat{\phi}_m^2 \le f^{\mathsf{cs}} \Bigg(\frac{2}{\sum_{m\in\mathcal{M}} C_m (\hat{\phi}_m^{(i)})^2} - \frac{\sum_{m\in\mathcal{M}} C_m \hat{\phi}_m^2}{\big(\sum_{m\in\mathcal{M}} C_m (\hat{\phi}_m^{(i)})^2\big)^2} \Bigg) & \text{(37a)} \\ \frac{2\hat{\phi}_m^{(i)}\hat{\phi}_m}{\alpha_m^{(i)}} - \frac{(\hat{\phi}_m^{(i)})^2\alpha_m}{(\alpha_m^{(i)})^2} \ge \sum_{k\in\mathcal{K}} \frac{1}{2}\Big(\frac{\beta_{mk}^{(i)}}{\pi_{mk}^{(i)}}\pi_{mk}^2 + \frac{\pi_{mk}^{(i)}}{\beta_{mk}^{(i)}}\beta_{mk}^2 \Big), \forall m. & \text{(37b)} \end{cases}$$

Convexity of (14d)*:* From (31a), we rewrite (14d) as $\tau^{\mathsf{co}} \ge D_m \alpha_m r_m$ and apply (B.5) to covexify $\alpha_m r_m$ as

$$\tau^{\mathsf{co}} \ge \frac{1}{2} D_m \Big(\frac{r_m^{(i)}}{\alpha_m^{(i)}} \alpha_m^2 + \frac{\alpha_m^{(i)}}{r_m^{(i)}} r_m^2 \Big),\ \forall m. \tag{38}$$

Convexity of (14e)*:* It follows from constraint (14e) that

$$\frac{f_k^{\mathsf{es}}}{\sum_{m\in\mathcal{M}} \pi_{mk}\alpha_m (1-\beta_{mk}) C_m} \ge \sum_{m\in\mathcal{M}} \lambda_m^{\mathsf{lo}} \pi_{mk}\alpha_m + \frac{1}{\tau^{\mathsf{es}}},\ \forall k \tag{39}$$

which can be transformed equivalently as

$$\frac{f_k^{\mathsf{es}}}{\sum_{m\in\mathcal{M}} C_m \check{\phi}_{mk}^2} \ge \sum_{m\in\mathcal{M}} \lambda_m^{\mathsf{lo}} \pi_{mk}\alpha_m + \frac{1}{\tau^{\mathsf{es}}} \tag{40}$$

by (33b). We apply inequalities (B.3) and (B.5) to lower bound the LHS and upper bound the function $\pi_{mk}\alpha_m$, respectively, i.e.,

$$f_k^{\mathsf{es}} \Bigg(\frac{2}{\sum_{m\in\mathcal{M}} C_m (\check{\phi}_{mk}^{(i)})^2} - \frac{\sum_{m\in\mathcal{M}} C_m \check{\phi}_{mk}^2}{\big(\sum_{m\in\mathcal{M}} C_m (\check{\phi}_{mk}^{(i)})^2\big)^2} \Bigg) \ge \sum_{m\in\mathcal{M}} \lambda_m^{\mathsf{lo}} \frac{1}{2} \Big(\frac{\alpha_m^{(i)}}{\pi_{mk}^{(i)}} \pi_{mk}^2 + \frac{\pi_{mk}^{(i)}}{\alpha_m^{(i)}} \alpha_m^2 \Big) + \frac{1}{\tau^{\mathsf{es}}} \tag{41}$$

Convexify of (14f)*:* We can express constraint (14f) as

$$\begin{cases} \tau^{\mathsf{fh}} \ge \sum_{m\in\mathcal{M}} \varphi_{mk}^2 \frac{D_m}{R_k^{\mathsf{fh}}},\ \forall k & \text{(42a)} \\ \frac{\varphi_{mk}^2}{\beta_{mk}} \ge \pi_{mk}\alpha_m,\ \forall m,k & \text{(42b)} \end{cases}$$

where $\boldsymbol{\varphi} \triangleq \{\varphi_{mk}\}_{\forall m,k}$ are new variables to tackle the product of $\pi_{mk}\alpha_m\beta_{mk}$. Constraint (42b) is non-convex. Similar to (33b), we have

$$\frac{2\varphi_{mk}^{(i)}\varphi_{mk}}{\beta_{mk}^{(i)}} - \frac{\varphi_{mk}^{2(i)}\beta_{mk}}{\big(\beta_{mk}^{(i)}\big)^2} \ge \frac{1}{2} \Big(\frac{\alpha_m^{(i)}}{\pi_{mk}^{(i)}} \pi_{mk}^2 + \frac{\pi_{mk}^{(i)}}{\alpha_m^{(i)}} \alpha_m^2 \Big),\ \forall m,k. \tag{43}$$

Algorithm 1 Proposed AO-IA based Algorithm for Solving the JCCO Problem (14)

Initialization: Set $i = 0$ and generate initial feasible points $\mathcal{S}_1^{(0)}$ and $\mathcal{S}_2^{(0)}$ to constraints in (25) and (47), respectively. Set the tolerance $\varepsilon = 10^{-3}$ and the maximum number of iterations $I^{\max}$.

1: **repeat**
2: Solve problem (25) for given $\mathcal{S}_2^{(i)}$ to obtain the optimal solution denoted by $(\mathbf{p}^\star, \mathbf{f}^\star, \mathbf{r}^\star, \boldsymbol{\tau}^\star)$ and update $\mathcal{S}_1^{(i+1)} := (\mathbf{p}^\star, \mathbf{f}^\star, \mathbf{r}^\star)$;
3: Solve problem (47) for given $\mathcal{S}_1^{(i+1)}$ to obtain the optimal solution denoted by $(\boldsymbol{\alpha}^\star, \boldsymbol{\beta}^\star, \boldsymbol{\pi}^\star, \check{\boldsymbol{\phi}}^\star, \hat{\boldsymbol{\phi}}^\star, \mathbf{r}^\star, \boldsymbol{\varphi}^\star, \boldsymbol{\tau}^\star)$ and update $\mathcal{S}_2^{(i+1)} := (\boldsymbol{\alpha}^\star, \boldsymbol{\beta}^\star, \boldsymbol{\pi}^\star, \check{\boldsymbol{\phi}}^\star, \hat{\boldsymbol{\phi}}^\star, \mathbf{r}^\star, \boldsymbol{\varphi}^\star)$;
4: Set $i := i + 1$;
5: **until** Convergence or $i > I^{\max}$
6: Recover binary values of $\boldsymbol{\pi}^\star$: $\pi_{mk}^\star = \lfloor \pi_{mk}^{(i)} + 0.5 \rfloor,\ \forall m, k$;
7: Repeat Steps 1-5 with fixed $\boldsymbol{\pi}^\star$ to refine the optimal solution;
8: **Output**: $(\boldsymbol{\alpha}^\star, \boldsymbol{\beta}^\star, \boldsymbol{\pi}^\star, \mathbf{p}^\star, \mathbf{f}^\star)$.

Convexity of (14g)*:* We first rewrite (14g) as

$$\frac{f^{\mathsf{cs}}}{\sum_{m\in\mathcal{M}}\sum_{k\in\mathcal{K}}\pi_{mk}\alpha_m\beta_{mk}C_m} \geq \sum_{m\in\mathcal{M}}\sum_{k\in\mathcal{K}}\pi_{mk}\alpha_m\beta_{mk}\lambda_m^{\mathsf{lo}} + \frac{1}{\tau^{\mathsf{cs}}} \tag{44}$$

which is equivalent to

$$\frac{f^{\mathsf{cs}}}{\sum_{m\in\mathcal{M}} C_m\hat{\phi}_m^2} \geq \sum_{m\in\mathcal{M}}\lambda_m^{\mathsf{lo}}\hat{\phi}_m^2 + \frac{1}{\tau^{\mathsf{cs}}} \tag{45}$$

by using (36b). It follows from (37a) that

$$f^{\mathsf{cs}}\left(\frac{2}{\sum\limits_{m\in\mathcal{M}} C_m(\hat{\phi}_m^{(i)})^2} - \frac{\sum\limits_{m\in\mathcal{M}} C_m\hat{\phi}_m^2}{\big(\sum\limits_{m\in\mathcal{M}} C_m(\hat{\phi}_m^{(i)})^2\big)^2}\right) \geq \sum_{m\in\mathcal{M}}\lambda_m^{\mathsf{lo}}\hat{\phi}_m^2 + \frac{1}{\tau^{\mathsf{cs}}}. \tag{46}$$

Summing up, we obtain the following approximate convex program of SP-2 solved at iteration i:

$$\textsf{SP-2: Convex}\ \underset{\substack{\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\pi},\boldsymbol{\tau},\check{\boldsymbol{\phi}},\hat{\boldsymbol{\phi}},\\ \mathbf{r},\boldsymbol{\varphi}|\mathbf{p}^{(i+1)},\mathbf{f}^{(i+1)}}}{\text{minimize}}\ \max_{\forall m\in\mathcal{M}}\big\{t_m(f_m^{\mathsf{lo},(i+1)}, \boldsymbol{\tau})\big\} \tag{47a}$$

$$\text{s.t. } (14\text{c}), (26\text{c}), (28), (29), (30), (31\text{a}), (32), (34),$$

$$(35), (37), (38), (41), (42\text{a}), (43), (46), \tag{47b}$$

which requires the per-iteration complexity of $\mathcal{O}\big(\sqrt{9M + 4MK + 3K}(3MK + 3M + 4)^2\big)$.

C. Proposed AO-IA based Algorithms

1) Proposed AO-IA based algorithm for solving the JCCO problem: Let denote by $\mathcal{S}_1^{(i)} \triangleq (\mathbf{p}^{(i)}, \mathbf{f}^{(i)}, \mathbf{r}^{(i)})$ and $\mathcal{S}_2^{(i)} \triangleq (\boldsymbol{\alpha}^{(i)}, \boldsymbol{\beta}^{(i)}, \boldsymbol{\pi}^{(i)}, \check{\boldsymbol{\phi}}^{(i)}, \hat{\boldsymbol{\phi}}^{(i)}, \mathbf{r}^{(i)}, \boldsymbol{\varphi}^{(i)})$ the feasible sets of (25) and (47) at iteration i, respectively. The overall algorithm for solving (14) is summarized in Algorithm 1.

The main drawback of solving problem (26) is that the exact binary solution of $\boldsymbol{\pi}$ is not guaranteed at optimum, resulting in an infeasible solution to the original problem (13). To overcome this issue, we consider Step 6 in Algorithm 1 using ceiling function to recover binary value of $\boldsymbol{\pi}$ as $\pi^{\star}_{mk} = \lfloor \pi^{(i)}_{mk} + 0.5 \rfloor, \ \forall m, k$. In Step 7, we repeat Steps 1-5 for given $\boldsymbol{\pi}^{\star}$ to minimize the performance loss due to Step 6. Algorithm 1 requires initial feasible points to start at the first iteration. The initial feasible points of $\mathcal{S}_1^{(0)}$ and $\mathcal{S}_2^{(0)}$ are generated as follows. Firstly, the transmission power and the processing rate are randomly generated with respect to constraint (13g). Secondly, the offloading portions and user association variables are initiated equally, i.e., $\alpha_m = 0.5$ and $\pi_{mk} = 0.5, \forall m, k$. Finally, we implement a validating function to guarantee that all constraints in (14) are satisfied before running the optimisation algorithm.

Convergence and complexity analysis: We recall that all the approximate functions presented in Section III-B satisfy IA properties listed in [31]. As proved in [24], Algorithm 1 generates sequences of the improved points of $\{\mathbf{p}^{(i)}, \mathbf{f}^{(i)}, \mathbf{r}^{(i)}\}$ and $\{\boldsymbol{\alpha}^{(i)}, \boldsymbol{\beta}^{(i)}, \boldsymbol{\pi}^{(i)}, \check{\boldsymbol{\phi}}^{(i)}, \hat{\boldsymbol{\phi}}^{(i)}, \mathbf{r}^{(i)}, \boldsymbol{\varphi}^{(i)}\}$ to SP-1 and SP-2, respectively, as well as sequences of non-increasing e2e latency values. In addition, the feasible sets of the convex programs (25) and (47) are connected and convex. As a result, the sequences $\{\mathbf{p}^{(i)}, \mathbf{f}^{(i)}\}$ and $\{\boldsymbol{\alpha}^{(i)}, \boldsymbol{\beta}^{(i)}, \boldsymbol{\pi}^{(i)}\}$ are guaranteed to arrive at least the local optimal solutions of SP-1 and SP-2, respectively. In term of the complexity analysis, in each iteration of Algorithm 1, the computational complexity of solving SP-2 dominates that of SP-1, especially in large-scale scenarios. Therefore, for a given number of I iterations that guarantee the convergence of Algorithm 1, its worst-case complexity is given as $\mathcal{O}\big(I\sqrt{9M + 4MK + 3K}(3MK + 3M + 4)^2\big)$.

2) Proposed AO-IA based algorithm for solving the parameterized JCCO problem: To improve the convergence speed of Algorithm 1, we incorporate a penalty function to tackle the uncertainty of binary variables of the relaxed SP-2 problem (26), inspired by [34], [35]. In particular, it is true that $\pi_{mk} \geq \pi^2_{mk}$ for any $\pi_{mk} \in [0,1], \forall m, k$. The equality holds if only if $\pi_{mk} = \{0, 1\}$. Without loss of optimality, $\boldsymbol{\pi} \in \Pi$ can be equivalently expressed as

$$\boldsymbol{\pi} \in \Pi \Leftrightarrow 0 \leq \pi_{mk} \leq 1 \ \& \ \pi_{mk} \leq \pi^2_{mk}, \ \ \forall m, k. \tag{48}$$

However, constraint $\pi_{mk} \leq \pi^2_{mk}$ is often infeasible to the relaxed SP-2 problem (26). Therefore, we remove this constraint but is spenalized by incorporating the function $\Psi(\boldsymbol{\pi}) = \sum_{m \in \mathcal{M}} \sum_{k \in \mathcal{K}} (\pi_{mk} - \pi^2_{mk})$ into the objective function of problem (26), which is always non-negative to guarantee the

Algorithm 2 Proposed AO-IA based Algorithm for Solving the parameterized JCCO Problem (14)

Initialization: Set $i = 0$ and randomly generate initial feasible points $\mathcal{S}_1^{(0)}$ and $\mathcal{S}_2^{(i)}$ to constraints in (25) and (50), respectively. Set the tolerance $\varepsilon = 10^{-3}$ and the maximum number of iterations $I^{\max}$.

1: **repeat**
2: Solve problem (25) for given $\mathcal{S}_2^{(i)}$ to obtain the optimal solution denoted by $(\mathbf{p}^\star, \mathbf{f}^\star, \mathbf{r}^\star, \boldsymbol{\tau}^\star)$ and update $\mathcal{S}_1^{(i+1)} := (\mathbf{p}^\star, \mathbf{f}^\star, \mathbf{r}^\star)$;
3: Solve problem (50) for given $\mathcal{S}_1^{(i+1)}$ to obtain the optimal solution denoted by $(\boldsymbol{\alpha}^\star, \boldsymbol{\beta}^\star, \boldsymbol{\pi}^\star, \check{\boldsymbol{\phi}}^\star, \hat{\boldsymbol{\phi}}^\star, \mathbf{r}^\star, \boldsymbol{\varphi}^\star, \boldsymbol{\tau}^\star)$ and update $\mathcal{S}_2^{(i+1)} := (\boldsymbol{\alpha}^\star, \boldsymbol{\beta}^\star, \boldsymbol{\pi}^\star, \check{\boldsymbol{\phi}}^\star, \hat{\boldsymbol{\phi}}^\star, \mathbf{r}^\star, \boldsymbol{\varphi}^\star)$;
4: Set $i := i + 1$;
5: **until** Convergence or $i > I^{\max}$
6: Recover binary values of $\boldsymbol{\pi}^\star$: $\pi_{mk}^\star = \lfloor \pi_{mk}^{(i)} + 0.5 \rfloor, \ \forall m, k$;
7: Repeat Steps 1-5 with fixed $\boldsymbol{\pi}^\star$ to refine the optimal solution;
8: **Output**: $(\boldsymbol{\alpha}^\star, \boldsymbol{\beta}^\star, \boldsymbol{\pi}^\star, \mathbf{p}^\star, \mathbf{f}^\star)$.

satisfaction in (48). The parameterized SP-2 (PSP-2) problem is expressed as

$$\text{PSP-2:} \underset{\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\pi},\boldsymbol{\tau}|\mathbf{p}^{(i+1)},\mathbf{f}^{(i+1)}}{\text{minimize}} \ \max_{\forall m \in \mathcal{M}} \big\{ t_m(f_m^{\mathsf{lo},(i+1)}, \boldsymbol{\tau}) \big\} + \eta^{(i)} \Psi(\boldsymbol{\pi}), \ \text{s.t.} \ (26\text{b}), (26\text{c}) \tag{49}$$

where $\eta^{(i)} > 0$ is the penalty parameter at iteration i. With an appropriate possible and sufficient large value of $\eta^{(i)}$, problem (49) is equivalent to (16) [34]. All the non-convex constraints in (49) were addressed in Section III-B, while the concave function $\Psi(\boldsymbol{\pi}) = \sum_{m\in\mathcal{M}} \sum_{k\in\mathcal{K}} (\pi_{mk} - \pi_{mk}^2)$ is directly convexified by (B.3). In particular, the approximate convex program of the PSP-2 (49) solved at iteration i is given as

$$\text{PSP-2: Convex} \underset{\substack{\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\pi},\boldsymbol{\tau},\check{\boldsymbol{\phi}},\hat{\boldsymbol{\phi}},\\ \mathbf{r},\boldsymbol{\varphi}|\mathbf{p}^{(i+1)},\mathbf{f}^{(i+1)}}}{\text{minimize}} \ \max_{\forall m \in \mathcal{M}} \big\{ t_m(f_m^{\mathsf{lo},(i+1)}, \boldsymbol{\tau}) \big\} + \eta^{(i)} \Psi^{(i)}(\boldsymbol{\pi}) \tag{50a}$$

$$\text{s.t.} \ (47\text{b}) \tag{50b}$$

where $\Psi^{(i)}(\boldsymbol{\pi}) \triangleq \sum_{m\in\mathcal{M}} \sum_{k\in\mathcal{K}} \big(\pi_{mk} - 2\pi_{mk}\pi_{mk}^{(i)} + (\pi_{mk}^{(i)})^2\big)$ is the first-order Taylor approximation of $\Psi^{(i)}(\boldsymbol{\pi})$. The proposed AO-IA based algorithm for solving the parameterized JCCO problem (14) is summarized in Algorithm 2.

IV. Sub-Optimal Designs

The major complexity of solving the JCCO problem (13) comes from the optimization of user associations $\boldsymbol{\pi}$. Towards low-complexity solutions, we consider two sub-optimal designs in which user associations are given in advance.

A. Optimization Designs

1) Best Channel Selection (BCS)-based Design: In the BCS-based design, UE m selects ES k with the best channel condition (i.e., with the highest channel gain). In particular, the optimal solution $\boldsymbol{\pi}^\star$ is found as:

$$\boldsymbol{\pi}^\star = \Big\{\pi^\star_{mk}, \forall m, k \Big| \pi^\star_{mk} = 1 \text{ if } k^* = \arg\max_{k\in\mathcal{K}}\{\|\hat{\mathbf{h}}_{mk}\|^2\}, \forall m; \text{ otherwise } \pi^\star_{mk} = 0, \forall k\Big\}. \quad (51)$$

The purpose of this design is to reduce the wireless transmission latency without a complicated optimization over ES-UE associations, which is the main bottleneck of the considered system model.

2) Random User Association (RUA)-based Design: In this scheme, we randomly associate UE k with any ES subject to constraint UE policies, such as

$$\boldsymbol{\pi}^\star = \Big\{\pi^\star_{mk} \texttt{ is randomly generated,} \Big| \text{s.t. } \pi^\star_{mk} \in \{0,1\} \,\&\, \sum_{k\in\mathcal{K}} \pi^\star_{mk} = 1, \forall m, k\Big\}. \quad (52)$$

Given $\boldsymbol{\pi}^\star$ in (51) and (52), we reformulate the JCCO problem as follows

$$\underset{\boldsymbol{\alpha},\boldsymbol{\beta},\mathbf{p},\mathbf{f}|\boldsymbol{\pi}^\star}{\text{minimize}} \ \max_{\forall m\in\mathcal{M}} \big\{t_m\big(f^{\mathsf{lo}}_m, \mathbf{p}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\pi}^\star\big)\big\} \quad (53a)$$

$$\text{s.t. } t_m\big(f^{\mathsf{lo}}_m, \mathbf{p}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\pi}^\star\big) \le T^{\max}_m, \forall m \quad (53b)$$

$$E_m(\alpha_m, f^{\mathsf{lo}}_m, \mathbf{p}, \boldsymbol{\pi}^\star) \le E^{\max}_m, \forall m \quad (53c)$$

$$R_m\left(\mathbf{p}, \boldsymbol{\pi}^\star_m\right) \ge \sum_{k\in\mathcal{K}} \pi^\star_{mk} R^{\min}_m, \forall m \quad (53d)$$

$$\sum_{m\in\mathcal{M}} \pi^\star_{mk}\alpha_m\lambda^{\mathsf{lo}}_m \le \frac{f^{\mathsf{es}}_k}{\sum\limits_{m\in\mathcal{M}} \pi^\star_{mk}\alpha_m\left(1-\beta_{mk}\right)C_m}, \forall k \quad (53e)$$

$$\sum_{m\in\mathcal{M}}\sum_{k\in\mathcal{K}} \pi^\star_{mk}\alpha_m\beta_{mk}\lambda^{\mathsf{lo}}_m \le \frac{f^{\mathsf{cs}}}{\sum\limits_{m\in\mathcal{M}}\sum\limits_{k\in\mathcal{K}} \pi^\star_{mk}\alpha_m\beta_{mk}C_m} \quad (53f)$$

$$\boldsymbol{\alpha}, \boldsymbol{\beta} \in \mathscr{D}, \mathbf{p} \in \mathscr{P}, \mathbf{f} \in \mathscr{F} \quad (53g)$$

which will be used to find the optimal solutions for both BCS and RUA-based designs.

B. Proposed Algorithm for Solving (53)

Similar to Section III-B, problem (53) can be transformed and decomposed into two sub-problems, one is similar to the SP-1 (15) and the other is the simplified form of SP-2 (16), called SSP-2, as follows

$$\text{SSP-2:} \underset{\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\tau}|\mathbf{p}^{(i+1)},\mathbf{f}^{(i+1)},\boldsymbol{\pi}^{\star}}{\text{minimize}} \quad \max_{\forall m\in\mathcal{M}} \left\{t_m(f_m^{\mathsf{lo},(i+1)},\boldsymbol{\tau})\right\} \tag{54a}$$

$$\text{s.t. } (13c),(13e),(13f),(14c),(14d),(14e),(14f),(14g) \tag{54b}$$

$$\boldsymbol{\alpha},\boldsymbol{\beta}\in\mathscr{D}. \tag{54c}$$

The non-convex constraints include (13e), (13f), (14e), (14f) and (14g), which can be convexified by the developments presented in Section III-B.

For given $(\mathbf{p}^{(i+1)},\mathbf{f}^{(i+1)},\boldsymbol{\pi}^{\star})$, (13e) is directly approximated by (34) and (35) as

$$\begin{cases} \sum_{m\in\mathcal{M}} \lambda_m^{\mathsf{lo}}\pi_{mk}^{\star}\alpha_m \le f_k^{\mathsf{es}}\left(\dfrac{2}{\sum_{m\in\mathcal{M}} C_m(\check{\phi}_{mk}^{(i)})^2} - \dfrac{\sum_{m\in\mathcal{M}} C_m\check{\phi}_{mk}^2}{\left(\sum_{m\in\mathcal{M}} C_m(\check{\phi}_{mk}^{(i)})^2\right)^2}\right),\ \forall k & (55a)\\ \dfrac{2\check{\phi}_{mk}^{(i)}\check{\phi}_{mk}}{1-\beta_{mk}^{(i)}} - \dfrac{(\check{\phi}_{mk}^{(i)})^2(1-\beta_{mk})}{(1-\beta_{mk}^{(i)})^2} \ge \pi_{mk}^{\star}\alpha_m,\ \forall m,k. & (55b)\end{cases}$$

Following (37) and (41), we approximate (13f) and (14e) as

$$\begin{cases} \sum_{m\in\mathcal{M}} \lambda_m^{\mathsf{lo}}\hat{\phi}_m^2 \le f^{\mathsf{cs}}\left(\dfrac{2}{\sum_{m\in\mathcal{M}} C_m(\hat{\phi}_m^{(i)})^2} - \dfrac{\sum_{m\in\mathcal{M}} C_m\hat{\phi}_m^2}{\left(\sum_{m\in\mathcal{M}} C_m(\hat{\phi}_m^{(i)})^2\right)^2}\right) & (56a)\\ \dfrac{2\hat{\phi}_m^{(i)}\hat{\phi}_m}{\alpha_m^{(i)}} - \dfrac{(\hat{\phi}_m^{(i)})^2\alpha_m}{(\alpha_m^{(i)})^2} \ge \sum_{k\in\mathcal{K}}\pi_{mk}^{\star}\beta_{mk},\ \forall m. & (56b)\end{cases}$$

and

$$f_k^{\mathsf{es}}\left(\frac{2}{\sum_{m\in\mathcal{M}} C_m(\check{\phi}_{mk}^{(i)})^2} - \frac{\sum_{m\in\mathcal{M}} C_m\check{\phi}_{mk}^2}{\left(\sum_{m\in\mathcal{M}} C_m(\check{\phi}_{mk}^{(i)})^2\right)^2}\right) \ge \sum_{m\in\mathcal{M}}\lambda_m^{\mathsf{lo}}\pi_{mk}^{\star}\alpha_m + \frac{1}{\tau^{\mathsf{es}}},\ \forall k. \tag{57}$$

In the same manner as in (42), constraint (14f) is expressed as (42) where the second constrain (i.e., $\frac{\varphi_{mk}^2}{\beta_{mk}} \ge \pi_{mk}$) can be easily approximated as

$$\frac{2\varphi_{mk}^{(i)}\varphi_{mk}}{\beta_{mk}^{(i)}} - \frac{\varphi_{mk}^{2(i)}\beta_{mk}}{\left(\beta_{mk}^{(i)}\right)^2} \ge \pi_{mk}^{\star}\alpha_m,\ \forall m,k. \tag{58}$$

Constraint (14g) was addressed in (46). We solve the following convex program at iteration i:

$$\text{SSP-2: Convex} \underset{\substack{\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\tau},\check{\boldsymbol{\phi}},\hat{\boldsymbol{\phi}},\boldsymbol{\varphi}|\mathbf{p}^{(i+1)},\\ \mathbf{f}^{(i+1)},\boldsymbol{\pi}^{\star}}}{\text{minimize}} \quad \max_{\forall m\in\mathcal{M}} \left\{t_m(f_m^{\mathsf{lo},(i+1)},\boldsymbol{\tau})\right\} \tag{59a}$$

$$\text{s.t. } (13c),(14c),(14d),(42a),(46),(55),(56),(57),(58) \tag{59b}$$

where the per-iteration complexity of solving SSP-2 is $\mathcal{O}\big(\sqrt{4M+2MK+2K+2}(2MK+2M+4)^2\big)$, which is seen lower than that of the convex program (47). The proposed algorithm is summarized in Algorithm 3 without the need of re-optimization after the exact binary recovery,

Algorithm 3 Proposed AO-IA based Algorithm for Solving Problem (53)

Initialization: Set $i = 0$ and randomly generate initial feasible points to constraints in (25) and (59), respectively. Set the tolerance $\varepsilon = 10^{-3}$ and the maximum number of iterations $I^{\max}$.
1: **repeat**
2: Solve problem (25) for given $\tilde{\mathcal{S}}_2^{(i)}$ to obtain the optimal solution denoted by $(\mathbf{p}^\star, \mathbf{f}^\star, \mathbf{r}^\star, \boldsymbol{\tau}^\star)$ and update $\mathcal{S}_1^{(i+1)} := (\mathbf{p}^\star, \mathbf{f}^\star, \mathbf{r}^\star)$;
3: Solve problem (59) for given $\mathcal{S}_1^{(i+1)}$ to obtain the optimal solution denoted by $(\boldsymbol{\alpha}^\star, \boldsymbol{\beta}^\star, \check{\boldsymbol{\phi}}^\star, \hat{\boldsymbol{\phi}}^\star, \boldsymbol{\varphi}^\star, \boldsymbol{\tau}^\star)$ and update $\tilde{\mathcal{S}}_2^{(i+1)} := (\boldsymbol{\alpha}^\star, \boldsymbol{\beta}^\star, \check{\boldsymbol{\phi}}^\star, \hat{\boldsymbol{\phi}}^\star, \boldsymbol{\varphi}^\star)$;
4: Set $i := i + 1$;
5: **until** Convergence or $i > I^{\max}$
6: **Output**: $(\boldsymbol{\alpha}^\star, \boldsymbol{\beta}^\star, \mathbf{p}^\star, \mathbf{f}^\star)$.

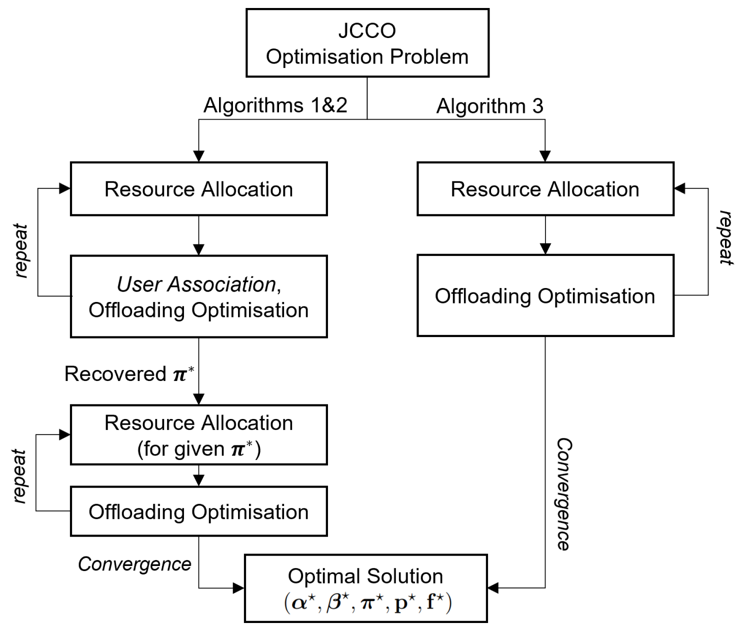


Fig. 2: The optimisation procedure for solving the JCCO problem.

where $\tilde{\mathcal{S}}_2^{(i)} \triangleq (\boldsymbol{\alpha}^{(i)}, \boldsymbol{\beta}^{(i)}, \check{\boldsymbol{\phi}}^{(i)}, \hat{\boldsymbol{\phi}}^{(i)}, \boldsymbol{\varphi}^{(i)})$. Overall, the optimisation procedure of the proposed algorithms is clearly illustrated in Fig. 2.

V. Numerical Results

In this section, we provide numerical examples to quantitatively evaluate the performance of the proposed algorithms.

A. Simulation Setup and Parameters

We consider a small-cell scenario where all APs (ESs) and UEs are located within an area of 100×100 m [4]. ESs are located at (50, 33) and (50, 66) for $K = 2$ and $(50, 20)$, $(50, 40)$, $(50, 60)$, $(50, 80)$ for $K = 4$. The large-scale fading of the channel between UE m and AP k is

TABLE I: Simulation Parameters [4], [36], [37]

Parameter	Value
System bandwidth, B	10 MHz
Noise power spectral density	-174 dBm/Hz
Transmission time interval, Δ_t	0.01 ms
Maximum blocklength, $N = \Delta_t B$	100
Number of UEs, M	10
Pilot transmit power, $p^p_m, \forall m$	10 dBm
Maximum processing rate of UEs, $F^{\max}_m$	3 GHz
Fronthaul capacity, $R_k \equiv R^{\mathsf{fh}}_k, \forall k$	1 Gbps
Minimum rate requirement, $R^{\min} \equiv R^{\min}_m, \forall m$	1 Mbps
Maximum energy consumption, $E^{\max} \equiv E^{\max}_m, \forall m$	1 Joule
Effective capacitance coefficient, $\theta_m, \forall m$	10^{-27} Watt.s^3/cycle^3
Penalty parameter, η	0.03

modeled as $g_{mk} = 10^{\mathsf{PL}(d_{mk})/10}$, where $\mathsf{PL}(d_{mk}) = -35.3 - 37.6 \log_{10} d_{mk}$ denotes the path loss in dB which is a function of the distance d_{mk} [32]. The number of antennas at each AP is set to $L = 8$. We assume that all UEs have the same power budget, i.e., $P^{\max}_m = 23$ dBm $\forall m$ [3]. The URLLC decoding error probability is set to $\epsilon_m = 10^{-9}, \forall m$ [25].

Following [38], we set the CPU cycles of ESs and CS to 25 and 30 Giga cycles/s, respectively. For UE m, the input task size and the required computation resource are set to $D_m = 100$ KB and $C_m = 800 \times 10^6$ (cycles) [4], respectively. The total e2e latency requirement of each UE is given as $T^{\max}_m = 2$s $\forall m$ [3]. The mean arrival rate of tasks is set to $\lambda^{\mathsf{lo}}_m = 10$ (task/s), $\forall m$ [29]. Unless specifically stated otherwise, other parameters are given in Table I. We implement the proposed algorithms in MATLAB environment and all the convex programs are solved by **SDPT3** solver in the modeling toolbox **CVX**.

To demonstrate the effectiveness of joint communication and computation offloading, we compare the performance of the proposed algorithms with the following three known schemes:

- "Fixed Power": Under the same setup with the JCCO problem, the transmit power of all UEs is fixed and set equal to the maximum power budget as $p_m = P^{\max}_m, \forall m$, subject to the energy constraint (13c). This scheme is considered in [30].
- "Fixed Frequency (Fixed Freq.)": Each UE is configured with its maximum processing rate, i.e., $f^{\mathsf{lo}}_m = F^{\max}_m, \forall m$ [2], [3], subject to the energy constraint (13c).
- "Without Cloud (W/o Cloud)": computation tasks are processed at UEs and ESs only [3].

The formulated problems of the above schemes can be solved directly by Algorithm 2 after some simple modifications. The results of two sub-optimal designs obtained by Algorithm 3 are labelled to as Algorithm 3-BCS and Algorithm 3-RUA.

B. Numerical Results and Discussions

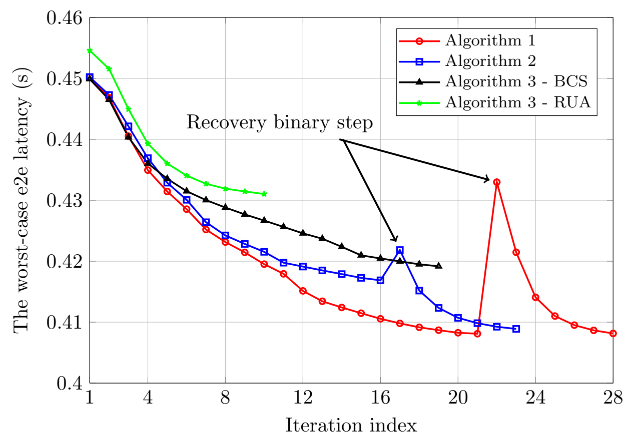
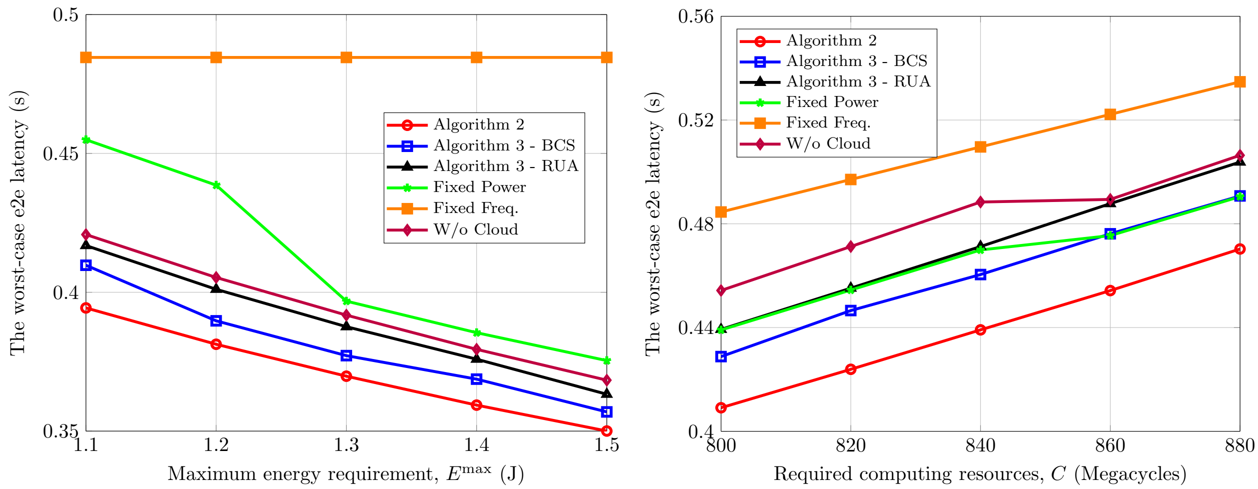


Fig. 3: Convergence behavior of the proposed algorithms for $M = 10$ UEs and $K = 2$ ESs.

Algorithm convergence: In Fig. 3, we illustrate the convergence behavior of the proposed algorithm over one random channel realization for $M = 10$ UEs and $K = 2$ ESs. As can bee seen from Fig. 3 that all the proposed algorithms generate sequences of non-increasing e2e latency values and converge within tens of iterations. The e2e latency of Algorithms 1 and 2 is degraded at iterations 22 and 17, respectively, due to the recovery binary step (Step 6). This also confirms the important role of Step 7 to refine the optimal solutions of $(\boldsymbol{\alpha}^\star, \boldsymbol{\beta}^\star, \mathbf{p}^\star, \mathbf{f}^\star)$, which achieves the same objective value but guaranteeing the exact binary solution of $\boldsymbol{\pi}^\star$. Algorithm 3 of the two sub-optimal designs requires fewer iterations to converge since $\boldsymbol{\pi}^\star$ is already known in advance. Another interesting observation is that Algorithm 2 converges faster than Algorithm 1, which is attributed to the fact that the incorporation of the penalty function helps speed up the optimal user association searching. Since Algorithms 1 and 2 offer the same objective value, we only show the performance for Algorithm 2 in the following numerical results.

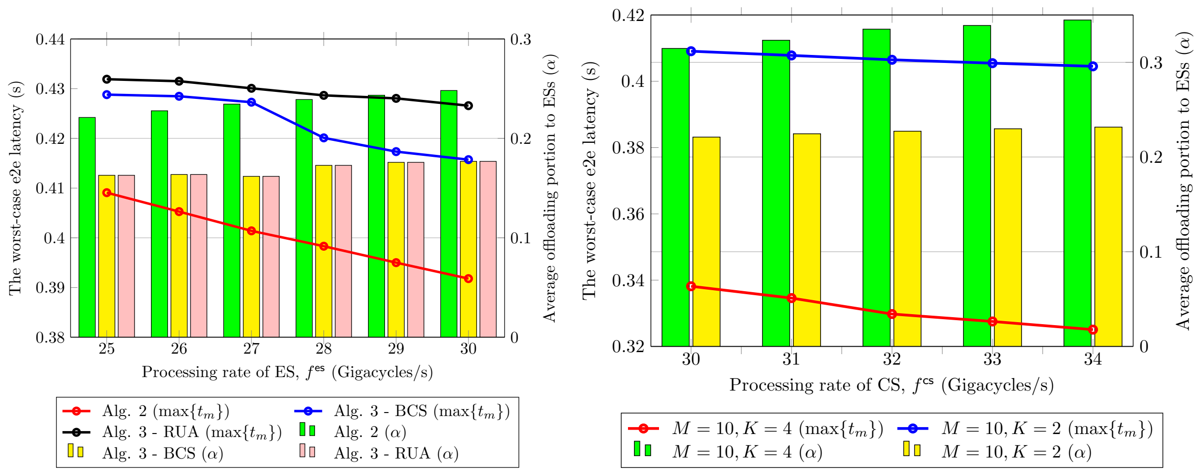


(a) Maximum energy consumption requirement

(b) Required computation resource

Fig. 4: The worst-case e2e latency of different resource allocation schemes versus (a) the maximum energy consumption requirement $E^{\max}$, and (b) the required computation resource $C \equiv C_m, \forall m$, with $M = 10$ UEs and $K = 2$ ESs.

Impact of the energy consumption and required computation resource: In Fig. 4(a), we investigate the impact of the maximum energy consumption requirement $E^{\max}$ on the e2e latency of different resource allocation schemes. As seen, the higher the energy consumption requirement, the lower the latency is achieved by the considered schemes, except the fixed frequency (processing rate). This is because a large value of $E^{\max}$ allows UEs to allocate higher processing rate f_m^{lo} and transmit power p_m to reduce the worst-case e2e latency while still meeting the energy constraint (13c). The proposed algorithms provide significant performance gains over the baseline schemes in terms of e2e latency. In addition, Algorithm 2 offers the lowest e2e latency performance which clearly confirms the effectiveness of jointly optimizing offloading probabilities, processing rates, user association policies and power control. Fig. 4(b) illustrates the e2e latency versus the required computation resource $C_m, \forall m$. Unsurprisingly, the e2e latency of all the considered schemes increases when C_m increases. We recall that a higher value of C_m will not only increase processing latency at UEs, ESs and CS but also force UEs to scale down their processing rate and transmit power to satisfy constraint (13c), resulting in higher total latency. Again, Algorithm 2 still provides the lowest e2e latency amongst all the considered schemes.



(a) Latency versus ESs' processing rate

(b) Latency versus CS's processing rate

Fig. 5: The worst-case e2e latency of different resource allocation schemes versus (a) ESs' processing rate, and (b) CS's processing rate $C \equiv C_m, \forall m$, with $M = 10$ UEs and $K = \{2, 4\}$ ESs.

Impact of offloading factor and processing rate: In hierarchical edge-cloud systems, the processing rates of ESs and CS have a strong impact on the system performance. In particular, the higher the processing rate, the higher the offloading portion from UEs to ESs. To verify this, we first show the e2e latency and average offloading portion from UEs to ESs versus ESs' processing rate $f^{\mathsf{es}} \equiv f_k^{\mathsf{es}}, \forall k$ in Fig. 5a. The offloading portion of UEs slightly increases when ESs have larger computation resource, f^{es}. An important observation is that Algorithm 2 always offloads the higher portion of computation tasks, compared to sub-optimal schemes, thanks to optimal user association policies, leading to lower e2e latency.

Fig. 5b studies the impact of the maximum processing rate at CS f^{cs}, on the e2e latency of Algorithm 2. As can be seen from the figure that, when the CS's processing rate increases from 30 to 34 gigacycles/s, the e2e latency gradually reduces in both scenarios of $K = 2$ ESs and $K = 4$ ESs. In addition, Fig. 5b also indicates that the average offloading portion of UEs to ESs continuously increases as CS is equipped with more powerful computing capacity.

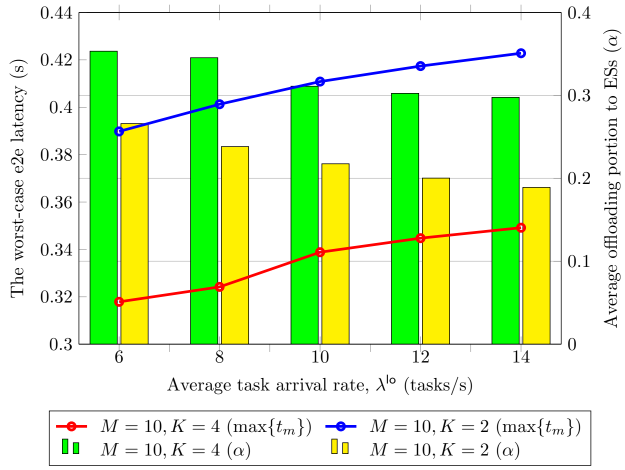


Fig. 6: The worst-case e2e latency and average offloading portion from UEs to ESs of Algorithm 2 versus the mean task arrival rate $\lambda^{\mathsf{lo}} \equiv \lambda^{\mathsf{lo}}_m, \forall m$, with $M = 10$ UEs and $K = \{2, 4\}$ ESs.

Impact of the arrival data rate: Fig. 6 depicts the worst-case e2e latency and average offloading portion from UEs to ESs of Algorithm 2 for different values of the task arrival rate $\lambda^{\mathsf{lo}} \equiv \lambda^{\mathsf{lo}}_m, \forall m$, with $M = 10$ UEs and $K = \{2, 4\}$ ESs. As expected, the e2e latency increases slightly when λ^{lo} increases, while the offloading portion decreases gradually to guarantee queuing stability constraints of ESs and CS. For instance, for the scenario with $M = 10$ UEs, $K = 4$ ESs and the mean task arrival rate is set to $\lambda^{\mathsf{lo}} = 14$ tasks/s, the worst-case e2e latency rises above 0.35 s, while the average offloading portion ($\boldsymbol{\alpha}$) is adjusted down to 30%.

VI. Conclusion

In this paper, we have investigated joint communication and computation task offloading in URLLC-based hierarchical edge-cloud systems. To address the practical issues of minimizing the worst-case e2e latency amongst UEs, we have formulated the optimization problem of jointly optimizing offloading probabilities, processing rates, user association policies and power control, taking into account UEs' maximum delay and energy consumption requirements, and queueing stability conditions at ESs and CS. To that end, we have proposed an alternating optimization framework to efficiently solve the formulated problem in an iterative manner. We have adopted a judicious approach by resorting to inner approximation method, where new convex approximate functions are developed to tackle non-convex constraints. In addition, two sub-optimal designs under given user association policies are proposed to obtain low-complexity solutions. Our proposed algorithms made evident the highly complex relationship between the communication and computation parameters. Finally, we provided extensive numerical results to demonstrate the fast convergence of the proposed algorithms as well as the significant performance gain achieved by joint optimization of the communication and computation variables in hierarchical edge-cloud systems.

Appendix A
Derivation of SINR in (4)

We follow steps similar to those in [25] to derive the average SINR in (4). For $x_{m,i,k} \triangleq \frac{\hat{\mathbf{h}}_{mk}^H \hat{\mathbf{h}}_{ik}}{\|\hat{\mathbf{h}}_{mk}\|}$, $y_{m,i,k} \triangleq \frac{\hat{\mathbf{h}}_{mk}^H \tilde{\mathbf{h}}_{ik}}{\|\hat{\mathbf{h}}_{mk}\|}$, and $\Phi_m(\mathbf{p}, \boldsymbol{\pi}_m)$ defined in (3), it is true that

$$\mathbb{E}\Big\{\frac{1}{\gamma_m(\mathbf{p}, \boldsymbol{\pi}_m)}\Big\} = \mathbb{E}\Bigg\{\frac{\Phi_m(\mathbf{p}, \boldsymbol{\pi}_m)}{\sum\limits_{k\in\mathcal{K}} \pi_{mk} p_m \|\hat{\mathbf{h}}_{mk}\|^2}\Bigg\}. \tag{A.1}$$

Conditioned on $\hat{\mathbf{h}}_{mk}$, we have that $x_{m,i,k}$, and $y_{m,i,k}$ are Gaussian random variables with zeros mean and variance σ_{mk}^2 and δ_{mk}^2, respectively. In addition, $x_{m,i,k}$, and $y_{m,i,k}$ are independent of $\hat{\mathbf{h}}_{mk}$. Thus, it follows that

$$\mathbb{E}\Big\{\gamma_m(\mathbf{p}, \boldsymbol{\pi}_m)^{-1}\Big\} = \bar{\Phi}_m(\mathbf{p}, \boldsymbol{\pi}_m)\mathbb{E}\Big\{\Big(\sum_{k\in\mathcal{K}} \pi_{mk} p_m \|\hat{\mathbf{h}}_{mk}\|^2\Big)^{-1}\Big\} \tag{A.2}$$

where $\bar{\Phi}_m(\mathbf{p}, \boldsymbol{\pi}_m)$ is defined in (4). By using the identity $\mathbb{E}\{\mathrm{tr}(\mathbf{W}^{-1})\} = \frac{m}{n-m}$, where $\mathbf{W} \sim \mathcal{W}_m(n, \mathbf{I}_n)$ is an $m \times m$ central complex Wishart matrix $n(n-m)$ degrees of freedom, we have

$$\mathbb{E}\Big\{\Big(\sum_{k\in\mathcal{K}} \pi_{mk} p_m \|\hat{\mathbf{h}}_{mk}\|^2\Big)^{-1}\Big\} = 1\Big/\Big(\sum_{k\in\mathcal{K}} \pi_{mk} p_m (L-1)\sigma_{mk}^2\Big), \text{ for } L \geq 2. \tag{A.3}$$

By substituting (A.3) into (A.2), we obtain the approximated SINR $\bar{\gamma}_m(\mathbf{p}, \boldsymbol{\pi}_m)$ in (4).

Appendix B
Inner Approximate Inequalities

We now provide some fundamental inequalities studied in [32], [39] based on the IA properties [31], which are used to approximate non-convex parts.

1) For all $x > 0, y > 0$, $\bar{x} > 0$ and $\bar{y} > 0$, the function $\ln(1 + x/y)$ is innerly approximated around the point $(\bar{x} > 0, \bar{y} > 0)$ as [32]

$$\ln\big(1 + x/y\big) \geq a - b/x - cy \tag{B.1}$$

where $a \triangleq \ln\big(1 + \frac{\bar{x}}{\bar{y}}\big) + 2\frac{\bar{x}}{\bar{x}+\bar{y}} > 0, b \triangleq \frac{\bar{x}^2}{\bar{x}+\bar{y}} > 0$, and $c \triangleq \frac{\bar{x}}{(\bar{x}+\bar{y})\bar{y}} > 0$.

2) For the convex function $f(x) = 1/x$ on the domain $x > 0$, its lower bounding concave function around the point $\bar{x}$ is

$$f(x) \geq f(\bar{x}) + \frac{\partial f(x)}{\partial(x)}\Big|_{x=\bar{x}} (x - \bar{x}) = \frac{1}{\bar{x}} - \frac{1}{\bar{x}^2}(x - \bar{x}) = \frac{2}{\bar{x}} - \frac{x}{\bar{x}^2}. \tag{B.2}$$

3) For the square-over-linear function $f(x, y) = x^2/y$ that is convex on $x \in \mathbb{R}, y > 0$, its lower bounding concave function around the point $(\bar{x}, \bar{y})$ is given as

$$f(x,y) \geq f(\bar{x}, \bar{y}) + \frac{\partial f(x,y)}{\partial(x)}\Big|_{x=\bar{x}} (x - \bar{x}) + \frac{\partial f(x,y)}{\partial(y)}\Big|_{y=\bar{y}} (y - \bar{y}) = \frac{2\bar{x}}{\bar{y}}x - \frac{\bar{x}^2}{\bar{y}^2}y. \tag{B.3}$$

4) The convex function $g(x) = 1/x^2$ with $x \in \mathbb{R}$ is innerly approximated around the point $\bar{x} \in \mathbb{R}$ as

$$g(x) \geq g(\bar{x}) + \frac{\partial g(x)}{\partial(x)}\Big|_{x=\bar{x}} (x - \bar{x}) = \frac{1}{\bar{x}^2} - \frac{2}{\bar{x}^3}(x - \bar{x}) = \frac{3}{\bar{x}^2} - \frac{2x}{\bar{x}^3}. \tag{B.4}$$

5) The upper bounding convex function of the product xy with $x > 0$ and $y > 0$ arround the point $(\bar{x}, \bar{y})$ is [40, Eq. (B1)]:

$$xy \leq \frac{1}{2}\Big(\frac{\bar{y}}{\bar{x}}x^2 + \frac{\bar{x}}{\bar{y}}y^2\Big). \tag{B.5}$$

6) Finally, for the concave function $h(x) = \sqrt{x}$ over $x > 0$, its upper bounding convex function at the point $\bar{x}$ is

$$h(x) \leq h(\bar{x}) + \frac{\partial h(x)}{\partial(x)}\Big|_{x=\bar{x}} (x - \bar{x}) = \frac{\sqrt{\bar{x}}}{2} + \frac{x}{2\sqrt{\bar{x}}}. \tag{B.6}$$

APPENDIX C
DERIVATION OF LEMMA 2

We first rewrite the SINR of UE m as $\gamma_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}) = p_m/q_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)})$. By applying the inequality (B.1) for $x = p_m$, $y = q_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)})$, $\bar{x} = p_m^{(i)}$, and $\bar{y} = q_m(\mathbf{p}^{(i)}, \boldsymbol{\pi}_m^{(i)})$, we have

$$G_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}) \geq a_m^{(i)} - b_m^{(i)}/p_m - c_m^{(i)} q_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}) \triangleq \mathcal{G}_m^{(i)}(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}) \tag{C.1}$$

where $a_m^{(i)} = \ln\big(1 + \frac{p_m^{(i)}}{q_m^{(i)}(\mathbf{p}, \boldsymbol{\pi}_m^{(i)})}\big) + 2\frac{p_m^{(i)}}{p_m^{(i)} + q_m^{(i)}(\mathbf{p}, \boldsymbol{\pi}_m^{(i)})}$, $b_m^{(i)} = \frac{(p_m^{(i)})^2}{p_m^{(i)} + q_m^{(i)}(\mathbf{p}, \boldsymbol{\pi}_m^{(i)})}$ and $c_m^{(i)} = \frac{p_m^{(i)}}{\big(q_m^{(i)}\big(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}\big) + p_m^{(i)}\big) q_m^{(i)}(\mathbf{p}, \boldsymbol{\pi}_m^{(i)})}$.

To find an upper bounding convex function approximation of $V_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)})$, we apply the inequality (B.6) for $x = 1 - 1/\big(1 + \bar{\gamma}_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)})\big)^2$ and $\bar{x} = 1 - 1/\big(1 + \bar{\gamma}_m(\mathbf{p}^{(i)}, \boldsymbol{\pi}_m^{(i)})\big)^2$, yielding

$$V_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}) \leq d_m^{(i)} - \frac{e_m^{(i)}}{\bar{\gamma}_m^2(\mathbf{p}, \boldsymbol{\pi}_m^{(i)})} = d_m^{(i)} - e_m^{(i)} \frac{q_m^2(\mathbf{p}, \boldsymbol{\pi}_m^{(i)})}{\big(q_m(\mathbf{p}, \boldsymbol{\pi}_m^{(i)}) + p_m\big)^2} \tag{C.2}$$

where

$$d_m^{(i)} = 0.5\sqrt{V_m(\mathbf{p}^{(i)}, \boldsymbol{\pi}_m^{(i)})} + 0.5/\sqrt{V_m(\mathbf{p}^{(i)}, \boldsymbol{\pi}_m^{(i)})}, \text{ and } e_m^{(i)} = 0.5/\sqrt{V_m(\mathbf{p}^{(i)}, \boldsymbol{\pi}_m^{(i)})}. \tag{C.3}$$

The function $\frac{q_m^2(\mathbf{p},\boldsymbol{\pi}_m^{(i)})}{\left(q_m(\mathbf{p},\boldsymbol{\pi}_m^{(i)})+p_m\right)^2}$ in (C.2) is still not convex [32], which can be further approximated by using inequalities (B.2) and (B.3) as

$$\frac{q_m^2(\mathbf{p},\boldsymbol{\pi}_m^{(i)})}{q_m(\mathbf{p},\boldsymbol{\pi}_m^{(i)})+p_m}\frac{1}{q_m(\mathbf{p},\boldsymbol{\pi}_m^{(i)})+p_m} \geq \frac{2}{q_m(\mathbf{p}^{(i)},\boldsymbol{\pi}_m^{(i)})+p_m^{(i)}}\Big(\frac{2q_m(\mathbf{p}^{(i)},\boldsymbol{\pi}_m^{(i)})q_m(\mathbf{p},\boldsymbol{\pi}_m^{(i)})}{q_m(\mathbf{p}^{(i)},\boldsymbol{\pi}_m^{(i)})+p_m^{(i)}} - \frac{q_m^2(p_m^{(i)})}{(q_m(p_m^{(i)})+p_m^{(i)})^2}\big(q_m(\mathbf{p},\boldsymbol{\pi}_m^{(i)})+p_m\big)\Big) - \frac{q_m^2(\mathbf{p},\boldsymbol{\pi}_m^{(i)})}{(q_m^{(i)}(\mathbf{p},\boldsymbol{\pi}_m^{(i)})+p_m^{(i)})^2} \tag{C.4}$$

over the trusted regions defined in (19) and (20). By substituting (C.4) to (C.2) yields

$$V_m(\mathbf{p},\boldsymbol{\pi}_m^{(i)}) \leq \mathcal{V}_m^{(i)}(\mathbf{p},\boldsymbol{\pi}_m^{(i)}) \triangleq d_m^{(i)} - \frac{2e_m^{(i)}}{q_m(\mathbf{p}^{(i)},\boldsymbol{\pi}_m^{(i)})+p_m^{(i)}}\Big(2f_m^{(i)}q_m(\mathbf{p},\boldsymbol{\pi}_m^{(i)}) - (f_m^{(i)})^2\big(q_m(\mathbf{p},\boldsymbol{\pi}_m^{(i)})+p_m\big)\Big) + \frac{(f_m^{(i)})^2}{q_m^2(\mathbf{p}^{(i)},\boldsymbol{\pi}_m^{(i)})}q_m^2(\mathbf{p},\boldsymbol{\pi}_m^{(i)}) \tag{C.5}$$

where $f_m^{(i)} \triangleq \dfrac{q_m(\mathbf{p}^{(i)},\boldsymbol{\pi}_m^{(i)})}{q_m(\mathbf{p}^{(i)},\boldsymbol{\pi}_m^{(i)})+p_m^{(i)}}$.

REFERENCES

[1] 3GPP, "Study on scenarios and requirements for next generation access technologies," 3GPP, Technical Report (TR) 38.913, 2018, version 15.0.0.

[2] T. T. Vu, D. N. Nguyen, D. T. Hoang, E. Dutkiewicz, and T. V. Nguyen, "Optimal energy efficiency with delay constraints for multi-layer cooperative fog computing networks," *IEEE Trans. Commun.*, vol. 69, no. 6, pp. 3911–3929, Jun. 2021.

[3] J. Wang, D. Feng, S. Zhang, A. Liu, and X.-G. Xia, "Joint computation offloading and resource allocation for MEC-enabled IoT systems with imperfect CSI," *IEEE Internet Things J.*, vol. 8, no. 5, pp. 3462–3475, Mar. 2021.

[4] C.-F. Liu, M. Bennis, M. Debbah, and H. V. Poor, "Dynamic task offloading and resource allocation for ultra-reliable low-latency edge computing," *IEEE Trans. Commun.*, vol. 67, no. 6, pp. 4132–4150, Jun. 2019.

[5] Z. Xiong *et al.*, "Cloud/edge computing service management in blockchain networks: Multi-leader multi-follower game-based ADMM for pricing," *IEEE Trans. Services Comput.*, vol. 13, no. 2, pp. 356–367, Mar./Apr. 2020.

[6] Q. Ye, W. Shi, K. Qu, H. H. W. Zhuang, and X. Shen, "Joint RAN slicing and computation offloading for autonomous vehicular networks: A learning-assisted hierarchical approach," *IEEE Open J. Veh. Technol.*, vol. 2, pp. 272–288, Jun. 2021.

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

[8] Y. Xiao and M. Krunz, "Distributed optimization for energy-efficient fog computing in the tactile Internet," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 11, pp. 2390–2400, Nov. 2018.

[9] P. Zhao, H. Tian, C. Qin, and G. Nie, "Energy-saving offloading by jointly allocating radio and computational resources for mobile edge computing," *IEEE Access*, vol. 5, pp. 11 255–11 268, 2017.

[10] F. Wang, J. Xu, and S. Cui, "Optimal energy allocation and task offloading policy for wireless powered mobile edge computing systems," *IEEE Trans. Wireless Commun.*, vol. 19, no. 4, pp. 2443–2459, 2020.

[11] F. Wang, J. Xu, and Z. Ding, "Multi-antenna noma for computation offloading in multiuser mobile edge computing systems," *IEEE Trans. Commun.*, vol. 67, no. 3, pp. 2450–2463, 2019.

[12] R. Lin, Z. Zhou, S. Luo, Y. Xiao, X. Wang, S. Wang, and M. Zukerman, "Distributed optimization for computation offloading in edge computing," *IEEE Trans. Wireless Commun.*, vol. 19, no. 12, pp. 8179–8194, Dec. 2020.

[13] D. T. Nguyen, L. B. Le, and V. Bhargava, "Price-based resource allocation for edge computing: A market equilibrium approach," *IEEE Trans. Cloud Comput.*, vol. 9, no. 1, pp. 302–317, Jan. 2021.

[14] J. Zhou, D. Tian, Z. Sheng, X. Duan, and X. S. Shen, "Distributed task offloading optimization with queueing dynamics in multi-agent mobile-edge computing networks," *IEEE Internet Things J.*, vol. 4662, no. c, pp. 1–19, 2021.

[15] Y. Polyanskiy, H. V. Poor, and S. Verdú, "Channel coding rate in the finite blocklength regime," *IEEE Trans. Inf. Theory*, vol. 56, no. 5, pp. 2307–2359, 2010.

[16] C. She, C. Yang, and T. Q. S. Quek, "Cross-layer optimization for ultra-reliable and low-latency radio access networks," *IEEE Trans. Wireless Commun.*, vol. 17, no. 1, pp. 127–141, 2018.

[17] C. Sun *et al.*, "Optimizing resource allocation in the short blocklength regime for ultra-reliable and low-latency communications," *IEEE Trans. Wireless Commun.*, vol. 18, no. 1, pp. 402–415, 2019.

[18] Z. Zhou, Z. Wang, H. Yu, H. Liao, S. Mumtaz, L. Oliveira, and V. Frascolla, "Learning-based URLLC-aware task offloading for Internet of health things," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 2, pp. 396–410, Feb. 2021.

[19] Y. Wang, X. Tao, X. Zhang, P. Zhang, and Y. T. Hou, "Cooperative task offloading in three-tier mobile computing networks: An admm framework," *IEEE Trans. Veh. Technol.*, vol. 68, no. 3, pp. 2763–2776, 2019.

[20] J. Wang, K. Liu, B. Li, T. Liu, R. Li, and Z. Han, "Delay-sensitive multi-period computation offloading with reliability guarantees in fog networks," *IEEE Trans. Mobile Comput.*, vol. 19, no. 9, pp. 2062–2075, 2020.

[21] H. Xing, L. Liu, J. Xu, and A. Nallanathan, "Joint task assignment and resource allocation for D2D-enabled mobile-edge computing," *IEEE Trans. Commun.*, vol. 67, no. 6, pp. 4193–4207, 2019.

[22] Y. Huang, Y. Liu, and F. Chen, "NOMA-aided mobile edge computing via user cooperation," *IEEE Trans. Commun.*, vol. 68, no. 4, pp. 2221–2235, 2020.

[23] T. Q. Dinh, B. Liang, T. Q. S. Quek, and H. Shin, "Online resource procurement and allocation in a hybrid edge-cloud computing system," *IEEE Trans. Wireless Commun.*, vol. 19, no. 3, pp. 2137–2149, Mar. 2020.

[24] B. R. Marks and G. P. Wright, "A general inner approximation algorithm for nonconvex mathematical programs," *Operations Research*, vol. 26, no. 4, pp. 681–683, July-Aug. 1978.

[25] H. Ren, C. Pan, Y. Deng, M. Elkashlan, and A. Nallanathan, "Joint pilot and payload power allocation for massive-MIMO-enabled URLLC IIoT networks," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 5, pp. 816–830, 2020.

[26] L. Fang and L. Milstein, "Performance of successive interference cancellation in convolutionally coded multicarrier DS/CDMA systems," *IEEE Trans. Commun.s*, vol. 49, no. 12, pp. 2062–2067, 2001.

[27] M. Merluzzi *et al.*, "Dynamic computation offloading in multi-access edge computing via ultra-reliable and low-latency communications," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 6, pp. 342–356, 2020.

[28] Q. Liu, T. Han, and N. Ansari, "Joint radio and computation resource management for low latency mobile edge computing," in *IEEE Global Communications Conference, GLOBECOM 2018*, Abu Dhabi, United Arab Emirates, 2018.

[29] J. Tang, W. P. Tay, T. Q. S. Quek, and B. Liang, "System cost minimization in cloud ran with limited fronthaul capacity," *IEEE Trans. Wireless Commun.*, vol. 16, no. 5, pp. 3371–3384, May 2017.

[30] T. Q. Dinh, J. Tang, Q. D. La, and T. Q. S. Quek, "Offloading in mobile edge computing: Task allocation and computational frequency scaling," *IEEE Trans. Commun.*, vol. 65, no. 8, pp. 3571–3584, 2017.

[31] A. Beck, A. Ben-Tal, and L. Tetruashvili, "A sequential parametric convex approximation method with applications to nonconvex truss topology design problems," *J. Global Optim.*, vol. 47, no. 1, pp. 29–51, May 2010.

[32] A. A. Nasir, H. D. Tuan, H. Nguyen, M. Debbah, and H. V. Poor, "Resource allocation and beamforming design in the short blocklength regime for URLLC," *IEEE Trans. Wireless Commun.*, vol. 20, no. 2, pp. 1321–1335, 2021.

[33] A. Ben-Tal and A. Nemirovski, *Lectures on Modern Convex Optimization.* Philadelphia: MPS-SIAM Series on Optimi., SIAM, 2001.

[34] H. V. Nguyen, V.-D. Nguyen, O. A. Dobre, D. N. Nguyen, E. Dutkiewicz, and O.-S. Shin, "Joint power control and user association for NOMA-based full-duplex systems," *IEEE Trans. Commun.*, vol. 67, no. 11, pp. 8037–8055, Nov. 2019.

[35] E. Che, H. D. Tuan, and H. H. Nguyen, "Joint optimization of cooperative beamforming and relay assignment in multi-user wireless relay networks," *IEEE Trans. Wireless Commun.*, vol. 13, no. 10, pp. 5481–5495, 2014.

[36] W. Hao, O. Muta, and H. Gacanin, "Price-based resource allocation in massive mimo h-crans with limited fronthaul capacity," *IEEE Trans. Wireless Commun.*, vol. 17, no. 11, pp. 7691–7703, 2018.

[37] C. She, C. Yang, and T. Q. S. Quek, "Radio resource management for ultra-reliable and low-latency communications," *IEEE Commun. Mag.*, vol. 55, no. 6, pp. 72–78, 2017.

[38] T. X. Tran and D. Pompili, "Joint task offloading and resource allocation for multi-server mobile-edge computing networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 1, pp. 856–868, 2018.

[39] V.-D. Nguyen, T. Q. Duong, H. D. Tuan, O.-S. Shin, and H. V. Poor, "Spectral and energy efficiencies in full-duplex wireless information and power transfer," *IEEE Trans. Commun.*, vol. 65, no. 5, pp. 2220–2233, May 2017.

[40] V.-D. Nguyen, H. V. Nguyen, O. A. Dobre, and O.-S. Shin, "A new design paradigm for secure full-duplex multiuser systems," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 7, pp. 1480–1498, July 2018.