Distributed Task Offloading Optimization With Queueing Dynamics in Multiagent Mobile-Edge Computing Networks

Jianshan Zhou, Daxin Tian, *Senior Member, IEEE*, Zhengguo Sheng, *Senior Member, IEEE*, Xuting Duan, and Xuemin Shen, *Fellow, IEEE*

***Abstract*—Task offloading decision making plays a key role in enabling mobile-edge computing (MEC) technologies in Internet of Things (IoT). However, it meets the significant challenges arising from the stochastic dynamics of task queueing in the application layer and coupled wireless interference in the physical layer in a distributed multiagent network without any centralized communication and computing coordination. In this article, we investigate the distributed task offloading optimization problem with consideration of the upper layer queueing dynamics and the lower-layer coupled wireless interference. We first propose a new optimization model that aims at maximizing the expected offloading rate of multiple agents by optimizing their offloading thresholds. Then, we transform the problem into a game-theoretic formulation, which further leads to the design of a distributed best-response (DBR) iterative optimization framework. The existence of Nash equilibrium strategies in the game-theoretic model has been analyzed. For the individual optimization of each agent's threshold policy, we further propose a programming scheme by transforming a constrained threshold optimization into an unconstrained Lagrangian optimization (ULO). The individual ULO is integrated into the DBR framework to enable agents to cooperate and converge to a global optimum in a distributed manner. Finally, simulation results are provided to validate the proposed method and demonstrate its significant advantage over other existing distributed methods. The numerical results also show that the proposed method can achieve comparable performance to a centralized optimization method.**

***Index Terms*—Distributed optimization, mobile-edge computing (MEC), multiagent networks, queueing dynamics, task offloading, wireless interference.**

Manuscript received September 11, 2020; revised January 16, 2021; accepted February 28, 2021. Date of publication March 3, 2021; date of current version July 23, 2021. This work was supported in part by the "Zhuoyue" Program of Beihang University (Postdoctoral Fellowship); in part by the China Postdoctoral Science Foundation under Grant 2020M680299; in part by the National Natural Science Foundation of China under Grant 61822101 and Grant U20A20155; in part by the Beijing Municipal Natural Science Foundation under Grant L191001 and Grant 4181002; in part by the Newton Advanced Fellowship under Grant 62061130221; in part by H2020-MSCA-RISE under Grant 101006411-SEEDS; and in part by the Royal Society Kan Tong Po International Fellowship under Grant KTP-R1-201007. *(Corresponding author: Daxin Tian.)*

Jianshan Zhou, Daxin Tian, and Xuting Duan are with the Beijing Advanced Innovation Center for Big Data and Brain Computing, Beijing Key Laboratory for Cooperative Vehicle Infrastructure Systems and Safety Control, School of Transportation Science and Engineering, Beihang University, Beijing 100191, China (e-mail: jianshanzhou@foxmail.com; dtian@buaa.edu.cn; duanxuting@buaa.edu.cn).

Zhengguo Sheng is with the Department of Engineering and Design, University of Sussex, Richmond 3A09, U.K. (e-mail: z.sheng@sussex.ac.uk).

Xuemin Shen is with the Electrical and Computer Engineering Department, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: sshen@uwaterloo.ca).

Digital Object Identifier 10.1109/JIOT.2021.3063509

I. Introduction

TREMENDOUS increases in the deployment of Internet-of-Things (IoT) devices and a wide range of emerging applications and services, such as connected autonomous driving, high-fidelity live multimedia, real-time social virtual reality and industry automation, etc., have boosted the need of higher data rates in wireless communication networks and more computing and storage capacities. This trend leads to a rapid development of many envisioned information communication and networking architectures, such as the new radio access networks in 5G [1], the software-defined vehicular networks [2], [3] and the space-air-ground integrated networks [4], etc., and has also spawned a new computing paradigm, termed mobile-edge computing (MEC) [5], [6]. Specifically, a MEC network enables mobile end-users and IoT devices with constrained computation and storage resources to offload their computation-intensive tasks to the close-proximity network edge which can provide advanced computing power to serve the resources-hungry users. In comparison to the traditional cloud computing, MEC deploys cloud computing resources at the edge of the network, which is in close proximity to the end users, and thus avoids a high service latency caused by a long data transmission distance between the end users and the remote centralized cloud servers. Therefore, MEC is promising for many existing and envisioned mobile applications that usually require low-latency and high-reliability communication and massive computing capacities. Currently, MEC has attracted much attention from both academia and industry communities [7]. To practically realize MEC for various IoT applications and services, task/computation offloading is one of the most important enabling technologies and has been extensively investigated from different system perspectives including 5G-enabled multiple access [8]–[10], unmanned aerial vehicles (UAVs)-aided communications [11], [12], space-air-ground integration [13], [14] and connected vehicles [15]–[17], etc. In particular, many efforts have been already dedicated to addressing the issue of joint optimization of communication

2327-4662 © 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: University of Waterloo. Downloaded on November 03,2023 at 14:29:46 UTC from IEEE Xplore. Restrictions apply.

and computing in MEC [5], [18], the goals of which range from the energy-efficiency maximization to the system-wide reliability optimization [9], [11], [15], [19].

In general, computation tasks generated by the application layer of end users can form a data queue, which involves the dynamic and stochastic processes of traffic arrivals, packet service and waiting in each user's buffer. The queueing dynamics of computation tasks, which is an exogenous factor in the network, can significantly affect the users' offloading decisions and the overall MEC performance. However, while most task offloading paradigms focus on the modeling, control and optimization with consideration of the dynamics of communication or/and computing in a specific MEC scenario, existing offloading approaches usually simplify or even ignore the impact of the queueing dynamics due to the increased complexity and stochastic nature of the queueing system, which will introduce additional challenges to the formulation and analysis of MEC. Clearly, in order to make offloading decisions effectively in MEC, the end users, who can be named as decision-making agents, must not only capture the dynamics of their communication and computing environment, but also the queueing dynamics of their own upper layer tasks. It remains to be an open question of how to bring the networked queueing dynamics into the offloading process and thus to design more practical decision-making strategies and achieve cross-layer optimization.

On the other hand, while a large number of task offloading frameworks have been developed for optimizing MEC performance with different objectives, a common limitation is the reliance on the deployment and availability of a system-level control or centralized management infrastructure for achieving the global optimization or coordinating resource allocation, as shown in the existing work [3], [9], [10], [13], [14], [20]. On the contrary, distributed optimization paradigms are more appealing and promising to deal with the increased complexity in large-scale networked systems, especially in *ad hoc* multiagent settings, such as infrastructureless wireless sensor networks (WSNs), vehicular *ad hoc* networks (VANETs), flying *ad hoc* networks, etc. To this end, several distributed approaches for task offloading in MEC have been proposed, such as [8], [17], and [21]–[23], among which the game-theoretical tools are widely exploited to model the competitive interactions of distributed agents contending for the limited communication resources and Nash equilibrium-specified strategies are usually adopted as the system solutions. The distributed decision-making paradigms based on game theory can explicitly characterize the interaction dynamics of the competitive individuals' decision-making behaviors in the source-constrained system. In spite of this progress, the stochastic dynamics of potential power interference incurred by multiple individuals simultaneously competing to access the same available spectrum has not been fully captured from the communication perspective and incorporated in the game-theoretical models. Indeed, in most previous work, such as [3], [8]–[10], [13], [14], [17], and [20]–[23], a static channel model based on a constant signal-to-noise ratio (SNR) or signal-to-interference-plus-noise ratio (SINR) is widely adopted to describe the wireless transmission dynamics. However, the communication interference is inherently related to multiple agents' offloading strategies and can also be random due to the fading in the physical-layer channel. This fact results in the strong coupling of the physical-layer communication dynamics and the upper layer decision-making behaviors, which makes the distributed optimization design and analysis of multiagent task offloading complicated and challenging.

Another challenge may arise when the coupled interactions of multiple agents are considered in the system model. That is, unfortunately, the optimization objective functions of most actual connected systems in the field of MEC are neither convex nor concave [11], [17]. Even though some researchers have investigated and successfully solved task offloading problems for some specific objectives by formulating them as convex optimization problems, such as the work [21], it is theoretically difficult or even impossible to search the global optimum of a nonconvex or a nonconcave objective function in a high-dimension problem domain. In the situation, it is important to design an efficient search mechanism with low complexity to obtain a suboptimal solution.

A. Motivation and Contributions

In a distributed multiagent edge computing scenario where multiple agents compete to access the limited common spectrum when the centralized coordination is not available, physical-layer concurrent interference among the agents should be properly considered. Each agent needs to determine whether to buffer each arrival task from the application layer or to offload it to the network edge for the remote computing immediately. Thus, the offloading decision of each agent can influence all the others: on the one side, when an agent chooses to offload the task, it can create wireless communication interference to others. Multiple agents' concurrent interferences are coupled and can potentially increase the physical-layer transmission error rate of the agents and the signal frame scheduling delay. On the other side, when the agent decides to enqueue the task, it will increase the occupancy of its buffer and the queueing delay in the buffer, which potentially leads to a high packet dropping probability because of exceeding a maximum allowed sojourn time (i.e., a delay deadline) and the buffer capacity. The resulting coupling in the agents' decision-making behaviors, the physical-layer interference, and the application-layer queueing dynamics can make the modeling and distributed optimization design of multiagent computation offloading a complicated and challenging issue, which, however, is remained to be explored at full length. Therefore, in this article, we aim at addressing the distributed offloading decision-making optimization of multiple agents competing to access limited spectral resources. Specifically, we jointly bring the queueing dynamics of the agents' tasks and the physical-layer coupled communication interference into the multiagent decision-making process, which leads to a cross-layer optimization design. More importantly, we model the effects of the buffer capacity and the

Authorized licensed use limited to: University of Waterloo. Downloaded on November 03,2023 at 14:29:46 UTC from IEEE Xplore. Restrictions apply.

sojourn time limit[1] of the agents' tasks in the queueing system and the effects of the physical-layer coupled interference from the probabilistic perspective, and then incorporate such effects into the offloading decision-making formulation.

To be specific, we propose to optimize the offloading decision thresholds of agents in a distributed manner by maximizing the expected successful offloading rate of agents' tasks. For this goal, we combine the game-theoretical analysis with constrained nonlinear optimization theory. From the perspective of game theory, we show the existence of at least a mixed-strategy Nash equilibrium rather than a pure-strategy Nash equilibrium in our system model, which does not require the convexity or concavity of the problem. This is different from most of existing game-theoretical approaches (as aforementioned) that need to well define a distributed convex or concave problem so as to guarantee the existence of a pure-strategy Nash equilibrium. This further motivates us to develop a distributed multiagent iterative framework based on the best response mechanism. To deal with the individual optimization at each iteration, we further propose an unconstrained Lagrangian optimization (ULO) algorithm by transforming the individual optimization into an augmented Lagrangian subproblem, which has the advantage in unconstrained programming. The best response method is combined with the proposed ULO scheme to enable collaborative optimization of the entire system performance via the iterative optimization of individual utilities in a distributed fashion.

The main contributions of this article are summarized as follows.

1) From the communication perspective, we first formulate the system optimization of task offloading in a MEC network as a multiagent decision-making problem, wherein each agent aims at determining an optimal offloading threshold to maximize its expected successful offloading rate. That is, when the received signal envelope at an agent's destination (i.e., an edge node) over a selected frequency channel exceeds the optimal offloading threshold, which indicates that the condition of the physical-layer channel is good enough, the agent decides to offload its task to the edge node for remote execution, otherwise it would buffer the task in a queue or forward it to its central processing unit (CPU) for local execution. We jointly consider the effects of the limited buffer capacity and the patience of the task with waiting for offloading, and thus model the task loss probability due to buffer overflow and queueing delay. We also capture the queueing dynamics, which is incorporated into the offloading decision-making behaviors of the agents.
2) We formulate a probabilistic model to approximately capture the joint distribution of the stochastic coupling transmission interference of the multiple agents contending in the same channel, which is connected to their offloading decisions. This model well captures the complex interaction dynamics of these agents, showing that the offloading decision of each agent relies on other peer agents via the proposed interference function.
3) We propose to transform the constrained individual offloading optimization into an unconstrained optimization with an augmented Lagrangian function. Based on this, we develop a distributed best response method in combination with a proposed ULO scheme, in which each agent only needs to optimize its own utility with its own offloading decision. This distributed ULO-based best response method can induce the agents to collaborate with each other via decision feedback and iteratively maximize the global system performance.
4) We conduct extensive numerical experiments to evaluate the effectiveness and advantages of our proposed distributed method. Specifically, we compare our proposed method with state-of-the-art distributed algorithms (i.e., a distributed dual-decomposition algorithm and a distributed stochastic learning-based algorithm), the distributed aggressive policy-based algorithm and the centralized optimization algorithm. The results show that our distributed method can well approximate the global optimum and also outperform other comparative methods.

[1]In queueing theory [24], a sojourn time limit is defined as a time threshold, which can be used to characterize the patience of an entity waiting for service and thus is related to the Quality-of-Service (QoS) requirement. In our considered context here, a task is admitted to the queueing system if the buffer queue is not full (i.e., the buffer is not fully occupied), otherwise it is dropped by the buffer and transferred to the processing unit for the local execution. When the task joins the buffer queue, it is served immediately (i.e., offloaded to the network edge for the remote execution) or needs to wait for its service. In this situation, if the sojourn time (waiting time plus service time) of this task exceeds its tolerant time threshold, this task will be dropped for the location execution rather than being offloaded for the remote execution.

B. Mathematical Notation

In this article, boldface uppercase letters like $\mathbf{X}$ and boldface lowercase letters $\mathbf{x}$ are used to represent matrices and vectors, respectively. All vectors are understood as column vectors when without additional specific statements. Sets are denoted by calligraphic letters like $\mathcal{K}$. The sets of nonnegative real numbers and of nonnegative integers are denoted by $\mathbb{R}_+$ and $\mathbb{Z}_+$, respectively. Besides, $\mathbb{R}_{++}$ and $\mathbb{Z}_{++}$ denote the sets of positive real numbers and positive integers, respectively. The kth row and lth column element of a matrix $\mathbf{X}$ is denoted by $[\mathbf{X}]_{k,l}$, while the kth component of a vector $\mathbf{x}$ is denoted by $[\mathbf{x}]_k$. The transpose of a matrix $\mathbf{X}$ and a vector $\mathbf{x}$ is denoted by $\mathbf{X}^{\mathrm{T}}$ and $\mathbf{x}^{\mathrm{T}}$, respectively. $\mathrm{diag}\{x_1, x_2, \ldots, x_n\}$ denotes a $n \times n$ diagonal matrix whose diagonal components are $x_1, x_2, \ldots, x_n$. $\mathrm{col}\{x_1, x_2, \ldots, x_n\}$ represents a column vector with n components. Vectorized functions are denoted by boldface symbols, such as $\mathbf{f}(x)$ and $\mathbf{F}(x)$. Throughout this article, the inner product of two (column) vectors, $\mathbf{x}$ and $\mathbf{y}$, is given as $\mathbf{x}^{\mathrm{T}}\mathbf{y}$. The Euclidean norm of a vector $\mathbf{x}$ is represented by $\|\mathbf{x}\|_2$ or $\|\mathbf{x}\|$ for the sake of simplicity, i.e., letting $\|\mathbf{x}\| = \sqrt{\mathbf{x}^{\mathrm{T}}\mathbf{x}}$.

C. Paper Organization

The remainder of this article is organized as follows. In Section II, the related work is reviewed. In Section III, the system model and the primal optimization formulation are described. The distributed optimization method for the offloading decision making of the multiple agents is proposed in

Authorized licensed use limited to: University of Waterloo. Downloaded on November 03,2023 at 14:29:46 UTC from IEEE Xplore. Restrictions apply.

details in Section IV. Section V comparatively evaluates the performance of the proposed methods. We conclude this work in Section VI.

II. Related Work

MEC has received much attention from both academia and industry due to many emerging IoT applications and services [5], [6], especially in the 5G era [7], [18]. Among various topics in the field of MEC, task/computation offloading plays a key role in enabling the system deployment for different application scenarios. There exist a significant number of studies focusing on the offloading decision making for a single user, such as [11], [12], [15], [16], and [25]. To be specific, Zhang *et al.* [25] have proposed an enumeration search algorithm for a single user to solve an optimal task offloading solution as well as a Lagrangian relaxation-based aggregated cost (LARAC) algorithm to solve a suboptimal solution with lower complexity under stochastic channels. In [15], the coupling reliability of vehicular communication and computing is explored and the authors propose an optimal task offloading method for a vehicular user to determine a reliability-optimal computing mode. Zhou *et al.* [16] also focused on a vehicular user, in which the offloading decision-making problem was formulated as a stochastic optimization problem and a stochastic dynamic programming method has been developed. Besides, many researchers also introduce MEC into other envisioned connected systems, such as aerial-ground integrated networks where vehicular networks and UAVs-aided flying *ad hoc* networks are connected to provide a large communication coverage and a high system capacity [26], [27]. Jeong *et al.* [11] jointly addressed the task offloading problem and the path planning problem of a single UAV by exploiting successive convex approximation techniques. Similarly, Zhou *et al.* [12] have derived closed-form expressions to obtain optimal CPU frequencies, offloading time and transmission power. It can be seen that many successful models and solutions have been designed and well validated in existing literature as mentioned above for scenario-specified task offloading problems. However, these solutions, in essence, focus on the decision-making behavior of a single entity (e.g., a mobile device [25], a vehicle [15], [16] or a UAV [11], [12]) and do not target more complex cases in self-organized multiuser settings, where interacting users' decisions are coupled with and affect each other's.

With the advancements of software-defined networks (SDNs) and multiaccess technologies, there are many research efforts that have been made to deal with MEC-related problems in SDN-based or/and multiaccess scenarios, such as [3], [9], [10], and [20]. Specifically, in [3], a centralized resource management framework is designed based on SDNs, which enables a tier-1 cloud-computing server and several tier-2 MEC servers to collaboratively process the application task of a connected autonomous vehicle (CAV). Alameddine *et al.* [10] jointly considered task offloading and scheduling problem, which combines three NP-hardness (nondeterministic polynomial-time hardness) subproblems, and proposes a Logic-Based Benders Decomposition method. Chen and Hao [9] also formulated the task offloading problem in a SDN-based ultradense network as a mixed-integer nonlinear program, which is NP-hardness. To obtain a suboptimal solution, the authors propose a decomposition method to transform this original problem into two subproblems [9]. Different from the efforts aforementioned, the work [20] studies the resource management with the goal of enhancing the secrecy against eavesdropping attacks in nonorthogonal multiaccess assisted computation offloading scenarios. In the above literature, diverse system models are usually established from a holistic design perspective and thus require centralized infrastructure, such as an SDN controller.

In addition, many researchers are currently engaged in applying artificial neural networks-based supervised learning and reinforcement learning theories to address some complex task offloading issues, such as the studies in [14], [28], and [29]. For example, Cheng *et al.* [14] presented the computation offloading problem in a space-air-ground integrated network as a Markov decision process and then propose a deep reinforcement learning approach for a UAV-user to learn an optimal policy for gaining a minimum long-term comprehensive cost (i.e., including delay, energy and server usage costs). To minimize the long-term delay of a task, [29] designs two reinforcement learning methods, including a Q-learning method and a deep reinforcement learning method, to obtain the optimal policies for computation offloading and resource allocation in a vehicle edge-computing network. Heydari *et al.* [28] also applied a deep reinforcement learning approach to make task offloading decisions with the goal of minimizing the task drop rate and the execution delay. From the methodological perspective, the reinforcement learning-based solutions can provide the advantages over traditional optimization techniques (e.g., convex optimization) in self-organization, evolvability and adaptability. However, these approaches rely on the fundamental assumption on stationary and Markovian environments, while the observation of any single decision maker in a multiuser context may be dynamic and nonstationary due to the interactions among different users' decision-making behaviors.

Due to the constraint on the communication and computing resources, selfish users' decision-making behaviors may be competitive. As thus, game-theoretical approaches are widely adopted in current literature to model and analyze the decision-making processes in task offloading. Zhao *et al.* [17] decomposed an original problem of joint task offloading and resource allocation into two subproblems, among which a potential game is formulated for obtaining distributed Nash-equilibrium offloading strategies. Cao and Cai [22] also study the multiuser offloading decision-making problem and develop an exact potential game model, for which they propose a stochastic learning-based distributed algorithm to get a pure Nash equilibrium solution. Based on the theory of potential game, Zheng *et al.* [23] have presented a weighted potential game to capture the dynamics of multiuser computation offloading environment, in which users can be active or inactive and the channels can vary randomly. Besides, a multiagent stochastic learning algorithm has been proposed to learn a Nash equilibrium solution for these users [23]. Different from these potential game models above, the work [8] presents a

Authorized licensed use limited to: University of Waterloo. Downloaded on November 03,2023 at 14:29:46 UTC from IEEE Xplore. Restrictions apply.

player-specific congestion game formulation for multiuser task offloading and provides a decentralized algorithm to converge to Nash equilibrium strategies. It can be summarized from the literature that game theory is really a powerful mathematical tool for designing distributed decision-making paradigms and dealing with the complexity in the interactions of multiple agents. Nevertheless, the implementation of a game-theoretical approach usually requires a well-defined individual utility function (e.g., the formulation satisfying the potential game property in [17], [22], and [23]), such that the game model can theoretically guarantee the existence of at least a Nash equilibrium point. Indeed, it is still an open and challenging issue to prove the existence of a pure Nash equilibrium point and obtain a Nash-equilibrium solution for a general game formulation.

Another direction in current existing literature is dedicated to applying convex optimization techniques, such as the widely used dual decomposition [21] and Lyapunov optimization [19], [30]–[33], to directly solve the optimization of multiuser task offloading. In [21], the Lagrangian dual-decomposition method has been used to address the energy-efficiency optimization problem of computation offloading and scheduling, which can lead to a distributed algorithm. Nevertheless, although the proposed algorithm in [21] aims at reducing the application completion time, it does not take into account the potential delay resulting from queueing the application task in the local buffer. It can be recognized that some other works like [19], [30]–[33] have considered the queueing dynamics of tasks in each mobile device and incorporated the queue delay, which is assumed to be proportional to the average queue length, into their system models. Their solutions are developed by invoking the Lyapunov optimization for different long-term optimization objectives, such as the minimization of the execution delay and the task dropping cost [19], the energy consumption minimization [30], the latency and reliability-constrained transmit power minimization [32], and the energy-delay tradeoff [31], [33]. In addition, some combinatorial optimization and heuristic designs have also been developed in the current literature. For instance, Gai *et al.* [34] have proposed an optimization algorithm based on dynamic programming, which takes into account the energy cost for selecting cloudlet and its computing service performance. In [35], a task assignment model and a scheduling table generation algorithm have been proposed based on the theory of directed acyclic graphs with the goal of improving the overall energy efficiency in heterogeneous cloud computing. To jointly optimize the energy cost and time consumption in heterogeneous fog computing scenarios, [36] has proposed a heuristic algorithm by combining graph theory with a hill-climbing search scheme. However, how the interaction of end users' offloading decisions and the physical-layer coupled interference can influence the long-term optimization objective remains unexplored in these existing works.

In this article, our approach also takes into consideration the queueing dynamics of the computation tasks and further captures the impacts of the task buffer capacity and the tolerant sojourn time in the queueing network, which is quite different from the queueing models in the existing literature mentioned above. Moreover, our optimization formulation explicitly characterizes the coupling dynamics of distributed users' interactions and connects the upper layer offloading decisions with the physical-layer communication dynamics, which have not been fully explored in the aforementioned works based on convex optimization, game theory or modern reinforcement learning. To the best of our knowledge, this work is the first to establish a distributed optimization paradigm in multiagent networks by coupling of the queueing dynamics-aware offloading decisions.

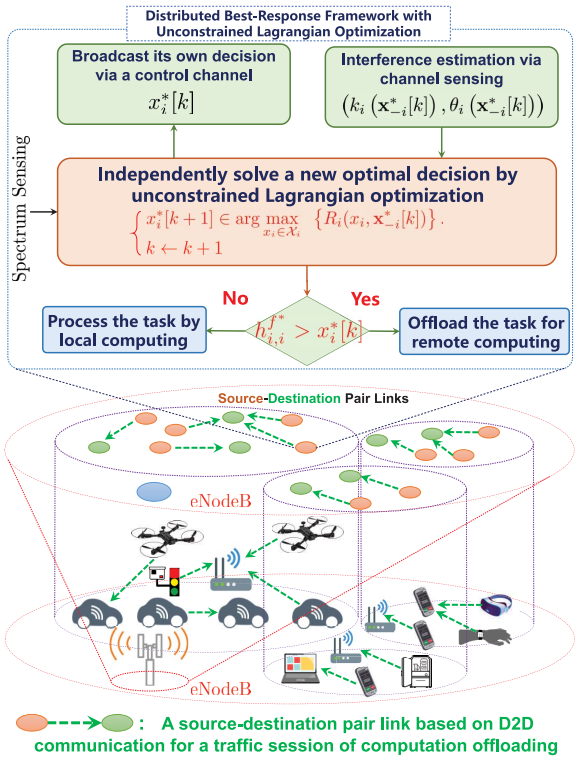

Fig. 1. Exemplary application scenario, where a D2D-based IoT system is considered for the sake of demonstration on the usage of our proposed methodological framework. In this scenario, diverse IoT devices are equipped with D2D communications. Each source agent can offload its computation to a destination in close proximity to itself via a D2D link or just perform local computing. Each offloading session is abstracted by a source-destination pair and each agent's destination is reachable via one hop. The multiple offloading sessions share the common spectrum provided by an eNodeB node as a licensed provider. In our proposed model, the common spectrum available for computation offloading is divided into a set of frequency channels, whereas the time horizon is discretized into a series of time slots. In each time slot, each agent can decide to either offload its computation to its destination by selecting a frequency channel from the available spectrum set when the channel quality is higher than an optimal threshold, or keep silent and enqueue the arrival task in its local buffer. It is remarked that such a paradigm is sufficiently general to support a variety of wireless communication and networking systems (besides the scenario shown here).

III. System Model

We consider such a general MEC scenario as illustrated in Fig. 1, where there are multiple traffic sessions with each denoting a task offloading connection between two agents, i.e., a source and its corresponding destination, and these offloading sessions share the same spectrum. We denote the sets of these offloading sessions and their shared frequency channels

Authorized licensed use limited to: University of Waterloo. Downloaded on November 03,2023 at 14:29:46 UTC from IEEE Xplore. Restrictions apply.

by $\mathcal{I} = \{1, 2, \ldots, N\}$ and $\mathcal{F} = \{1, 2, \ldots, M\}$, respectively. The task offloading is divided into a series of discrete time slots, and the duration of each time slot is denoted by $\Delta\tau$ seconds. We consider that the source agent of each offloading session can either offload an application task from its upper layer to a network edge for remote computation in each time slot with a selected physical-layer frequency, $f \in \mathcal{F}$, or keep silent in this time slot and push this arrival task into a first-in-first-out (FIFO) queue buffer. It can be seen that such a proposed model is general and thus can be adapted to many types of wireless networks, such as wireless *ad hoc* networks, cellular systems with multiuser wireless interferences, device-to-device (D2D) communication networks in the envisioned 5G systems, etc.

Besides, we design a threshold-based offloading decision-making policy for each agent. More specifically, for any source agent of the offloading session $i \in \mathcal{I}$, its offloading threshold is represented as a decision variable $x_i \in \mathbb{R}_+$. When the agent's selected channel, $f \in \mathcal{F}$, has a good communication condition in which the channel gain (i.e., the received signal envelope) is higher than the threshold x_i, it decides to offload an arrival application task for the remote execution; Otherwise, it will enqueue this task in its limited buffer if the channel quality is bad. The agent can adapt (optimally configure) its offloading decision x_i in order to maximize its expected successfully offloading rate. In the considered scenario, a larger value of an agent's offloading threshold x_i can reduce the opportunity of task offloading and thus lose the benefit gained from the edge computing, while, on the contrary, a smaller value of x_i can lead to a higher probability of task offloading, which increases the intensity of physical-layer channel contention and wireless interferences in the multiagent environment. The packet loss rate can be increased with the communication concurrence when the spectrum resource is limited, which further results in a worse offloading performance. At this point, it is a key and fundamental issue to optimize the joint offloading policies of all the agents, $\mathbf{x} \triangleq (x_i)_{i\in\mathcal{I}}$, to realize an optimal tradeoff between task offloading and task queueing.

A. Dynamics of Wireless Channels and Task Offloading

For any source agent $i \in \mathcal{I}$, we also let its destination be represented by i for the sake of simplicity. The transmit power of this agent is denoted by p_i and the nonsingular path loss coefficient is $a_{i,i}$, which is related to the relative distance between the agent i and its corresponding destination, $d_{i,i}$, i.e., $a_{i,i} = (1 + d_{i,i}^{\alpha})^{-1/2}$ where α denotes the path loss factor. We also denote the received signal envelope at a selected channel frequency $f \in \mathcal{F}$ by $h_{i,i}^f$. Hence, the transmission channel gain for the offloading session associated with the agent i can be expressed as $g_{i,i}^f = a_{i,i}h_{i,i}^f$. Moreover, we consider that the fading characteristics of the wireless channel follows the Nakagami distribution with the fading parameter, $m_{i,i}$, and the average received power in the fading envelope, $\omega_{i,i}$. Note that the Nakagami distribution has been widely adopted to properly capture the stochastic dynamics of the wireless channel in various communication systems, such as vehicular networks [37], [38], UAV-aided networks [30], [39], aerial-ground integrated networks [40], [41], and many other mobile radio systems [42], [43], due to its high parametric scalability and good fitting performance.[2] To be specific, the probability density function (PDF) of the received signal envelope associated with the agent i is expressed as

$$f_{h_{i,i}^f}\left(x; m_{i,i}, \omega_{i,i}\right) = \frac{2m_{i,i}^{m_{i,i}} x^{2m_{i,i}-1}}{\Gamma\left(m_{i,i}\right)\omega_{i,i}^{m_{i,i}}} \exp\left(-\frac{m_{i,i}}{\omega_{i,i}}x^2\right) \quad (1)$$

for all $x \geq 0$ and $f \in \mathcal{F}$, where the fading parameter $m_{i,i}$ is usually ranging within $[0.5, 5]$ and $\Gamma(m_{i,i})$ is the Gamma function, i.e., $\Gamma(m_{i,i}) = \int_0^\infty s^{m_{i,i}-1}e^{-s}ds$. According to (1), we can further derive the closed-form expression for the cumulative density function (CDF) of the squared signal envelope, $(h_{i,i}^f)^2$, as follows:

$$\Pr\left\{\left(h_{i,i}^f\right)^2 \leq x\right\} = \frac{\gamma\left(m_{i,i}, \frac{m_{i,i}}{\omega_{i,i}}x\right)}{\Gamma\left(m_{i,i}\right)} \quad (2)$$

where the function $\gamma(\cdot,\cdot)$ is the lower incomplete Gamma function, i.e., $\gamma(m, s) = \int_0^s t^{m-1}e^{-t}dt$.

Now, we consider the threshold-based offloading policy in which the agent i can always choose the best channel that has the highest channel transmission gain for its task offloading, i.e., with the frequency $f^* = \arg\max_{f\in\mathcal{F}}\{g_{i,i}^f\}$. Given the offloading threshold $x_i \in \mathbb{R}_+$, for any $f \in \mathcal{F}$, we can get the probability that $h_{i,i}^f$ does not exceed the threshold x_i as follows:

$$\Pr\left\{h_{i,i}^f \leq x_i\right\} = \Pr\left\{(h_{i,i}^f)^2 \leq x_i^2\right\} = \frac{\gamma\left(m_{i,i}, \frac{m_{i,i}}{\omega_{i,i}}x_i^2\right)}{\Gamma\left(m_{i,i}\right)}. \quad (3)$$

Accordingly, the agent i decides to offload its task at the frequency f^* if the selected channel satisfies $h_{i,i}^{f^*} > x_i$. At this point, the probability of task offloading for i, denoted by $\mu_i(x_i)$, can be formulated as $\mu_i(x_i) = \Pr\{h_{i,i}^{f^*} > x_i\} = 1 - \Pr\{h_{i,i}^{f^*} \leq x_i\} = 1 - \Pr\{h_{i,i}^f \leq x_i, f = 1, 2, \ldots, M\}$, i.e.,

$$\mu_i(x_i) = 1 - \prod_{f=1}^{M}\Pr\left\{h_{i,i}^f \leq x_i\right\} = 1 - \left[\frac{\gamma\left(m_{i,i}, \frac{m_{i,i}}{\omega_{i,i}}x_i^2\right)}{\Gamma\left(m_{i,i}\right)}\right]^M \quad (4)$$

where M is the cardinality of the set $\mathcal{F}$. It can be seen from (4) that the offloading decision of the agent is coupled with the stochastic dynamics of the wireless channel.

B. Dynamics of Task Queueing

We let the number of time slots the agent takes for offloading a packet-level task be $\xi_i \in \mathbb{Z}_+$. ξ_i obviously follows the geometric distribution, the PDF of which is:

$$\Pr\{\xi_i = n\} = (1 - \mu_i(x_i))^{n-1}\mu_i(x_i), \ n \in \mathbb{Z}_+. \quad (5)$$

Furthermore, since the expected geometric distribution above is $\mathbb{E}[\xi_i] = 1/\mu_i(x_i)$, to make the system model mathematically tractable, we can use an exponential distribution with

[2] In fact, the Nakagami distribution can be reduced to a Rayleigh distribution or a Ricean distribution, but it gives more scalable control over the extent of the channel fading. This is the main reason that we adopt it in this work. Nevertheless, the methodology developed in this article can be also extended to other fading channels.

Authorized licensed use limited to: University of Waterloo. Downloaded on November 03,2023 at 14:29:46 UTC from IEEE Xplore. Restrictions apply.

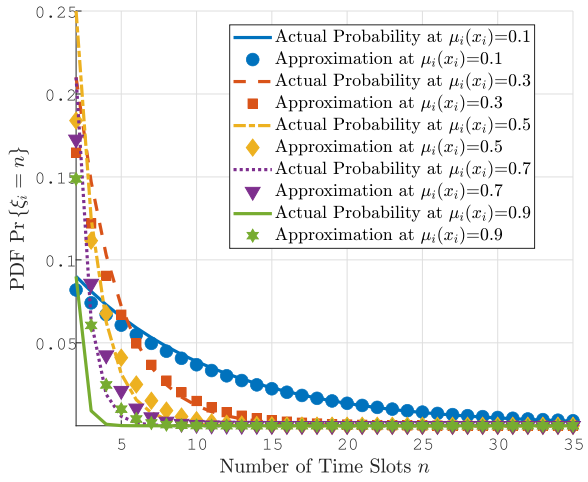

Fig. 2. Comparison between the actual probability distribution of ξ_i and the exponential distribution approximation.

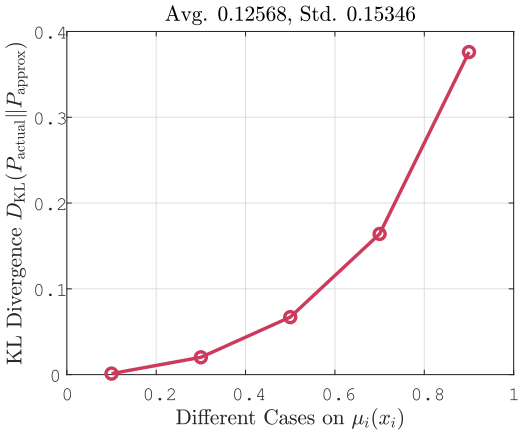

Fig. 3. Test results on the KL divergence between the probability distributions of the time slot number ξ_i obtained by the actual model and its approximated model, respectively.

the parameter $\mu_i(x_i)$ to approximate the geometric distribution [44]. That is

$$\Pr\{\xi_i = n\} \approx \mu_i(x_i) \exp(-\mu_i(x_i)n). \tag{6}$$

Therefore, the offloading rate per unit time can be given as $\phi_i(x_i) = \mu_i(x_i)/\Delta\tau$, which represents the service rate of the agent i's queue buffer. We compare the actual probability distribution of ξ_i with its approximation based on (6) in Fig. 2. Furthermore, we also evaluate the Kullback–Leibler (KL) divergence, $D_{\mathrm{KL}}(P_{\mathrm{actual}} \| P_{\mathrm{approx}})$, between both the actual and the approximated probability distributions P_{actual} and P_{approx} in Fig. 3. Note that the KL divergence is a well-known measure characterizing the difference between two probability distributions from the perspective of information theory, and a KL divergence of 0 indicates that two probability distributions are identical. As can be seen, the KL divergence metric is only about 0.13 on average under different test cases, which is sufficiently small. In Fig. 3, the KL divergence metric even in the worst-performance case is smaller than 1. Combining the results in Figs. 2 and 3, we can confirm that the exponential distribution can approximate the actual distribution with a sufficiently good precision.

Now, we consider the capacity of the agent i's queue buffer to be K_i, and that the packet-level task flow coming from the upper layer of any agent $i \in \mathcal{I}$ follows a Poisson process with an average arrival rate λ_i. Hence, the queueing dynamics of tasks at i can be captured by an M/M/1/K_i queue model. From the viewpoint of queueing theory, the offloading decision-related parameter $\rho_i(x_i) = \lambda_i/\phi_i(x_i)$ represents the traffic intensity. Let Q_i be the number of tasks presented in the queue. The probability that there are k ($0 \le k \le K_i$) tasks presented in the queue buffer in the equilibrium state when a new task arrives can be calculated by

$$\Pr\{Q_i = k\} = \frac{(1-\rho_i(x_i))(\rho_i(x_i))^k}{1-(\rho_i(x_i))^{K_i+1}} \tag{7}$$

where $k = 0, 1, \ldots, K_i$.

Besides, we denote by T_{k+1} the sojourn time (i.e., the sum of both the waiting time and the service time) of the $(k+1)$th arrival task when there already exist k tasks waiting in its front, $k = 0, 1, \ldots, K_i - 1$. The tolerant maximum sojourn time of the task at i is denoted by β_i. Thus, we can use $\Pr\{T_{k+1} < \beta_i | Q_i = k\}$ to denote the probability that the sojourn time of the $(k+1)$th incoming task is shorter than the threshold under the condition that k tasks are presented in the queue, which is given as follows [24]:

$$\Pr\{T_{k+1} < \beta_i | Q_i = k\} = 1 - e^{-\phi_i(x_i)\beta_i} \sum_{v=0}^{k} \frac{(\phi_i(x_i)\beta_i)^v}{v!} \tag{8}$$

for $k = 0, 1, \ldots, K_i - 1$. Following (8), we can also have $\Pr\{T_{k+1} \ge \beta_i | Q_i = k\} = 1 - \Pr\{T_{k+1} < \beta_i | Q_i = k\}$.

Based on (7) and (8), we can further derive the probability that a new incoming task is removed from the queue (for the local execution rather than the remote execution) as follows:

$$\begin{aligned} q_i(x_i) &= \Pr\{Q_i = K_i\} + \sum_{k=0}^{K_i-1} \Pr\{Q_i = k\} \Pr\{T_{k+1} \ge \beta_i | Q_i = k\} \\ &= \frac{(1-\rho_i(x_i))(\rho_i(x_i))^{K_i}}{1-(\rho_i(x_i))^{K_i+1}} \\ &\quad + \sum_{k=0}^{K_i-1} \frac{(1-\rho_i(x_i))(\rho_i(x_i))^k}{1-(\rho_i(x_i))^{K_i+1}} \left(e^{-\phi_i(x_i)\beta_i} \sum_{v=0}^{k} \frac{(\phi_i(x_i)\beta_i)^v}{v!} \right). \end{aligned} \tag{9}$$

It is remarked that the maximum allowed queueing delay β_i can be treated as the Quality-of-Service (QoS)-related aggregate delay requirement of the agent i's upper layer application task. In the queue model, we impose a constraint on the traffic intensity, i.e., $\rho_i(x_i) \le \rho_{\mathrm{upper}}$ where $\rho_{\mathrm{upper}} < 1$ is a specified upper bound of the traffic intensity, in order to guarantee the stability of the queue. According to the definition of $\rho_i(x_i)$, this constraint is equivalent to satisfy

$$C(x_i) \triangleq \mu_i(x_i) - \frac{\lambda_i \Delta\tau}{\rho_{\mathrm{upper}}} \ge 0. \tag{10}$$

Note that $\mu_i(x_i)$ is a monotonically decreasing function with respect to x_i in the domain $\mathbb{R}_+$. $C(x_i)$ is also a monotonically decreasing function with respect to x_i in the domain $\mathbb{R}_+$. Thus,

Authorized licensed use limited to: University of Waterloo. Downloaded on November 03,2023 at 14:29:46 UTC from IEEE Xplore. Restrictions apply.

$C(x_i) \geq 0$ leads to an upper bound on the offloading decision of i, x_i, i.e., $x_i \leq x_i^{\text{upper}}$, where x_i^{upper} is given by solving the following equation:

$$x_i^{\text{upper}} = \arg_{x \in \mathbb{R}_+}\{C(x) = 0\}. \tag{11}$$

C. Dynamics of Coupled Wireless Interference

In the considered multiagent wireless network, the offloading decisions of the agents can heavily rely on the physical-layer performance. A significant metric for the physical-layer link performance is the SINR. Let the profile of the offloading decisions of the agents in $\mathcal{I}$ except agent i be $\mathbf{x}_{-i} \triangleq (x_j)_{j \in \mathcal{I}\backslash\{i\}}$, and the background noise power at the destination of each agent be σ_0^2. The resulting SINR of agent i's offloading session over the wireless channel at frequency $f \in \mathcal{F}$ can be

$$\text{SINR}_i^f(\mathbf{x}_{-i}) = \frac{p_i a_{i,i}^2 \left(h_{i,i}^f\right)^2}{\sigma_0^2 + \sum_{j \in \mathcal{I}\backslash\{i\}} p_j a_{j,i}^2 \left(h_{j,i}^f\right)^2 y_j^f(x_j)} \tag{12}$$

where p_j is the transmit power of agent j, $a_{j,i}$ is the path loss coefficient associated with the interference link from j to the destination of i, and $h_{j,i}^f$ is the interference signal envelope at the same frequency f. $y_j^f(x_j)$ is a binary variable depending on the offloading decision of agent j, which is equal to 1 if j decides to offload its task via the frequency channel f and 0 otherwise. Namely, we have

$$y_j^f(x_j) = \begin{cases} 1, & h_{j,j}^f > x_j \\ 0, & h_{j,j}^f \leq x_j. \end{cases} \tag{13}$$

For simplicity, we rewrite the coupled interference incurred by the other agents to agent i's offloading session as

$$I_i^f(\mathbf{x}_{-i}) = \sum_{j \in \mathcal{I}\backslash\{i\}} p_j a_{j,i}^2 \left(h_{j,i}^f\right)^2 y_j^f(x_j). \tag{14}$$

As shown in (12) and (14), when many agents decide to offload their tasks to their destinations via the same channel, the coupled interference $I_i^f(\mathbf{x}_{-i})$ will increase, which reduces the SINR, $\text{SINR}_i^f(\mathbf{x}_{-i})$, of agent i's offloading link and then makes i's offloading rate degrade. At this point, the individual offloading decisions are connected to the physical-layer communication dynamics and coupled with each other.

Another observation from (14) is that $I_i^f(\mathbf{x}_{-i})$ is a compound random variable due to the fact that it depends on multiple stochastic processes, such as the stochastic fading of the interference link between each j and i, $h_{j,i}^f$, and the j's own offloading link, $h_{j,j}^f$. It is impossible to mathematically derive an exact closed-form expression for the distribution of $I_i^f(\mathbf{x}_{-i})$. Nonetheless, noticing that the squared random variable, $(h_{j,i}^f)^2$, follows a Gamma distribution as shown in (2) and that $y_j^f(x_j)$ is a random binary variable as shown in (13), the value of $I_i^f(\mathbf{x}_{-i})$ is the weighted summation of multiple independent random variables following the Gamma distribution. Thus, we can propose to approximate the PDF of $I_i^f(\mathbf{x}_{-i})$ based on a Gamma distribution function. Let $\mathbb{E}[I_i^f(\mathbf{x}_{-i})]$ be the first-order moment of $I_i^f(\mathbf{x}_{-i})$ and $\text{Var}[I_i^f(\mathbf{x}_{-i})]$ be its second-order moment. In statistics theory, we can have the relationship

$$\begin{aligned} \text{Var}\left[I_i^f(\mathbf{x}_{-i})\right] &= \mathbb{E}\left[\left(I_i^f(\mathbf{x}_{-i}) - \mathbb{E}\left[I_i^f(\mathbf{x}_{-i})\right]\right)^2\right] \\ &= \mathbb{E}\left[\left(I_i^f(\mathbf{x}_{-i})\right)^2\right] - \left(\mathbb{E}\left[I_i^f(\mathbf{x}_{-i})\right]\right)^2. \end{aligned} \tag{15}$$

According to the definition of the first-order moment, we derive $\mathbb{E}[I_i^f(\mathbf{x}_{-i})]$ as follows:

$$\begin{aligned} \mathbb{E}\left[I_i^f(\mathbf{x}_{-i})\right] &= \sum_{j \in \mathcal{I}\backslash\{i\}} p_j a_{j,i}^2 \mathbb{E}\left[\left(h_{j,i}^f\right)^2 y_j^f(x_j)\right] \\ &= \sum_{j \in \mathcal{I}\backslash\{i\}} p_j a_{j,i}^2 \frac{\mu_j(x_j)}{M} \int_0^\infty x^2 f_{h_{j,i}^f}(x; m_{j,i}, \omega_{j,i}) dx. \end{aligned} \tag{16}$$

Due to the fact that for any two agents $j_1, j_2 \in \mathcal{I}\backslash\{i\}$ and $j_1 \neq j_2$, $h_{j_1,i}^f$, $h_{j_2,i}^f$, $y_{j_1}^f(x_{j_1})$ and $y_{j_2}^f(x_{j_2})$ are independent, we can see

$$\mathbb{E}\left[\left(h_{j_1,i}^f\right)^2 y_{j_1}^f(x_{j_1}) \left(h_{j_2,i}^f\right)^2 y_{j_2}^f(x_{j_2})\right] = \frac{\mu_{j_1}(x_{j_1})\mu_{j_2}(x_{j_2})}{M^2} E_{j_1} E_{j_2} \tag{17}$$

where for simplicity E_j ($\forall j \in I$) represents

$$E_j \triangleq \int_0^\infty x^2 f_{h_{j,i}^f}(x; m_{j,i}, \omega_{j,i}) dx. \tag{18}$$

Additionally, for any $j \in \mathcal{I}$, we can have

$$A_j \triangleq \mathbb{E}\left[\left(h_{j,i}^f\right)^4\right] = \int_0^\infty x^4 f_{h_{j,i}^f}(x; m_{j,i}, \omega_{j,i}) dx. \tag{19}$$

Combining the above results can further lead to

$$\begin{aligned} \mathbb{E}\left[\left(I_i^f(\mathbf{x}_{-i})\right)^2\right] &= \sum_{j \in \mathcal{I}\backslash\{i\}} p_j^2 a_{j,i}^4 \frac{\mu_j(x_j)}{M} A_j \\ &\quad + \sum_{j_1 \neq j_2 \in \mathcal{I}\backslash\{i\}} p_{j_1} p_{j_2} a_{j_1,i}^2 a_{j_2,i}^2 \frac{\mu_{j_1}(x_{j_1})\mu_{j_2}(x_{j_2})}{M^2} E_{j_1} E_{j_2}. \end{aligned} \tag{20}$$

Now, combining (15), (16), and (20) can yield the closed-form expression for the second-order moment $\text{Var}[I_i^f(\mathbf{x}_{-i})]$.

Using the first-order and the second-order moments of $I_i^f(\mathbf{x}_{-i})$, we can estimate the shape and the scale parameters for the Gamma distribution, respectively. Specifically, let the shape parameter be $k_i(\mathbf{x}_{-i})$ and the scale parameter be $\theta_i(\mathbf{x}_{-i})$, which inherently depends on the offloading decision profile of the other agents except i, $\mathbf{x}_{-i}$. We can have

$$\begin{cases} k_i(\mathbf{x}_{-i}) = \dfrac{\left(\mathbb{E}\left[I_i^f(\mathbf{x}_{-i})\right]\right)^2}{\text{Var}\left[I_i^f(\mathbf{x}_{-i})\right]} \\ \theta_i(\mathbf{x}_{-i}) = \dfrac{\text{Var}\left[I_i^f(\mathbf{x}_{-i})\right]}{\mathbb{E}\left[I_i^f(\mathbf{x}_{-i})\right]}. \end{cases} \tag{21}$$

The PDF of $I_i^f(\mathbf{x}_{-i})$ is then expressed as

$$f_{I_i^f(\mathbf{x}_{-i})}(x; k_i(\mathbf{x}_{-i}), \theta_i(\mathbf{x}_{-i})) = \frac{x^{k_i(\mathbf{x}_{-i})-1} \exp\left(-\frac{x}{\theta_i(\mathbf{x}_{-i})}\right)}{\theta_i(\mathbf{x}_{-i})^{k(\mathbf{x}_{-i})} \Gamma(k_i(\mathbf{x}_{-i}))} \tag{22}$$

Authorized licensed use limited to: University of Waterloo. Downloaded on November 03,2023 at 14:29:46 UTC from IEEE Xplore. Restrictions apply.

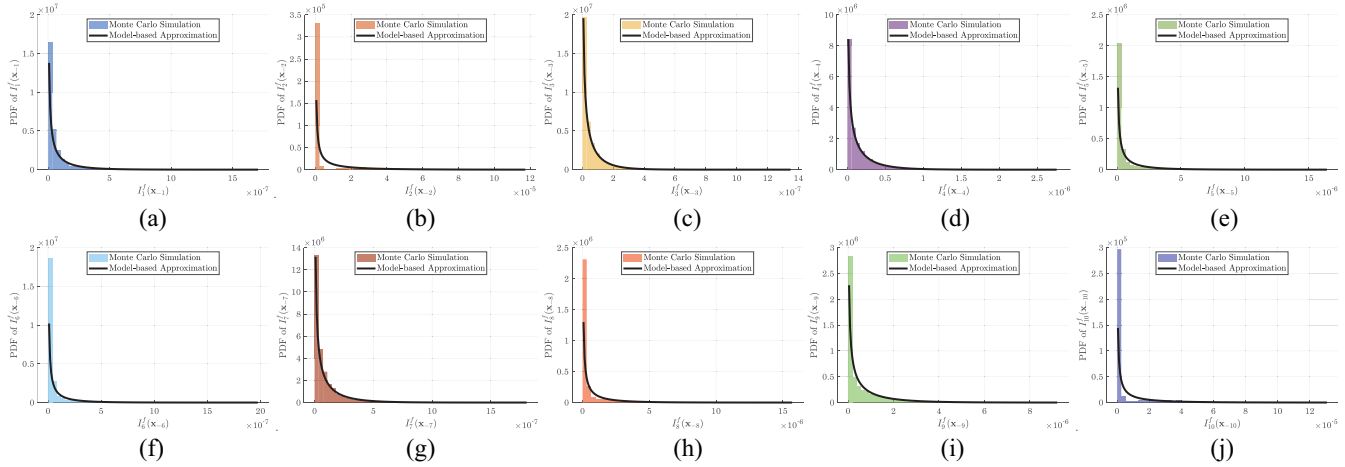

Fig. 4. Comparison of Monte Carlo simulations and model-based approximation on the coupled interference distribution. (a) Agent 1. (b) Agent 2. (c) Agent 3. (d) Agent 4. (e) Agent 5. (f) Agent 6. (g) Agent 7. (h) Agent 8. (i) Agent 9. (j) Agent 10.

where $\Gamma(k_i(\mathbf{x}_{-i}))$ is the Gamma function as follows:

$$\Gamma(k_i(\mathbf{x}_{-i})) = \int_0^\infty s^{k_i(\mathbf{x}_{-i})-1} e^{-s} ds. \tag{23}$$

Now, we conduct extensive Monte Carlo simulations to verify the PDF of the coupled interference function $I_i^f(\mathbf{x}_{-i})$ based on the model (22), in which a region of 300 m $\times$ 300 m with 10 uniformly distributed is set up. The total available channel number M is set to 5. The path loss factor α is fixed at 3.0, the transmit power of each agent p_i is set to 23 dBm, and the noise power σ_0^2 is -96 dBm. In addition, the average received power ω_i is normalized to 1, while the fading parameter $m_{i,i}$ or $m_{i,j}$ is considered to depend on the distance between the transmitter and the corresponding receiver according to the real-world measurement in [37]. The 10 000 Monte Carlo simulations are performed and the simulation-based PDF of the coupled interference received at each agent's destination is compared with the model-based result in Fig. 4. Besides, we also provide the results on the KL divergence $D_{\mathrm{KL}}(P_{\mathrm{simulation}} \| P_{\mathrm{model}})$ between the probability distributions $P_{\mathrm{simulation}}$ and P_{model} obtained by the simulation and the theoretical model, respectively, in Fig. 5. From Fig. 5, it is seen that the KL divergence is only about 0.23 on average and the worst-case metric is still smaller than 1. The small KL divergence indicates that the theoretical distribution approximates the actual observation well. From both Figs. 4 and 5, we can conclude that the proposed theoretical model based on (22) is able to appropriately capture the actual distribution of the coupled interference, which makes the system optimization and analysis tractable.

Next, using the theoretical model (22), we further derive the upper CDF of the coupled interference $I_i^f(\mathbf{x}_{-i})$ as follows:

$$g_i(x, \mathbf{x}_{-i}) \triangleq \Pr\left\{I_i^f(\mathbf{x}_{-i}) > x\right\} = 1 - \frac{\gamma\left(k_i(\mathbf{x}_{-i}), \frac{x}{\theta_i(\mathbf{x}_{-i})}\right)}{\Gamma(k_i(\mathbf{x}_{-i}))}. \tag{24}$$

Given a minimum SINR threshold γ_0 required for a transmission link to correctly receive an offloaded packet, the task

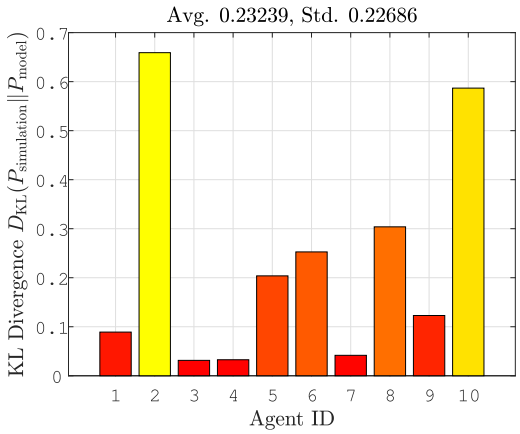

Fig. 5. Test results on the KL divergence between the probability distributions of the coupled interference obtained by Monte Carlo simulations and model-based approximation, respectively.

loss probability due to the packet error for agent i is

$$\begin{aligned} p_i(x_i, \mathbf{x}_{-i}) &\triangleq \Pr\left\{\mathrm{SINR}_i^f(\mathbf{x}_{-i}) y_i^f(x_i) < \gamma_0\right\} \\ &= \Pr\left\{\frac{p_i a_{i,i}^2 \left(h_{i,i}^f\right)^2 y_i^f(x_i)}{\gamma_0} - \sigma_0^2 < I_i^f(\mathbf{x}_{-i})\right\}. \end{aligned} \tag{25}$$

Combining (1) and (24), we can derive the joint CDF of $h_{i,i}^f$ and $I_i^f(\mathbf{x}_{-i})$, which results in the closed-form expression for $p_i(x_i, \mathbf{x}_{-i})$ as follows:

$$p_i(x_i, \mathbf{x}_{-i}) = \int_{x_i}^\infty f_{h_{i,i}^f}(x; m_{i,i}, \omega_{i,i}) g_i\left(\frac{p_i a_{i,i}^2 x^2}{\gamma_0} - \sigma_0^2, \mathbf{x}_{-i}\right) dx. \tag{26}$$

From (26), we can see that the decision-making behaviors of the agents interact with each other via the coupled wireless interference function.

In addition, note that $[(p_i a_{i,i}^2 x^2)/(\gamma_0)] - \sigma_0^2 \geq 0$ in (26). We can obtain a lower bound on the offloading decision of agent

Authorized licensed use limited to: University of Waterloo. Downloaded on November 03,2023 at 14:29:46 UTC from IEEE Xplore. Restrictions apply.

i, x_i^{lower}, as follows:

$$x_i \geq x_i^{\text{lower}} = \sqrt{\frac{\gamma_0 \sigma_0^2}{p_i a_{i,i}^2}}. \tag{27}$$

D. Global Optimization Model for Task Offloading

Taking into account the queueing dynamics and the coupled wireless interference among the interactive agents, we can derive the expected successful offloading rate (i.e., the expected offloading throughput successfully completed by the agent), $R_i(x_i, \mathbf{x}_{-i})$, of any agent $i \in \mathcal{I}$ as follows:

$$R_i(x_i, \mathbf{x}_{-i}) = \lambda_i \big[1 - q_i(x_i) - (1 - q_i(x_i)) p_i(x_i, \mathbf{x}_{-i})\big] \tag{28}$$

where the offloading decision x_i is within $[x_i^{\text{lower}}, x_i^{\text{upper}}]$. Let $\mathcal{X}_i$ denote the bound constraint on each offloading decision x_i, i.e., $\mathcal{X}_i \triangleq [x_i^{\text{lower}}, x_i^{\text{upper}}]$. The feasible space for the decision profile of all the agents is denoted by $\mathcal{X} \triangleq \prod_i^N \mathcal{X}_i$. We formulate an optimization model for maximizing the global expected successful offloading rates of the agents as follows:

$$\mathcal{P}: \max_{\mathbf{x}} \; J(\mathbf{x}) = \sum_{i=1}^{N} R_i(x_i, \mathbf{x}_{-i}) \quad \text{s.t.} \; \begin{cases} \mathbf{x} = (x_i)_{i \in \mathcal{I}} \\ \mathbf{x} \in \mathcal{X}. \end{cases} \tag{29}$$

We remark from (29) that the system model jointly takes into account the coupled impacts of the physical-layer transmission error rate and the upper layer packet loss probability, which allows a QoS-oriented optimization design to be realized for the agents. The box constraints in the optimization model are also proposed according to the queueing dynamics and the coupling interference condition, such that a nonempty feasible solution space can be guaranteed.

IV. Distributed Task Offloading Optimization

In this section, we transform the original system model (29) into a game-theoretic formulation in order to develop a distributed low-complexity multiagent algorithm. The key idea is to enable agents to self-organize the allocation of the available spectral resource without the assistance of a centralized control by exploiting the game-theoretic approach. We first analyze the structural properties of the primal problem from the game-theoretical perspective and then propose a distributed best response algorithm for task offloading optimization of the agents, in which an augmented Lagrangian optimization is proposed and embedded to solve the individual optimization problem of each agent.

A. Game-Theoretical Model and Nash Equilibrium Analysis

From the game-theoretical sense, the original problem can be easily modeled as the following individual utility maximization problem, i.e., a normal game formulation, which is naturally suitable for the distributed computation:

$$\mathcal{G}_1: \max_{x_i \in \mathcal{X}_i} \; R_i(x_i, \mathbf{x}_{-i}) \quad \forall i \in \mathcal{I}. \tag{30}$$

However, it is difficult to directly analyze the game-theoretical properties from $\mathcal{G}_1$ due to the fact that the individual objective function (i.e., its individual utility function) is highly nonlinear, and neither nonconvex nor nonconcave with respect to the continuous individual decision x_i and the joint decisions $\mathbf{x}_{-i}$. To address this challenge, we transform our analysis viewpoint from the threshold-based optimization to the strategy-based optimization. That is, we note that each agent can only take two different decision actions, i.e., to offload or not offload its task in a time slot. Thus, rather than treating the thresholds x_i ($i \in \mathcal{I}$) as the decision variables to be optimized, we consider to optimize the strategy of each agent, i.e., the selection probability distribution over the finite decision actions of each agent. In fact, from (4), the probability of agent i deciding to offload a task can be denoted by μ_i, and the probability of deciding not to offload a task is $1 - \mu_i$. Once an optimal offloading probability μ_i^* can be obtained, an optimal threshold x_i^* can also be uniquely determined by solving the univariate equation $x_i^* = \arg_{x_i}\{\mu_i^* - \mu_i(x_i) = 0\}$. Therefore, denoting the policy of agent i by $\pi_i \triangleq (\mu_i, 1 - \mu_i)$, we can equivalently transform $\mathcal{G}_1$ to another game formulation $\mathcal{G}_2$

$$\mathcal{G}_2: \max_{\pi_i \in [0,1]^2} \; R_i(\pi_i, \boldsymbol{\pi}_{-i}) \quad \forall i \in \mathcal{I} \tag{31}$$

where $R_i(\pi_i, \boldsymbol{\pi}_{-i})$ is the counterpart of $R_i(x_i, \mathbf{x}_{-i})$ and it denotes the expected successfully offloading rate of agent i with respect to its own decision policy π_i when given the policy profile of the other agents $\boldsymbol{\pi}_{-i}$, i.e., $\boldsymbol{\pi}_{-i} = \prod_{j \in \mathcal{I} \setminus \{i\}} \pi_j$. Based on $\mathcal{G}_2$, we obtain the following results.

Lemma 1 (Existence of Mixed Strategy): For any finite game, there exists at least a mixed-strategy Nash equilibrium.

Proof: This lemma follows John Forbes Nash's famous theory, in which every finite game must have at least a (mixed-strategy) Nash equilibrium while not all the games have pure-strategy Nash equilibria. ■

Lemma 2: For the finite game $\mathcal{G}_2$, there do not exist pure-strategy Nash equilibria.

Proof: We prove this lemma by contradiction as follows. First, we assume that there exists at least a pure-strategy Nash equilibrium. According to the definition of a pure-strategy Nash equilibrium, it must hold in the pure-strategy Nash equilibrium that $\mu_i \in \{0, 1\}$ for all $i \in \mathcal{I}$. Thus, we can observe two situations.

1) If at least an agent, for instance $i \in \mathcal{I}$, sets $\mu_i = 0$ in the pure-strategy Nash equilibrium, indicating that this agent decides not to offload its task at all, its received utility is then zero, i.e., $R_i(\pi_i, \boldsymbol{\pi}_{-i}) = 0$. This means that agent i cannot get any benefit from the game. But, in fact, it can get a nonzero utility $R_i(\pi_i, \boldsymbol{\pi}_{-i})$ as long as it unilaterally changes its action, i.e., deciding to offload tasks. At this point, the system does not reach the Nash equilibrium, which is contrary to the assumption.
2) Otherwise, all the agents can only set their offloading probabilities to 1 in the pure-strategy Nash equilibrium, i.e., $\mu_i = 1$ for all $i \in \mathcal{I}$. This indicates that all the agents decide to transmit all the time. In such a situation, the multiagent system will have the worst communication performance because the wireless coupled interference

Authorized licensed use limited to: University of Waterloo. Downloaded on November 03,2023 at 14:29:46 UTC from IEEE Xplore. Restrictions apply.

is the most intensive. Namely, the interference $I_i^f(\mathbf{x}_{-i})$ attains its maximum level for each i. In this state, as long as a small fraction of the agents change to keep silent at certain time slots, i.e., deciding not to transmit, the agents can improve its offloading performance. Thus, this state is not a Nash equilibrium, which is also contrary to the assumption.

To sum up, the contradiction always exists in the pure-strategy Nash equilibrium. Therefore, the game $\mathcal{G}_2$ cannot have a pure-strategy Nash equilibrium. ■

Theorem 1: For the finite game $\mathcal{G}_2$, there must exist at least a Nash equilibrium and all its Nash equilibria can only be mixed-strategy.

Proof: The theorem follows Lemmas 1 and 2. ■

B. Distributed Best-Response Iterative Framework

In game theory, a Nash equilibrium point is always in coincidence with the fixed point of a best-response mapping. Based on this fact and Theorem 1, we can design a distributed best-response (DBR) iterative framework to compute the mixed-strategy Nash equilibrium solution. Namely, considering the connection between $\mathcal{G}_1$ and $\mathcal{G}_2$, we are allowed to solve a specific point $\mathbf{x}^*$ corresponding to the mixed-strategy Nash equilibrium $\boldsymbol{\pi}^*$ by the best response method. We denote by $k \in \mathbb{Z}_+$ the iteration. Let $x_i^*[k]$ be the optimal offloading decision of agent i and $\mathbf{x}_{-i}^*[k]$ the optimal offloading decision profile of the others at iteration k. The maximum iteration number of agent i is specified as maxIter_i. The proposed DBR framework is detailed in Algorithm 1. To be specific, to compute the best response, each agent i first needs to estimate the statistics on the physical-layer interference, i.e., the parameters $k_i(\mathbf{x}_{-i}^*)$ and $\theta_i(\mathbf{x}_{-i}^*)$, which are related to the others' decisions $\mathbf{x}_{-i}^*$. Importantly, this can be realized locally without any information exchange among the global agents. In reality, each agent i can track the physical-layer interference experienced in a series of successive time slots and then estimate the statistics parameters by using the interference measures. In this manner, the agents only need to solve their individual optimization locally as in Algorithm 1. Thus, they are enabled to make offloading decisions in a parallel and separate manner and collaborate with each other. In addition, Fig. 1 shows how each source agent can make offloading decisions based on our proposed method in a distributed manner in an exemplary application scenario with D2D communications. In Fig. 1, the source agents compute their own optimal thresholds for offloading decision making in parallel based on an embedded ULO scheme, which is detailed in the following section.

C. Individual Offloading Optimization Scheme

As shown in Algorithm 1, it is important to obtain an optimal offloading decision $x_i^*[k]$ at each k for each agent $i \in \mathcal{I}$. To address the local optimization problem (32) regarding its nonlinearity and nonconvexity, we aim at developing a local optimization scheme. Specifically, for any $i \in \mathcal{I}$, we first transform the bound constraint $x_i \in \mathcal{X}_i$ to two inequalities $b_{i,1}(x_i) = x_i - x_i^{\text{lower}} \geq 0$ and $b_{i,2}(x_i) = x_i^{\text{upper}} - x_i \geq 0$ and let $\mathbf{b}_i(x_i) = [b_{i,1}(x_i), b_{i,2}(x_i)]^{\mathrm{T}}$. Then, we can derive the following result.

Algorithm 1: DBR Framework for Multiagent Offloading Optimization

```
/* Initialization                                          */
Set iteration k = 0 and uniformly and randomly select
  x_i^*[k] ∈ X_i for all i ∈ I.
/* Loop for iterations by each agent
     i ∈ I                                                 */
while k ≤ maxIter_i for each i ∈ I do
    /* Sense the channel quality                          */
    Estimate k_i(x_{-i}^*[k]) and θ_i(x_{-i}^*[k]) via tracking the
    experienced interference locally.
    /* Do best response                                   */
    Solve the optimal individual decision x_i^*[k+1]
        x_i^*[k+1] ∈ arg max_{x_i ∈ X_i} {R_i(x_i, x_{-i}^*[k])}.   (32)
    /* Update iteration                                   */
    set k = k + 1.
```

Theorem 2: For any $i \in \mathcal{I}$, let x_i^* is an optimal point for the following unconstrained optimization problem $\mathcal{G}_3$. x_i^* is also an optimal point for the individual offloading optimization problem $\mathcal{G}_1$ in (30)

$$\mathcal{G}_3 : \min_{x_i, \mathbf{w}_i} : \phi_i(x_i, \mathbf{w}_i, \sigma_i) \tag{33}$$

where $\mathbf{w}_i$ represents $\mathbf{w}_i = [w_{i,1}, w_{i,2}]^{\mathrm{T}}$, which are the Lagrangian multipliers; $\sigma_i \in \mathbb{R}_+$ is a sufficiently large real parameter; $\phi_i(x_i, \mathbf{w}_i, \sigma_i)$ is defined as

$$\phi_i(x_i, \mathbf{w}_i, \sigma_i) \triangleq -R_i(x_i, \mathbf{x}_{-i}) + \frac{1}{2\sigma_i} \sum_{l=1}^{2} \Big(g_{i,l}^2\big(x_i, w_{i,l}, \sigma_i\big) - w_{i,l}^2 \Big). \tag{34}$$

The auxiliary function $g_{i,l}(x_i, w_{i,l}, \sigma_i)$ $(l = 1, 2)$ is given by

$$g_{i,l}\big(x_i, w_{i,l}, \sigma_i\big) = \max\big\{0, w_{i,l} - \sigma_i b_{i,l}(x_i)\big\}. \tag{35}$$

Proof: To prove the theorem, we construct the augmented Lagrange function. To achieve this, we first transform the problem $\mathcal{G}_1$ into the following equality-constrained minimization problem by introducing two auxiliary variables $y_{i,l} \in \mathbb{R}$, $l = 1, 2$:

$$\begin{aligned} \min_{x_i, \mathbf{y}_i} :\ & -R_i(x_i, \mathbf{x}_{-i}) \\ \text{s.t.}\ \ & b_{i,l}(x_i) - y_{i,l}^2 = 0,\ \ l = 1, 2. \end{aligned} \tag{36}$$

From the optimization above, we can construct the augmented Lagrangian function with the multipliers $\mathbf{w}_i$ and the penalty parameter σ_i as follows:

$$\begin{aligned} \mathcal{L}\big(x_i, \mathbf{y}_i, \mathbf{w}_i, \sigma_i\big) = & -R_i(x_i, \mathbf{x}_{-i}) - \sum_{l=1}^{2} w_{i,l} \Big(b_{i,l}(x_i) - y_{i,l}^2 \Big) \\ & + \frac{\sigma_i}{2} \sum_{l=1}^{2} \Big(b_{i,l}(x_i) - y_{i,l}^2 \Big)^2 \end{aligned} \tag{37}$$

Authorized licensed use limited to: University of Waterloo. Downloaded on November 03,2023 at 14:29:46 UTC from IEEE Xplore. Restrictions apply.

where $\boldsymbol{y}_i = [y_{i,1}, y_{i,2}]^{\mathrm{T}}$. Using this augmented Lagrangian function, the problem in (36) can be equivalent to an unconstrained optimization with additional decision variables $\boldsymbol{y}_i$

$$\min_{x_i, \boldsymbol{y}_i}: \ \mathcal{L}\big(x_i, \boldsymbol{y}_i, \mathbf{w}_i, \sigma_i\big). \tag{38}$$

Next, we are allowed to solve (38) with respect to $\boldsymbol{y}_i$ for obtaining the optimal $\boldsymbol{y}_i^*$. Specifically, we can rearrange the augmented Lagrangian function $\mathcal{L}(x_i, \boldsymbol{y}_i, \mathbf{w}_i, \sigma_i)$ as

$$\begin{aligned}\mathcal{L}\big(x_i, \boldsymbol{y}_i, \mathbf{w}_i, \sigma_i\big) = &-R_i(x_i, \mathbf{x}_{-i}) \\ &+ \sum_{l=1}^{2} \left\{ \frac{\sigma_i}{2} \left[y_{i,l}^2 - \frac{1}{\sigma_i}\big(\sigma_i b_{i,l}(x_i) - w_{i,l}\big) \right]^2 \right. \\ &\left. - \frac{w_{i,l}^2}{2\sigma_i} \right\}. \end{aligned} \tag{39}$$

From this result above, we can find that when fixing $x_i, \mathbf{w}_i, \sigma_i$, the augmented Lagrangian function can attain its minimum value in two situations with respect to $\boldsymbol{y}_i$

$$y_{i,l}^2 = \begin{cases} \frac{(\sigma_i b_{i,l}(x_i) - w_{i,l})}{\sigma_i}, & \text{if } \sigma_i b_{i,l}(x_i) - w_{i,l} \geq 0 \\ 0, & \text{if } \sigma_i b_{i,l}(x_i) - w_{i,l} < 0. \end{cases} \tag{40}$$

Combining (39) and (40) then results in the new objective function as given in (34). As thus, the theorem is proven. ■

In addition, substituting (40) into the equality constraint in (36) can get

$$h_{i,l}\big(x_i, w_{i,l}\big) = b_{i,l}(x_i) - y_{i,l}^2 = \min\left(\frac{w_{i,l}}{\sigma_i}, b_{i,l}(x_i) \right) \tag{41}$$

for $l = 1, 2$. Thus, we can define

$$\mathbf{h}_i(x_i, \mathbf{w}_i) = \big[h_{i,1}(x_i, i, 1), h_{i,2}\big(x_i, w_{i,2}\big) \big]^{\mathrm{T}} \tag{42}$$

for simplicity. Now, based on Theorem 2, we propose an ULO scheme to obtain the individual optimal decision $x_i^*[k+1]$ for each $i \in \mathcal{I}$, which is embedded in the DBR framework in Algorithm 1. Let a tolerant error be $\epsilon > 0$, two constants be $r > 1$ and $\eta \in (0, 1)$. The iteration for solving $\mathcal{G}_3$ is indexed by $k' \in \mathbb{Z}_+$. The ULO scheme is summarized in Algorithm 2.

Algorithm 2: ULO for Individual Offloading Optimization

Input: The previous offloading decision $x_i^*[k]$.
Output: The current offloading decision $x_i^*[k+1]$.
/* Initialization */
Set $x_i^{(0)} = x_i^*[k]$ and initialize an estimate on $\mathbf{w}_i^{(1)}$.
Set $k' = 1$ and set a while-loop flag FLAG = TRUE.
/* While loop for iterations by each agent */
while FLAG *is* TRUE **do**
  /* Solve the unconstrained optimization */
  Solve $x_i^{(k')} = \operatorname{argmin}\ \phi_i(x_i, \mathbf{w}_i^{(k')}, \sigma_i)$ with initialization at the previous point $x_i^{(k'-1)}$.
  /* Check the stopping condition */
  if $\|\mathbf{h}_i(x_i^{(k')}, \mathbf{w}_i^{(k')})\| \leq \epsilon$ **then**
    Set FLAG = FALSE.
  /* Adapt the penalty parameter */
  if $\frac{\|\mathbf{h}_i(x_i^{(k')}, \mathbf{w}_i^{(k')})\|}{\|\mathbf{h}_i(x_i^{(k'-1)}, \mathbf{w}_i^{(k'-1)})\|} \geq \eta$ **then**
    Update $\sigma_i = r\sigma_i$.
  /* Update the Lagrangian multipliers */
  Update $w_{i,l}^{(k'+1)} = g_{i,l}(x_i^{(k')}, w_{i,l}^{(k')}, \sigma_i)$ for $l = 1, 2$.
  Set $k' = k' + 1$.
Return $x_i^*[k+1] = x_i^{(k')}$.

D. Complexity Analysis

This section is devoted to the complexity analysis on the proposed DBR algorithm (Algorithm 1) with the individual constrained nonlinear optimization (Algorithm 2) in the worst case. As can be seen, thanks to the distributed computation paradigm, the most computational cost is incurred only by performing Algorithm 2 locally and independently at each agent. For Algorithm 2, let the upper bound of the Lagrangian penalty parameter σ_i for any agent $i \in \mathcal{I}$ be σ_i^{upper}, i.e., $\sigma_i \leq \sigma_i^{\text{upper}}$ for all k', and the initial penalty is denoted by $\sigma_i^{(0)}$. Thus, the number of iterations, denoted by N_{σ_i}, such that the penalty under the update $\sigma_i^{(k'+1)} = r\sigma_i^{(k')}$ as in Algorithm 2 is bounded above by $\sigma_i^{(N_{\sigma_i})} = r^{N_{\sigma_i}}\sigma_i^{(0)} \leq \sigma_i^{\text{upper}}$, i.e.,

$$N_{\sigma_i} \leq \frac{\log_2\left(\frac{\sigma_i^{\text{upper}}}{\sigma_i^{(0)}} \right)}{\log_2(r)}. \tag{43}$$

Indeed, the right term of (43) is also the bound for the number of iterations at which the condition $[\|\mathbf{h}_i(x_i^{(k')}, \mathbf{w}_i^{(k')})\| / \|\mathbf{h}_i(x_i^{(k'-1)}, \mathbf{w}_i^{(k'-1)})\|] \geq \eta$ does hold.

Similarly, due to the continuity of $\mathbf{b}_i(x_i)$, the compactness of the closed domain $\mathcal{X}_i$ of x_i, and the boundedness of the Lagrangian multipliers $\mathbf{w}_i$, there exists an upper bound for $\|\mathbf{h}_i(x_i^{(k')}, \mathbf{w}_i^{(k')})\|$ for all k'. Let such an upper bound be ϵ^{upper}. Therefore, if Algorithm 2 performs at least N_ϵ consecutive iterations during which $[\|\mathbf{h}_i(x_i^{(k')}, \mathbf{w}_i^{(k')})\| / \|\mathbf{h}_i(x_i^{(k'-1)}, \mathbf{w}_i^{(k'-1)})\|] \geq \eta$ dose not hold, i.e.,

$$\left\| \mathbf{h}_i\left(x_i^{(N'_\epsilon)}, \mathbf{w}_i^{(N'_\epsilon)} \right) \right\| \leq \eta^{N_\epsilon} \left\| \mathbf{h}_i\left(x_i^{(0)}, \mathbf{w}_i^{(0)} \right) \right\| \leq \epsilon \tag{44}$$

the upper bound of N_ϵ can be derived by

$$N_\epsilon \leq \frac{\log_2\left(\frac{\epsilon}{\left\| \mathbf{h}_i\left(x_i^{(0)}, \mathbf{w}_i^{(0)} \right) \right\|} \right)}{\log_2(\eta)} \leq \frac{\log_2\left(\frac{\epsilon}{\epsilon^{\text{upper}}} \right)}{\log_2(\eta)}. \tag{45}$$

Combining the results of (43) and (45), we can obtain the complexity of Algorithm 2 for convergence in the worst case as follows:

$$\mathcal{O}\left(\frac{\log_2\left(\frac{\sigma_i^{\text{upper}}}{\sigma_i^{(0)}} \right)}{\log_2(r)} \times \frac{\log_2\left(\frac{\epsilon}{\epsilon^{\text{upper}}} \right)}{\log_2(\eta)} \right). \tag{46}$$

In addition, Algorithm 2 involves solving an unconstrained nonconvex univariate optimization (33). Hence, we can let $c\epsilon^{-\chi}$ be the upper bound of the number of iterations and evaluations that are needed by a specific unconstrained optimization algorithm to obtain an ϵ-precision on the first-order optimality

Authorized licensed use limited to: University of Waterloo. Downloaded on November 03,2023 at 14:29:46 UTC from IEEE Xplore. Restrictions apply.

condition of (33) at each k', i.e., $\|\nabla_{x_i}\phi_i(x_i, \mathbf{w}_i^{(k')}, \sigma_i)\| \leq \epsilon$. The parameters c and χ depend on the adopted algorithm and the characteristics of the targeted problem. For example, as shown in [45], the steepest descent method usually requires $\mathcal{O}(\epsilon^{-2})$ iterations for solving an unconstrained nonconvex optimization with ϵ-precision, while the complexity bound of the cubically regularized Newton methods can be $\mathcal{O}(\epsilon^{-3/2})$. Hence, based on (46) and recalling the given maximum number of iterations in Algorithm 1, maxIter_i, the worst-case complexity bound of each agent i's optimization computation in Algorithm 1 is approximated by

$$\mathcal{O}\left(\text{maxIter}_i \times c\epsilon^{-\chi} \times \frac{\log_2\left(\frac{\sigma_i^{\text{upper}}}{\sigma_i^{(0)}}\right)}{\log_2(r)} \times \frac{\log_2\left(\frac{\epsilon}{\epsilon^{\text{upper}}}\right)}{\log_2(\eta)}\right). \quad (47)$$

From (46) and (47), it is seen that the complexity bounds mainly depend on the optimality tolerance ϵ. Moreover, noticing that $\log_2(\epsilon^{-1}) \leq \epsilon^{-1}$ for $\epsilon \in (0, 1)$, (47) is also bounded above by $\mathcal{O}(C\epsilon^{-(\chi+1)})$ where C is related to the other algorithmic parameters except ϵ in (47). The above results indicate that the algorithm is approximately of the polynomial complexity with respect to the optimality precision ϵ.

V. Performance Evaluation

In this section, we evaluate the performance of the proposed distributed offloading optimization and compare it with several other representative methods by simulation experiments.

A. Simulation Setup

For the sake of implementing a consistent comparison, we consider a computation offloading scenario with size of 300 m × 300 m, where multiple pairs of source agents and their destinations are uniformly distributed. The offloading distance between the source and the destination is randomly selected from the range [10, 100] (m). The average received power ω_i for each agent i is normalized to 1, while the fading channel parameters $m_{i,i}$ or $m_{j,i}$ are considered to depend on the link distance as from [37]. The numerical results are obtained with setting $\alpha = 3.0$, $p_i = 23$ dBm, and $\sigma_0^2 = -96$ dBm. The SINR threshold is set to a typical value as $\gamma_0 = 10$, the traffic intensity constraint is $\rho_{\text{upper}} = 0.9$, and the unit time slot is $\Delta\tau = 5$ ms. The algorithm related parameters are set by $\text{maxIter}_i = 20$, $\epsilon = 10^{-4}$, $\eta = 0.5$, $r = 10$, $\sigma_i^{(0)} = 3.0$, and the Lagrangian multipliers are initialized as $\mathbf{w}_i^{(0)} = [0, 0]^{\text{T}}$. It is remarked that the subproblem (33) is an univariate unconstrained optimization which can be efficiently solved by using many existing numerical optimization algorithms like Newton's methods. All the simulation experiments are carried out on a 2.2-GHz Intel Core i7-8750H CPU with 8-GB RAM. The other system model parameters, such as the number of the source agents N, the available channel number M, the task arrival rate λ_i, the sojourn time threshold β_i, and the queue capacity K_i, will be varied to simulate different scenario conditions for performance comparison.

Besides, we would like to compare our DBR method, denoted by "DBR," with a centralized global optimization method based on swarm intelligence, i.e., the centralized particle swarm optimization method (PSO), and with other three distributed methods, including the distributed stochastic learning method (DSL), the distributed dual-decomposition method (DDD), and the distributed aggressive policy-based method (DAP). To be specific, we implement the centralized global PSO as a performance benchmark to solve the primal system model $\mathcal{P}$ given in (29). The DSL and the DDD are two well-known representative solutions, which are based on two different methodologies, i.e., game theory and dual optimization theory, respectively. They have been widely adopted in the field of computation offloading as in [21]–[23]. The agents with the DSL use their individual expected offloading rate as a reinforcement signal to learn the probabilities of properly selecting the offloading action and the keeping-silent action and then make offloading decisions according to the action selection probabilities in a distributed manner. With the DDD, the primal problem $\mathcal{P}$ is transformed into a dual optimization formulation that is further solved by using a distributed subgradient-descent algorithm. With the aggressive policy-based method, the agents decide to offload their tasks all the time.

Fig. 6. Evolution of the individual expected successful offloading rate $R_i(x_i, \mathbf{x}_{-i})$ for each agent i.

B. Global Convergence and Optimality

To demonstrate the validity of the proposed method in terms of convergence and optimality, we perform the simulation experiment with $N = 10$, $M = 10$, $\lambda_i = 100$ packets/s, $\beta_i = 50$ ms, and $K_i = 20$ for all i. In Fig. 6, the individual expected successful offloading rate $R_i(x_i, \mathbf{x}_{-i})$ of each agent i is shown varying with the individual iteration. As observed, the agents can converge to a steady state by only performing a few iterations, which is a Nash equilibrium since no one can improve its individual performance by unilaterally changing its offloading decision. Moreover, we compare our method with the centralized PSO method in term of the decision-making evolution in Fig. 7. From this figure, it can be seen that the proposed DBR can well approximately converge to the decision solution in the steady state as obtained by the centralized PSO. Besides, Fig. 7 also shows that our method only requires

Authorized licensed use limited to: University of Waterloo. Downloaded on November 03,2023 at 14:29:46 UTC from IEEE Xplore. Restrictions apply.

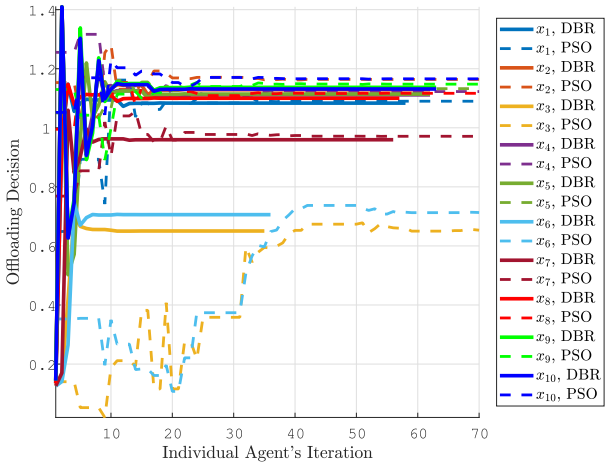

Fig. 7. Evolution of the individual offloading decision x_i.

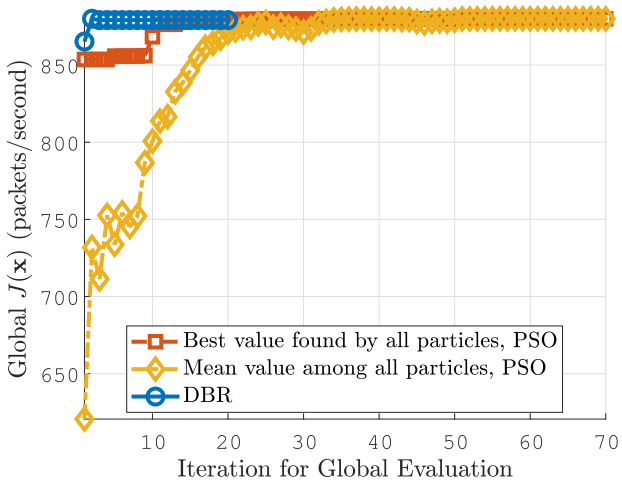

Fig. 8. Evolution of the global performance.

a fewer number of individual iterations than the centralized optimization. The main reason is that the centralized PSO needs to search the solution space in a stochastic optimization manner in an early iteration stage to avoid converging to a local minimum. In Fig. 8, we compare the global performance of our proposed method with the centralized PSO. Fig. 8 illustrates that the final system-wide performance achieved by our method is about 879.167 packets/s, which is quite closed to the global optimal performance (about 879.988 packets/s) achieved by the centralized PSO while our method can converge more faster than the centralized PSO. Additionally, we also carry out the other methods in this experiment and Fig. 9 compares the global performance of these different methods. It can be seen that the global performance of our method is the closest to the global optimality provided by the centralized PSO and our method can achieve about 27.29% and 27.56% improvement over the DSL and the DDD methods, respectively. The DAP method has the worst performance since it can incur sever wireless coupled interferences among the agents.

C. Global Performance Comparison

In the section, we compare our method with the centralized PSO and the other three distributed methods, i.e., DSL, DDD, and DAP, under different situations to demonstrate the advantage of our method from a comprehensive perspective.

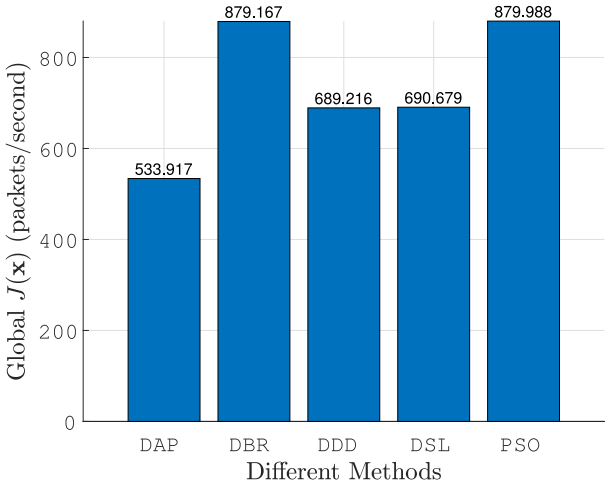

Fig. 9. Performance comparison among different methods.

1) Social Welfare Comparison: For the comparison of the social welfare under different methods, we first present the performance metric reflecting the social welfare in the multiagent computation offloading network. According to game theory, the social welfare of players is defined as the overall net benefit gained by the players from a game, which can be formulated as the difference between the total payoffs received by the players and the total cost incurred in their game [46], [47]. In our targeted scenario, the individual payoff utility is directly related to the benefit gained by each agent from the offloading game, i.e., his expected successful task offloading rate $R_i(x_i, \mathbf{x}_{-i})$, and the load demand from the application layer, i.e., his average task arrival rate λ_i. Let $U_i(x_i, \mathbf{x}_{-i})$ be the agent i's payoff utility. Thus, we formulate $U_i(x_i, \mathbf{x}_{-i})$ by

$$U_i(x_i, \mathbf{x}_{-i}) = \frac{R_i(x_i, \mathbf{x}_{-i})}{\lambda_i}, i = 1, \ldots, N. \tag{48}$$

On the other side, according to [46] and [47], the incurred cost of each agent in the offloading game can be formulated as a quadratic function of his power consumption. Denote the agent i's transmission power by P_i, and the coefficients of the quadratic cost function by $c_{i,1}$, $c_{i,2}$, $c_{i,3}$. The individual cost function can be expressed as follows [46], [47]:

$$U_i(x_i, \mathbf{x}_{-i}) = C_i(P_i) = c_{i,1}P_i^2 + c_{i,2}P_i + c_{i,3}. \tag{49}$$

By combining the above individual payoff (48) and the individual cost (49), the overall social welfare of the multiagent network, denoted by SW, is represented as

$$\text{SW} = \sum_{i\in\mathcal{I}} U_i(x_i, \mathbf{x}_{-i}) - \sum_{i\in\mathcal{I}} C_i(P_i). \tag{50}$$

We set the transmission power $P_i = 23$ dBm, the average task arrival rate $\lambda_i = 100$ packets/s, and the cost coefficients $c_{i,1} = c_{i,2} = c_{i,3} = 10^{-3}$ for all $i \in \mathcal{I}$ for the sake of performance comparison. The social welfare results obtained by different methods under different numbers of agents are shown in Fig. 10. As can be seen, the social welfare metric

Authorized licensed use limited to: University of Waterloo. Downloaded on November 03,2023 at 14:29:46 UTC from IEEE Xplore. Restrictions apply.

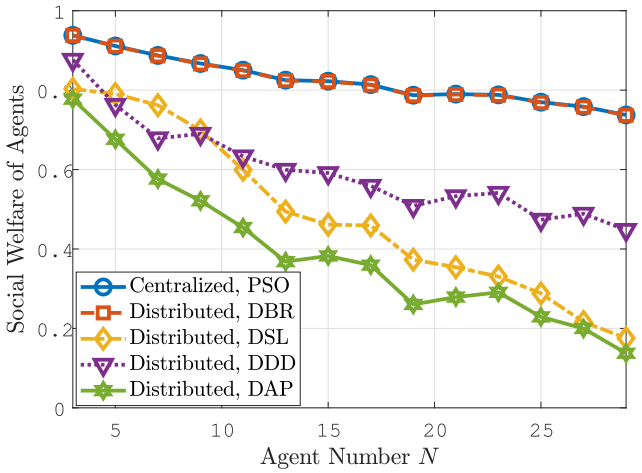

Fig. 10. Social welfare comparison under different agent numbers.

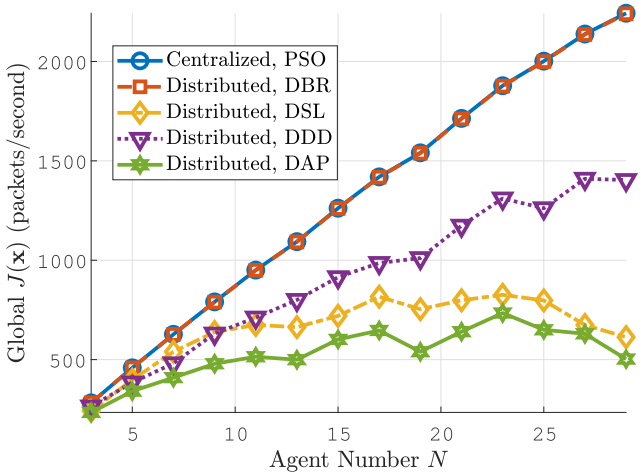

Fig. 11. Performance comparison under different agent numbers.

is decreased along with increasing the agent number, which is due to the fact that more agents result in severer resource competition and higher communication interference. However, our proposed method, DBR, can still achieve a comparable social welfare with the centralized optimization, PSO. The average social welfare gap between our DBR result and the centralized optimization result is about 0.0981%. More importantly, our method can significantly outperform the other distributed methods, DSL, DDD and DAP. Specifically, the average social welfare obtained by our method is about 69.49%, 37.58%, and 109.30% higher than that of DSL, DDD, and DAP methods, respectively.

2) Impact of Agent Number: For performance comparison, we first vary the number of the agents N from 3 to 29 while the other parameters are fixed as in Section V-B. In this situation, Fig. 11 shows the variation of the global performance of different methods under different agent numbers. As can be seen, the global performance, i.e., the sum of the overall individual expected successful offloading rate, can increase along with increasing the number of the agents in both the proposed DBR and the centralized PSO methods. Besides, our distributed method can achieve the performance very closed to that of the centralized PSO. The performance gap between the DBR and

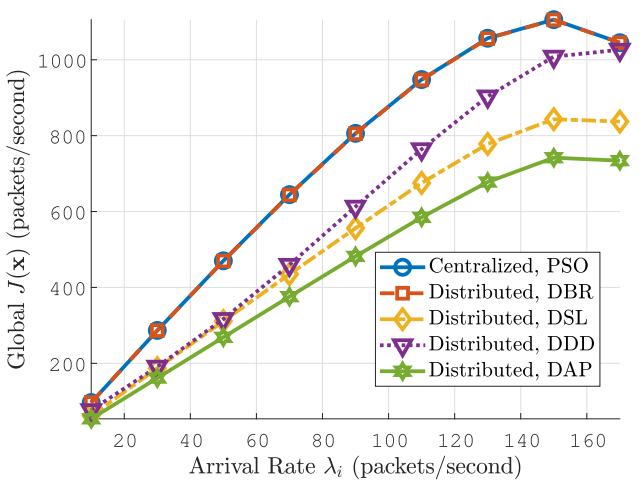

Fig. 12. Performance comparison under different task arrival rates.

the centralized PSO is about 0.0969% on average. On the other side, the other distributed methods including DSL, DDD and DAP perform worse than DBR. In particular, the performance of DDD and DAP slightly degrades when the agent number increases from 21 to 29. The main reason is that increasing the number of agents will potentially promote the intensity of channel contention and wireless interference since the channel resource is limited. The proposed DBR can coordinate the decision-making behaviors of the agents in a distributed manner and thus can achieve higher global performance, while the other distributed methods cannot deal with the increased interference well and results in a larger performance loss.

3) Impact of Task Arrival Rate: To show the impact of the task arrival rate λ_i on the system performance, we let λ_i increase from 10 packets/s to 170 packets/s with an increasing step of 20, and then fix the agent number $N = 10$. The other parameters are set the same as in Section V-B. The results of different methods are compared in Fig. 12, which demonstrates that increasing λ_i can increase the global objective function value $J(\mathbf{x})$. However, when λ_i exceeds 140 packets/s, the system performance will slightly degrade, since the amount of arriving packets that are lost due to the lack of enough queue space will increase. Fig. 12 also illustrates that our method can approximate the global optimal performance of the centralized PSO very well, while there exists a much greater performance gap between each of the other distributed methods, i.e., DSL, DDD, and DAP, and the centralized method. Specifically, DBR improves the global expected successful offloading rate by about 42.80% and 27.58% on average when compared to DSL and DDD, respectively.

4) Impact of Channel Number: In Fig. 13, we compare the proposed DBR with the other methods under different numbers of channels. In this situation, M is set to [1, 3, 5, 7, 9, 10], respectively, while let $N = 10$ and $\lambda_i = 100$ packets/s for all i. As can be seen, the number of available channels increases can promote the global system performance since the available spectrum resource increases. The wireless interference and channel contention can be relieved to improve the successful offloading rate. By comparison, the global performance of DBR is higher than that of DSL, DDD, and DAP by about

Authorized licensed use limited to: University of Waterloo. Downloaded on November 03,2023 at 14:29:46 UTC from IEEE Xplore. Restrictions apply.

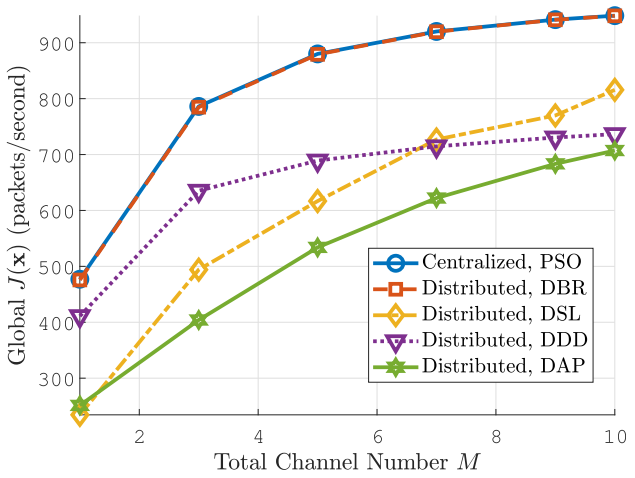

Fig. 13. Performance comparison under different channel numbers.

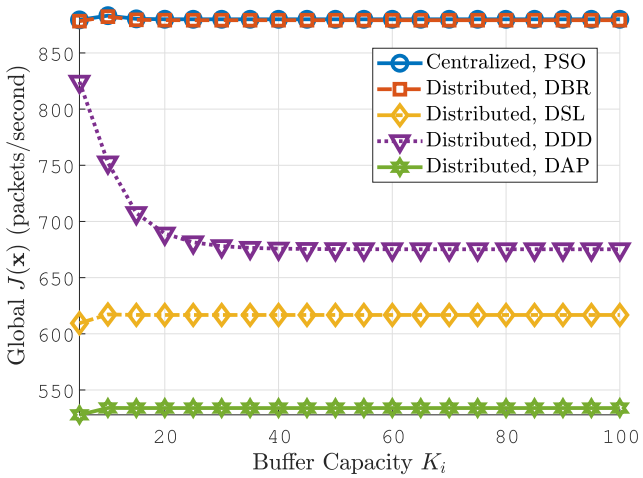

Fig. 15. Performance comparison under different buffer capacities.

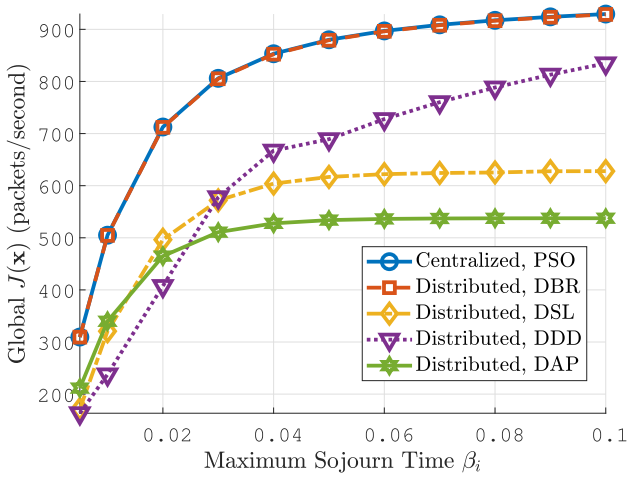

Fig. 14. Performance comparison under different sojourn thresholds in the queue.

44.87%, 25.51%, and 61.29% on average, respectively. The performance gap between DBR and the centralized PSO is about 0.1068% on average. This fact implies that our proposed method can perform better than DSL, DDD, and DAP in the offloading network no matter with sufficient or insufficient spectrum.

5) Impact of Maximum Sojourn Time: The different methods are also compared under different queueing dynamics situations where the sojourn time threshold β_i is varied from 5 ms to 100 ms in Fig. 14. The channel number is fixed at $M = 5$, and the other parameters are set according to Section V-C4. In Fig. 14, a larger value of β_i, usually implying a higher patience for waiting in the queue, will reduce the probability that an new incoming task is removed from the offloading queue and delivered for the local computation rather than for the remote edge computation. With different β_i, our method can outperform the other three distributed methods, DSL, DDD, and DAP, by the global performance improvement of about 48.81%, 41.33%, and 62.10% on average, respectively. Additionally, the performance gap between our method and the centralized PSO is about 0.0839% on average.

6) Impact of Buffer Capacity: In Fig. 15, we finally compare our method with the other methods under different buffer capacities K_i. The parameter K_i increases from 5 to 100, and we set $\beta_i = 50$ ms, while the other parameters are fixed as in Section V-C5. It is shown from Fig. 15 that the capacity of the buffer will have a slight impact on the offloading performance when $K_i > 20$. The main reason is that a larger buffer can cache more tasks to mitigate the transmission collision in the same channel. In fact, when the capacity of the queueing system approaches a sufficiently large level, e.g., being infinity, the system can approximately boil down to an M/M/1/∞ queueing model. In such a situation, the probability of the actual sojourn time exceeding the given threshold, $q_i(x_i)$, will be approximately a simple exponential function that only depends on the sojourn time threshold β_i, the offloading rate $\phi_i(x_i)$, and the task arrival rate λ_i, i.e., $q_i(x_i) \to \exp(-(\phi_i(x_i) - \lambda_i)\beta_i)$ with $K_i \to \infty$. Therefore, the global system performance will be approximately steady with increasing the buffer capacity K_i as shown in Fig. 15. Nevertheless, our proposed method can achieve comparable global optimality when compared to the centralized PSO. The performance gap between our method and the centralized PSO is about 0.0952% on average. Besides, our method can outperform the other three distributed methods, the global performance of which is higher than DSL, DDD, and DAP by about 42.64%, 27.83%, and 64.78% on average, respectively.

VI. Conclusion and Future Work

In this article, we have investigated the task offloading optimization of a multiagent interference-coupled and queue-aware network. We formulate the problem as a distributed offloading threshold optimization model to maximize the expected successful offloading rate of the agents under a set of bound constraints, which takes into consideration the coupled wireless interference in the physical layer and the queueing dynamics in the application layer. Stochastic models have been proposed to capture these effects. To address the distributed optimization problem, we have presented a game-theoretic analysis and then developed a DBR frame with the integration

Authorized licensed use limited to: University of Waterloo. Downloaded on November 03,2023 at 14:29:46 UTC from IEEE Xplore. Restrictions apply.

of an individual programming scheme. We have theoretically transformed the constrained nonlinear optimization of each agent into an ULO model such that we can propose the individual programming scheme, which enables the agents to locally and independently optimize their objective functions. Additionally, the agents can cooperate via decision feedback, such that they can approach the global system optimum. Finally, we conduct a series of simulation experiments to validate the effectiveness of our proposed method and demonstrate that our method can achieve comparable global performance compared to the centralized PSO optimization method and significantly outperform the other distributed methods including the distributed stochastic learning, the distributed dual-decomposition, and the aggressive policy-based methods. We are currently developing a CAV testbed which allows the implementation of the proposed distributed algorithm to enhance vehicular computation offloading via vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communications. As the future work, we expect to model and incorporate the energy-efficiency and computing-reliability optimization into the distributed offloading decision-making model, and extend the interference-coupled queue-aware edge computing network to heterogeneous systems.

REFERENCES

[1] M. Shafi *et al.*, "5G: A tutorial overview of standards, trials, challenges, deployment, and practice," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 6, pp. 1201–1221, Jun. 2017.

[2] X. Ge, Z. Li, and S. Li, "5G software defined vehicular networks," *IEEE Commun. Mag.*, vol. 55, no. 7, pp. 87–93, Jul. 2017.

[3] H. Peng, Q. Ye, and X. S. Shen, "SDN-based resource management for autonomous vehicular networks: A multi-access edge computing approach," *IEEE Wireless Commun.*, vol. 26, no. 4, pp. 156–162, Aug. 2019.

[4] N. Zhang, S. Zhang, P. Yang, O. Alhussein, W. Zhuang, and X. S. Shen, "Software defined space-air-ground integrated vehicular networks: Challenges and solutions," *IEEE Commun. Mag.*, vol. 55, no. 7, pp. 101–109, Jul. 2017.

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

[6] P. Porambage, J. Okwuibe, M. Liyanage, M. Ylianttila, and T. Taleb, "Survey on multi-access edge computing for Internet of Things realization," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 4, pp. 2961–2991, 4th Quart., 2018.

[7] Q.-V. Pham *et al.*, "A survey of multi-access edge computing in 5G and beyond: Fundamentals, technology integration, and state-of-the-art," 2019. [Online]. Available: arXiv:1906.08452.

[8] S. Jošilo and G. Dán, "Selfish decentralized computation offloading for mobile cloud computing in dense wireless networks," *IEEE Trans. Mobile Comput.*, vol. 18, no. 1, pp. 207–220, Jan. 2019.

[9] M. Chen and Y. Hao, "Task offloading for mobile edge computing in software defined ultra-dense network," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 3, pp. 587–597, Mar. 2018.

[10] H. A. Alameddine, S. Sharafeddine, S. Sebbah, S. Ayoubi, and C. Assi, "Dynamic task offloading and scheduling for low-latency IoT services in multi-access edge computing," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 3, pp. 668–682, Mar. 2019.

[11] S. Jeong, O. Simeone, and J. Kang, "Mobile edge computing via a UAV-mounted cloudlet: Optimization of bit allocation and path planning," *IEEE Trans. Veh. Technol.*, vol. 67, no. 3, pp. 2049–2063, Mar. 2018.

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

[13] C. Zhou *et al.*, "Delay-aware IoT task scheduling in space-air-ground integrated network," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2019, pp. 1–6.

[14] N. Cheng *et al.*, "Space/aerial-assisted computing offloading for IoT applications: A learning-based approach," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 5, pp. 1117–1129, May 2019.

[15] J. Zhou, D. Tian, Y. Wang, Z. Sheng, X. Duan, and V. C. M. Leung, "Reliability-optimal cooperative communication and computing in connected vehicle systems," *IEEE Trans. Mobile Comput.*, vol. 19, no. 5, pp. 1216–1232, May 2020.

[16] J. Zhou, D. Tian, Y. Wang, Z. Sheng, X. Duan, and V. C. M. Leung, "Reliability-oriented optimization of computation offloading for cooperative vehicle-infrastructure systems," *IEEE Signal Process. Lett.*, vol. 26, no. 1, pp. 104–108, Jan. 2019.

[17] J. Zhao, Q. Li, Y. Gong, and K. Zhang, "Computation offloading and resource allocation for cloud assisted mobile edge computing in vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 8, pp. 7944–7956, Aug. 2019.

[18] S. Barbarossa, S. Sardellitti, and P. D. Lorenzo, "Communicating while computing: Distributed mobile cloud computing over 5g heterogeneous networks," *IEEE Signal Process. Mag.*, vol. 31, no. 6, pp. 45–55, Nov. 2014.

[19] Y. Mao, J. Zhang, and K. B. Letaief, "Dynamic computation offloading for mobile-edge computing with energy harvesting devices," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 12, pp. 3590–3605, Dec. 2016.

[20] Y. Wu *et al.*, "Secrecy-driven resource management for vehicular computation offloading networks," *IEEE Netw.*, vol. 32, no. 3, pp. 84–91, May/Jun. 2018.

[21] S. Guo, J. Liu, Y. Yang, B. Xiao, and Z. Li, "Energy-efficient dynamic computation offloading and cooperative task scheduling in mobile cloud computing," *IEEE Trans. Mobile Comput.*, vol. 18, no. 2, pp. 319–333, Feb. 2019.

[22] H. Cao and J. Cai, "Distributed multiuser computation offloading for cloudlet-based mobile cloud computing: A game-theoretic machine learning approach," *IEEE Trans. Veh. Technol.*, vol. 67, no. 1, pp. 752–764, Jan. 2018.

[23] J. Zheng, Y. Cai, Y. Wu, and X. Shen, "Dynamic computation offloading for mobile cloud computing: A stochastic game-theoretic approach," *IEEE Trans. Mobile Comput.*, vol. 18, no. 4, pp. 771–786, Apr. 2019.

[24] P. R. de Waal, "Performance analysis and optimal control of an M/M/1/k queueing system with impatient customers," in *Messung, Modellierung und Bewertung von Rechensystemen*, U. Herzog and M. Paterok, Eds. Berlin, Germany: Springer, 1987, pp. 28–40.

[25] W. Zhang, Y. Wen, and D. O. Wu, "Collaborative task execution in mobile cloud computing under a stochastic wireless channel," *IEEE Trans. Wireless Commun.*, vol. 14, no. 1, pp. 81–93, Jan. 2015.

[26] N. Cheng *et al.*, "Air-ground integrated mobile edge networks: Architecture, challenges, and opportunities," *IEEE Commun. Mag.*, vol. 56, no. 8, pp. 26–32, Aug. 2018.

[27] Y. Zhou, N. Cheng, N. Lu, and X. S. Shen, "Multi-UAV-aided networks: Aerial-ground cooperative vehicular networking architecture," *IEEE Veh. Technol. Mag.*, vol. 10, no. 4, pp. 36–44, Dec. 2015.

[28] J. Heydari, V. Ganapathy, and M. Shah, "Dynamic task offloading in multi-agent mobile edge computing networks," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2019, pp. 1–6.

[29] Y. Liu, H. Yu, S. Xie, and Y. Zhang, "Deep reinforcement learning for offloading and resource allocation in vehicle edge computing and networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 11, pp. 11158–11168, Nov. 2019.

[30] J. Kwak, Y. Kim, J. Lee, and S. Chong, "DREAM: Dynamic resource and task allocation for energy minimization in mobile cloud systems," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 12, pp. 2510–2523, Dec. 2015.

[31] Z. Jiang and S. Mao, "Energy delay tradeoff in cloud offloading for multi-core mobile devices," *IEEE Access*, vol. 3, pp. 2306–2316, 2015.

[32] C. Liu, M. Bennis, and H. V. Poor, "Latency and reliability-aware task offloading and resource allocation for mobile edge computing," in *Proc. IEEE Globecom Workshops (GC Wkshps)*, Dec. 2017, pp. 1–7.

[33] Y. Mao, J. Zhang, S. H. Song, and K. B. Letaief, "Power-delay tradeoff in multi-user mobile-edge computing systems," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2016, pp. 1–6.

[34] K. Gai, M. Qiu, H. Zhao, L. Tao, and Z. Zong, "Dynamic energy-aware cloudlet-based mobile cloud computing model for green computing," *J. Netw. Comput. Appl.*, vol. 59, pp. 46–54, Jan. 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S108480451500123X

[35] K. Gai, M. Qiu, and H. Zhao, "Energy-aware task assignment for mobile cyber-enabled applications in heterogeneous cloud computing," *J. Parallel Distrib. Comput.*, vol. 111, pp. 126–135, Jan. 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0743731517302319

Authorized licensed use limited to: University of Waterloo. Downloaded on November 03,2023 at 14:29:46 UTC from IEEE Xplore. Restrictions apply.

[36] K. Gai, X. Qin, and L. Zhu, "An energy-aware high performance task allocation strategy in heterogeneous fog computing environments," *IEEE Trans. Comput.*, early access, May 11, 2020, doi: 10.1109/TC.2020.2993561.
[37] L. Cheng, B. E. Henty, D. D. Stancil, F. Bai, and P. Mudalige, "Mobile vehicle-to-vehicle narrow-band channel measurement and characterization of the 5.9 GHz dedicated short range communication (DSRC) frequency band," *IEEE J. Sel. Areas Commun.*, vol. 25, no. 8, pp. 1501–1516, Oct. 2007.
[38] R. Chen *et al.*, "Connectivity analysis for cooperative vehicular ad hoc networks under Nakagami fading channel," *IEEE Commun. Lett.*, vol. 18, no. 10, pp. 1787–1790, Oct. 2014.
[39] A. A. Khuwaja, Y. Chen, and G. Zheng, "Effect of user mobility and channel fading on the outage performance of UAV communications," *IEEE Wireless Commun. Lett.*, vol. 9, no. 3, pp. 367–370, Mar. 2020.
[40] W. Khawaja, I. Guvenc, and D. Matolak, "UWB channel sounding and modeling for UAV air-to-ground propagation channels," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2016, pp. 1–7.
[41] Z. Qiu, X. Chu, C. Calvo-Ramirez, C. Briso, and X. Yin, "Low altitude UAV air-to-ground channel measurement and modeling in semi-urban environments," *Wireless Commun. Mobile Comput.*, vol. 2017, Nov. 2017, Art. no. 1587412.
[42] M.-S. Alouini, A. Abdi, and M. Kaveh, "Sum of gamma variates and performance of wireless communication systems over Nakagami-fading channels," *IEEE Trans. Veh. Technol.*, vol. 50, no. 6, pp. 1471–1480, Nov. 2001.
[43] Y. R. Kwok and V. K. N. Lau, *The Mobile Radio Propagation Channel*. Piscataway, NJ, USA: Wiley-IEEE Press, 2007, pp. 1–17. [Online]. Available: https://ieeexplore.ieee.org/document/5236933
[44] F. M. Dekking, C. Kraaikamp, H. P. Lopuhaä, and L. E. Meester, *A Modern Introduction to Probability and Statistics: Understanding Why and How* (Springer Texts in Statistics). London, U.K.: Springer, 2005. [Online]. Available: https://books.google.com.hk/books?id=XLUMIlombgQC
[45] C. Cartis, N. I. M. Gould, and P. L. Toint, "On the complexity of steepest descent, Newton's and regularized Newton's methods for nonconvex unconstrained optimization problems," *SIAM J. Optim.*, vol. 20, no. 6, pp. 2833–2852, 2010. [Online]. Available: https://doi.org/10.1137/090774100
[46] Y. Ma, W. Zhang, W. Liu, and Q. Yang, "Fully distributed social welfare optimization with line flow constraint consideration," *IEEE Trans. Ind. Informat.*, vol. 11, no. 6, pp. 1532–1541, Dec. 2015.
[47] S. Wang, L. Du, J. Ye, and L. He, "Noncooperative social welfare optimization with resiliency against network anomaly," *IEEE Trans. Ind. Informat.*, vol. 16, no. 4, pp. 2403–2412, Apr. 2020.

Jianshan Zhou received the B.Sc., M.Sc., and Ph.D. degrees in traffic information engineering and control from Beihang University, Beijing, China, in 2013, 2016, and 2020, respectively.

From 2017 to 2018, he was a Visiting Research Fellow with the School of Informatics and Engineering, University of Sussex, Brighton, U.K. He is currently a Postdoctoral Research Fellow supported by the Zhuoyue Program of Beihang University. He has authored or coauthored of more than 20 international scientific publications. His research interests include the modeling and optimization of vehicular communication networks and air–ground cooperative networks, the analysis and control of CAVs, and intelligent transportation systems.

Dr. Zhou was a recipient of the First Prize in the Science and Technology Award from the China Intelligent Transportation Systems Association in 2017, the First Prize in the Innovation and Development Award from the China Association of Productivity Promotion Centers in 2020, the National Scholarships in 2017 and 2019, the Outstanding Top-Ten Ph.D. Candidate Prize from Beihang University in 2018, and the Outstanding China-SAE Doctoral Dissertation Award in 2020. He is or was the Technical Program Session Chair with the IEEE EDGE 2020 and the Youth Editorial Board Member of the Unmanned Systems Technology.

Daxin Tian (Senior Member, IEEE) received the Ph.D. degree in computer science from Jilin University, Changchun, China, in 2007.

He is currently a University Professor with the School of Transportation Science and Engineering, Beihang University, Beijing, China. His research focuses on mobile computing, intelligent transportation systems, vehicular ad hoc networks, and swarm intelligent.

Zhengguo Sheng (Senior Member, IEEE) received the B.Sc. degree from the University of Electronic Science and Technology of China, Chengdu, China, in 2006, and the M.S. and Ph.D. degrees from Imperial College London, London, U.K., in 2007 and 2011, respectively.

He is currently a Senior Lecturer with the University of Sussex, Brighton, U.K. He was with UBC, Vancouver, BC, Canada, as a Research Associate, and with Orange Labs, Santa Monica, CA, USA, as a Senior Researcher. He has more than 100 publications. His research interests cover IoT, vehicular communications, and cloud/edge computing.

Xuting Duan received the Ph.D. degree in traffic information engineering and control from Beihang University, Beijing, China, in 2017.

He is currently an Assistant Professor with the School of Transportation Science and Engineering, Beihang University. His current research interests are focused on vehicular ad hoc networks.

Xuemin (Sherman) Shen (Fellow, IEEE) received the Ph.D. degree in electrical engineering from Rutgers University, New Brunswick, NJ, USA, in 1990.

He is currently a University Professor with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada. His research focuses on network resource management, wireless network security, Internet of Things, 5G and beyond, and vehicular *ad hoc* and sensor networks.

Dr. Shen received the R.A. Fessenden Award in 2019 from IEEE, Canada, the Award of Merit from the Federation of Chinese Canadian Professionals (Ontario) in 2019, the James Evans Avant Garde Award in 2018 from the IEEE Vehicular Technology Society, the Joseph LoCicero Award in 2015 and Education Award in 2017 from the IEEE Communications Society, the Technical Recognition Award from Wireless Communications Technical Committee in 2019, the AHSN Technical Committee in 2013, the Excellent Graduate Supervision Award in 2006 from the University of Waterloo, and the Premier's Research Excellence Award in 2003 from the Province of Ontario, Canada. He served as the Technical Program Committee Chair/Co-Chair for IEEE Globecom'16, IEEE Infocom'14, IEEE VTC'10 Fall, and IEEE Globecom'07, and the Chair for the IEEE Communications Society Technical Committee on Wireless Communications. He is the elected IEEE Communications Society Vice President for Technical and Educational Activities, a Vice President for Publications, the Member-at-Large on the Board of Governors, the Chair of the Distinguished Lecturer Selection Committee, and a Member of IEEE ComSoc Fellow Selection Committee. He was/is the Editor-in-Chief of the IEEE INTERNET OF THINGS JOURNAL, IEEE NETWORK, *IET Communications*, and *Peer-to-Peer Networking and Applications*. He is a registered Professional Engineer of Ontario, Canada, a Fellow of the Engineering Institute of Canada, the Canadian Academy of Engineering, the Royal Society of Canada, a Foreign Member of the Chinese Academy of Engineering, and a Distinguished Lecturer of the IEEE Vehicular Technology Society and Communications Society.

Authorized licensed use limited to: University of Waterloo. Downloaded on November 03,2023 at 14:29:46 UTC from IEEE Xplore. Restrictions apply.