A Time- and Energy-Efficient Massive MIMO-NOMA MEC Offloading Technique: A Distributed ADMM Approach

Mohammad H. Alharbi*, Minhee Jun*, Hang Liu*
*The Catholic University of America, Washington, D.C., United States
{01alharbi, junm, liuh}@cua.edu

GLOBECOM 2022 - 2022 IEEE Global Communications Conference | 978-1-6654-3540-6/22/$31.00 ©2022 IEEE | DOI: 10.1109/GLOBECOM48099.2022.10001104

***Abstract*—Mobile edge computing (MEC) is an emerging platform that enables mobile devices to offload computation intensive tasks to the edge servers co-located with base stations (BSs) at the network edge for enhanced computation capabilities and low latency. This paper investigates the computation offloading problem in next-generation massive multiple-input multiple-output (M-MIMO) non-orthogonal multiple access (NOMA) MEC networks using a distributed alternating direction method of multipliers (ADMM) approach. Specifically, we develop a novel ADMM-based offloading algorithm to optimize latency and increase energy efficiency over next-generation mobile networks in a multiuser M-MIMO NOMA configuration. Simulation results demonstrate that the proposed offloading scheme significantly improves the system performance.**

***Index Terms*—MIMO, NOMA, Energy efficiency, latency, Distributed ADMM.**

I. Introduction

The increasing growth of smart, computation-heavy and latency-critical Internet of Things (IoT) applications impose great challenges for future wireless communication systems regarding offloading delay and power efficiency. Non-orthogonal multiple access (NOMA), multiple-input-multiple-output (MIMO), and recently mobile edge computing (MEC) [1] have been acknowledged as promising techniques to address these challenges [2]. The IoT devices, in MEC setups, can execute computation offloading by transferring their computationally demanding tasks to the base station (BS). The BS, being in close proximity of the IoT devices, then sends the results, processed by the edge servers, back to the devices with less latency and traffic load [1].

Quite a few works [3]–[6] study the effective design of joint communication and computation in multiuser MEC systems. For example, in [3], You et al. used the orthogonal frequency-division multiple access (OFDMA) based computation offloading to minimize the user's energy consumption [7]. In another work [4], Chen et al. used game theory and code-division multiple access (CDMA) based offloading for the energy efficiency among the users. A time-division multiple access (TDMA) based offloading for MEC [5] is proposed by Bi and Zhang where the computation offloading and local computing at the users are powered by wireless power transfer from the BS. Although the research is ongoing, generally these works cannot achieve the full capacity of the multiple access channel from multiple users to the BS, because they use orthogonal multiple access (OMA) for computation offloading (e.g., OFDMA and TDMA) or CDMA where interference is treated as noise. This inspires us to explore new multiple access schemes for computation offloading in this paper.

NOMA has been regarded as one of the key 5G cellular network strategies throughout the past few years [7]. Unlike conventional OMA, NOMA enables multiple users to communicate with the base station (BS) at the same time and frequency resources. By using advanced multi-user detection schemes such as the successive interference cancellation (SIC) at receivers, the NOMA-based communication system executes better spectral efficiency than OMA [8]. NOMA is expected to considerably enhance the performance of multi-user computation offloading for MEC systems as a result of its advantages over OMA. Study, e.g. [7], [8] have already demonstrated the benefits of applying NOMA to MEC; in [7], for example, the authors exploited NOMA for computation offloading for enabling multiple users to share the allotted spectrum. However, there is a lack of total response-time (i.e., latency) optimization, and performance comparison for the two schemes – NOMA and OMA on the response-time and energy consumption, which is the motivation of this paper and which can be crucial for network design. Moreover, investigation on the analytical and realistic computer emulation (i.e. simulation analysis) performance for a better understanding of the impact of NOMA on MEC is also lacking in the literature.

In this paper, we investigate the NOMA-based multi-user computation offloading technique for a multi-user MEC system, which consists of one multi-antenna BS and multiple users. To the best knowledge of our knowledge, the impact of M-MIMO NOMA on the average task-delay for the users and energy consumption in the task offloading for edge computing has not been analyzed yet; not even for M-MIMO only scenarios without NOMA. The contributions of this paper can be summarized as follows:

- A time- and energy-efficient MEC design is proposed by considering three baseline schemes: NOMA offloading, non-NOMA offloading and local computation only. This approach minimizes the response time (i.e., latency) at all users while ensuring the successful task execution at each user, by jointly optimizing the users' offloading decision, transmission powers and rates for offloading.

- Optimization of the offloading delay for NOMA-MEC by deriving the offloading delay minimization problem into a fractional programming using an iterative distributed ADMM algorithm.
- Analytical and simulation results in terms of average response time and number of tasks per second are presented and compared for the far- and near users. Moreover, the convergence rate of the proposed ADMM algorithm is presented to showcase the effect of network structure on the convergence rate.

The remainder of this paper is organized as follows. Section II presents the multiuser MEC system model and network model with multi-antenna NOMA-based computation offloading, and formulates the overall task-delay and energy minimization problem for the scope of this paper. Section III describes the problem formulation, discusses the optimization and decomposition, and proposes an efficient algorithm to obtain an optimal solution to the problem of latency reduction and energy optimization. Section IV presents the simulation details. Section V provides numerical results to evaluate the performance of our NOMA-based offloading technique by comparing it to other benchmark schemes. Finally, Section VI concludes this paper.

II. System model and network model

We assume here a transmission, in an uplink M-MIMO NOMA MEC network, for a BS that is connected to two users: a near and a far user. To that end, we describe an end-to-end system model with mathematical details along with problem formulation to assess the overall task-delay. Here, first, we define the effect of uploading in a M-MIMO system for clarifying the channel coefficient. Then we derive various parameter (e.g., data rate, power consumption, task delay etc.) details of the computation for near and far users. The details of the network model is discussed afterwards.

A. System model and network model

In the system model formulation, we consider that a set, U of mobile users are associated with a BS. The BS communicates with the users using the M-MIMO NOMA scheme. To reduce the system complexity, here we assume that NOMA SIC is only applied to a pair of users, $i, j \in U$; we call them the *near user* and the *far user*. The *near user* employs an energy harvesting technique to amplify and forward the signal to the *far user* assuming a power splitting ratio method. Fig. 1 illustrates the system model.

We assume the task arrival rate at user i (*near user*) is λ_i, and a task contains b_i bits of data. User i's task processing rate is μ_i . A proportion of $\epsilon_i\lambda_i$ tasks are offloaded to BS, and $(1-\epsilon_i)\lambda_i$ proportion of tasks are processed locally. In the proposed system model, we use a M/M/1 queuing system. For the tasks processed by user i locally, the average service delay (total time a task spends in the system including the time spent in waiting and executing) does not depend on scheduling discipline and is computed using Little's law [9] as in Eq. 1.

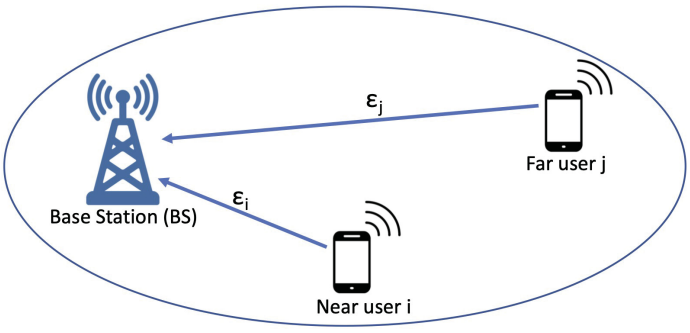


Fig. 1. System model with one base station and two users.

$$d_{p,i} = \frac{1}{\mu_i - (1-\varepsilon_i)\,\lambda_i} \tag{1}$$

Assuming the achieved data rate for user i to send data to the BS is r_i, the task transmission rate and the transmission queue delay can be obtained from Eq. 2 and Eq. 3 respectively. The service delay at the edge server is expressed as in Eq. 4.

$$v_i = \frac{r_i}{b_i} \tag{2}$$

$$d_{t,i} = \frac{1}{v_i - \varepsilon_i\lambda_i} \tag{3}$$

$$d_c = \frac{1}{\mu_c - \sum_{i\in U}\varepsilon_i\lambda_i} \tag{4}$$

The average task delay for user i can be written as in Eq. 5.

$$D_i = \frac{\lambda_i}{\sum_{i\in U}\lambda_i}\left[(1-\epsilon_i).d_{p,i} + \epsilon_i\left(d_{t,i} + d_c + d_{ri}\right)\right] \tag{5}$$

For the purpose of exposition, we focus our study on the computation offloading and execution phases, by assuming the duration d_{ri} for computation results for downloading to be constant. In addition, $\frac{\lambda_i}{\sum_{i\in U}\lambda_i}$ is used as a normalization factor due to the different task arrival rates of the users.

B. Time delay model

Here, we discuss how to control transmit power of the users and obtain the achieved offloading data rate. The mathematical model can be achieved using the following assumptions:

- BS is equipped with N antennas.
- Here, for M-MIMO we use a similar millimeter-wave (mmWave) model to [10] stated in Eq. 6.

$$h_i = \frac{\beta_i}{1+z_i^{\alpha}}[1 \quad e^{-j\pi\theta_i} \quad ...e^{-j\pi(N-1)\theta_i}]^T \tag{6}$$

here, z_i is distance between user i and BS.

Now, assuming two users i (near) and j (far), and considering a maximal ratio combining (MRC) beamformer used at the BS with a receiver beamforming matrix $w_i = h_i^H$, SINR for user i can be written as in Eq. 7.

$$SINR_i = \frac{|h_i^H h_i|^2 p_i}{|h_i^H h_j|^2 p_j + \sum_{g\in U\setminus\{i,j\}}|h_i^H h_g|^2 p_g + \sigma_i^2} \tag{7}$$

here, σ_i^2 is the noise power, $|h_i^H h_j|^2 p_j$ is the interference by j, and $\sum_{g\in U\setminus(i,j)} |h_i^H h_g|^2 p_g$ is the interference by the mobile users except i, j that transmit data using other M-MIMO beams. Ideally, when the number of antenna is large, the beams are narrow, then, $\sum_{g\in U\setminus\{i,j\}} |h_i^H h_g|^2 p_g \to 0$ [10].

For user j, MRC beamforming with $w_j = h_j^H$ is used to decode user j's signal which maximize SINR as in Eq. 8.

$$h_j = \frac{\beta_j}{1+z_j^{\alpha}}[1 \quad e^{-j\pi\theta_j} \quad ...e^{-j\pi(N-1)\theta_j}]^T \tag{8}$$

If the BS decode the message of user j without removing the interference of user i using SIC, the SINR of user j becomes as in Eq. 9.

$$SINR_j^{i\to j} = \frac{|h_j^H h_j|^2 p_j}{|h_j^H h_i|^2 p_j + \sum_{g\in U\setminus\{i,j\}} |h_i^H h_j|^2 p_g + \sigma_j^2} \tag{9}$$

If the BS decodes the message from user i and then use SIC to cancel user i's signal to decode user j's signal, the SINR of j without the interference of i is as in Eq. 10. The data rate r_j for user j can be expressed as in Eq. 11.

$$SINR_j = \frac{|h_j^H h_j|^2 p_j}{\sum_{g\in U\setminus\{i,j\}} |h_j^H h_g|^2 p_g + \sigma_j^2} \tag{10}$$

$$r_j = B\log_2[1+SINR_j] \tag{11}$$

We use the user selection criterion as stated in Eq. 12.

$$i = \underset{g\in U\setminus i}{\operatorname{argmax}} \left\{ |h_1^H h_j|^2, ...|h_g^H h_j|^2, ...|h_{|U|}^H h_j|^2 \right\} \tag{12}$$

For user j, due to SIC, the interference of user i is cancelled, thus its achieved data rate is the same as that with no user i's interference. For user i, if there is no interference from user j, its SINR and data rate can be expressed as in Eq. 13 and Eq. 14 respectively.

$$SINR_{i}' = \frac{|h_i^H h_j|^2 p_i}{\sum_{g\in M\setminus\{i,j\}} |h_i^H h_g|^2 p_g + \sigma_i^2} \tag{13}$$

$$r_{i}' = B\log_2[1+SINR_i'] \tag{14}$$

In the next part, we will describe our power consumption model; there, as described in Eq. (2 - 4), the delay depends on the data rate r_i as in Eq. 15,

$$r_i \ = B\log_2[1+SINR_i] \tag{15}$$

C. Power consumption model

The power consumption for local task processing is derived by Eq. 16.

$$E_{p,i} = (1-\varepsilon_i)\,\lambda_i e_{p,i} \tag{16}$$

The transmission power consumption to upload the tasks and the total power consumption rate are expressed as in Eq. 17 and Eq. 18 respectively.

$$E_{t,i} = p_i \frac{\epsilon_i \lambda_i b_i}{r_i} \tag{17}$$

$$E_i = E_{p,i} + E_{t,i} = (1-\varepsilon_i)\,\lambda_i e_{p,i} + p_i \frac{\epsilon_i \lambda_i b_i}{r_i} \tag{18}$$

Next, we describe the problem formulation, optimization for total delay at far and near users, and the decomposition details.

III. Problem formulation, Optimization, and decomposition

A. Problem formulation and optimization

The NOMA technique offers nodes that have enhanced links with a high signal-to-noise ratio (SNR). This, in turn, offers enhanced signal quality and data rates. Among the several evident limitations and challenges, the increase of the total offloading delay in uplink scheme of the M-MIMO wireless networks is substantial. Furthermore, in specific applications, offloading delay is considered as major component in wireless network, where minimizing the offloading delay is more challenging than energy consumption. In this work, the minimization of the offloading delay for NOMA-MEC is applied by deriving the offloading delay minimization problem into a fractional programming using an iterative algorithm. The problem (P_{LE}) thus can be formulated as in Eq. 19.

$$\begin{aligned} P_{LE}: &\underset{\epsilon_i, p_i}{\operatorname{argmin}} \left\{ \sum_{i\in U} D_i \right\} = \\ &\underset{\epsilon_i, p_i}{\operatorname{argmin}} \left\{ \sum_{i\in U} \frac{\lambda_i}{\sum_{i\in U}\lambda_i} \left[(1-\epsilon_i).d_{p,i} + \epsilon_i\,(d_{t,i} + d_c + d_{ri})\right] \right\} \end{aligned} \tag{19}$$

The objective of optimization is to reduce the total latency including the local data processing and to minimize the achievable sum-rate consumption at the users. In order to obtain insight into the performance of the proposed M-MIMO NOMA offloading optimization scheme, here, we focus on the special case that two users offload their tasks to an edge server with M-MIMO NOMA transmission technique. After defining and formulating the problem (P_{LE}), the optimization problem is formulated, in which the proportion of the tasks ϵ_i to be offloaded to the edge server as well as the transmit power p_i used for offloading by each of the users are determined to minimize the total average service delay under the power consumption constraints as stated in Eq. 20.

$$E_i = E_{p,i} + E_{t,i} = (1-\epsilon_i)\lambda_i e_{p,i} + p_i \frac{\epsilon_i \lambda_i b_i}{r_i} \le E_{max} \tag{20a}$$

$$0 \le p_i \le P_{max} \tag{20b}$$

$$0 \le \epsilon_i \le 1 \tag{20c}$$

To address the power consumption constraints, stated in Eq. 20(b), we propose to use a distributed ADMM approach, stated in the next section, which can effectively address the constraint optimization problem. It is worth to mention that the decomposition process is part of the distributed ADMM approach discussed in the next section.

B. Decomposition

According to the problem formulation expression in Eq. 19, it can be easily observed that the objective function is composed of two parts ϵ_i and p_i. Note that p_i is shown in the constraint in Eq. 20. Since ADMM can only be utilized to solve the optimization problem without constraints, our problem cannot be directly solved by ADMM. For that, decomposition of Eq. 19 should be applied first; the users

maximize their offloaded computation tasks, while on the other hand, the BS aims to minimize its energy consumption.

Here, we introduce $\mathbb{T}_i$ auxiliary non-negative variable with size N indicator function such as $\mathcal{T} = \{\mathcal{T}_1, \mathcal{T}_2, ..., \mathcal{T}_n\}$ where $\mathcal{T}_N \in \mathcal{T}$, $\mathbf{K} = \{K_1, K_2, ..., Kn\}$ where $K_N \in \mathbf{K}$, in order to convert Eq. 19 to an unconstrained one. As shown in Eq. 20a, the constraints contain of two parts: $E_{p,i}$ and $E_{t,i}$, while in Eq. 20b, it is p_i. We introduce variables $\mathbf{x} = \{x_1, x_2, ..., x_n\}$ and $\mathbf{y} = \{y_1, y_2, ..., y_n\}$ as copies of $\mathcal{T}$ and $\mathbf{K}$ respectively; here x_n and y_n are as in Eq. 21.

$$\begin{cases} x_n = \mathcal{T}_n, & \forall n \in N \\ y_n = K_n, & \forall n \in N \end{cases} \tag{21}$$

The relation for x and y can be expressed as in Eq. 22.

$$\Lambda = \begin{cases} 0 \leq x_n \leq E_{max}, & \forall n \in N \\ 0 \leq y_n \leq P_{max}, & \forall n \in N \end{cases} \tag{22}$$

From Eq. 22, we can define the total transmission latency of P_{LE} as in Eq. 23.

$$U_n(\mathbf{x}, \mathbf{y}) = \begin{cases} P_{LE}(\mathbf{x}, \mathbf{y}) & \mathbf{x}, \mathbf{y} \in \lambda \\ \infty & else \end{cases} \tag{23}$$

An equivalent formulation of P_{LE} can be expressed as in Eq. 24.

$$P_{LE^1} : \underset{\mathbf{x},\mathbf{y}}{\operatorname{argmin}} \sum_{n=1}^{N} U_n(\mathbf{x}, \mathbf{y}) \tag{24}$$

Eq. 24 is a convex optimization problem. The augmented Lagrangian function of P_{LE^1} is given as in Eq. 25.

$$\begin{aligned} &L(\{\mathbf{x}, \mathbf{y}\}, \{\mathcal{T}, \mathbf{K}\}, \{\alpha, \beta\}) \\ &\quad = \sum_{n=1}^{N} U_n(\mathbf{x}, \mathbf{y}) \\ &\quad + \sum_{n=1}^{N} \alpha_n(x_n - \mathcal{T}_n) + \sum_{n=1}^{N} \beta_n(y_n - K_n) \\ &\quad + \frac{v}{2} \sum_{n=1}^{N} (x_n - \mathcal{T}_n)^2 + \frac{v}{2} \sum_{n=1}^{N} (y_n - K_n)^2 \end{aligned} \tag{25}$$

here, $\alpha = \{\alpha_1, \alpha_2, ..., \alpha_n\}$ and $\beta = \{\beta_1, \beta_2, ..., \beta_n\}$ are the Lagrangian multipliers, and v is a penalty parameter [11] related to the convergence speed of the ADMM algorithm. To that end, the related dual function can be written as in Eq. 26 while the dual problem can be expressed as in Eq. 27.

$$d(\alpha, \beta) = \underset{\{\mathbf{x},\mathbf{y}\},\{\mathcal{T},\mathbf{K}\}}{\operatorname{argmin}} L(\{\mathbf{x}, \mathbf{y}\}, \{\mathcal{T}, \mathbf{K}\}, \{\alpha, \beta\}) \tag{26}$$

$$\max_{\alpha,\beta} \; d(\alpha, \beta) \tag{27}$$

C. Purpose of the ADMM offloading algorithm

We use ADMM to solve the dual problem stated in Eq. 27; here we denote the value γth iteration as $\{\mathbf{x}^\gamma, \mathbf{y}^\gamma\}, \{\mathcal{T}^\gamma, \mathbf{K}^\gamma\}, \{\alpha^\gamma, \beta^\gamma\}$. The sequential iterative optimization steps are given as follows.

Step-1: Given $\{\mathcal{T}^\gamma, \mathbf{K}^\gamma\}, \{\alpha^\gamma, \beta^\gamma\}$, we update $\{\mathbf{x}^\gamma, \mathbf{y}^\gamma\}$ by maximizing L where,

$$\{\mathbf{x}^{\gamma+1}, \mathbf{y}^{\gamma+1}\} = \underset{\{\mathbf{x_n},\mathbf{y_n}\}}{\operatorname{argmin}} L(\{\mathbf{x}, \mathbf{y}\}, \{\mathcal{T}^\gamma, \mathbf{K}^\gamma\}, \{\alpha^\gamma, \beta^\gamma\}) \tag{28}$$

Now, Eq. 28 can be decomposed into N parallel sub problems, and each sub-problem solves the following Eq. 29.

$$\begin{aligned} \{\mathbf{x}^{\gamma+1}, \mathbf{y}^{\gamma+1}\} = \underset{\{\mathbf{x_n},\mathbf{y_n}\}}{\operatorname{argmin}} \{&U_n(\mathbf{x}, \mathbf{y}) \\ &+ \alpha_n^\gamma(\mathbf{x}_n - \mathcal{T}_n^\gamma) + \beta_n^\gamma(\mathbf{y}_n - \mathbf{K}_n^\gamma) \\ &+ \frac{v}{2}(\mathbf{x}_n - \mathcal{T}_n^\gamma)^2 + \frac{v}{2}(\mathbf{y}_n - \mathbf{K}_n^\gamma)^2\} \end{aligned} \tag{29}$$

It is worth noting that Eq. 29 is an unconstrained convex optimization problem, where the optimal solution can be obtained by the gradient descent method. After solving N parallel sub-problems, we update $\{x^\gamma, y^\gamma\}$ with $\{x^{\gamma+1}, y^{\gamma+1}\}$.

Algorithm 1 Distributed Solution using ADMM Algorithm

1: **procedure** Initialize MEC
2: $Itr \leftarrow$ Number of iterations
3: Solve Eq. 29 to obtain $\{x^{\gamma+1}, y^{\gamma+1}\}$
4: Set $\{x^\gamma, y^\gamma\} \leftarrow \{x^{\gamma+1}, y^{\gamma+1}\}$
5: Update $\{\mathcal{T}, K\}$ using Eq. 31
6: Update $\{\alpha, \beta\}$ using Eq. 32
7: Return to step 3 until convergence
8: **END**

Step-2: Given $\{x^{\gamma+1}, y^{\gamma+1}\}$, we minimize L with respect to $\{\mathcal{T}, \mathbf{K}\}$ as in Eq. 30.

$$\begin{aligned} \left\{\mathcal{T}^{\gamma+1}, \mathbf{K}^{\gamma+1}\right\} = \underset{\{\mathcal{T},\mathbf{K}\}}{\operatorname{argmin}} \Bigg\{ &\sum_{n=1}^{N} \alpha_n^\gamma(x_n^\gamma - \mathcal{T}_n) \\ &+ \sum_{n=1}^{N} \beta_n^\gamma(y_n^\gamma - K_n) + \frac{v}{2} \sum_{n=1}^{N} (x_n^\gamma - \mathcal{T}_n)^2 \\ &+ \frac{v}{2} \sum_{n=1}^{N} (y_n^\gamma - K_n)^2 \Bigg\} \end{aligned} \tag{30}$$

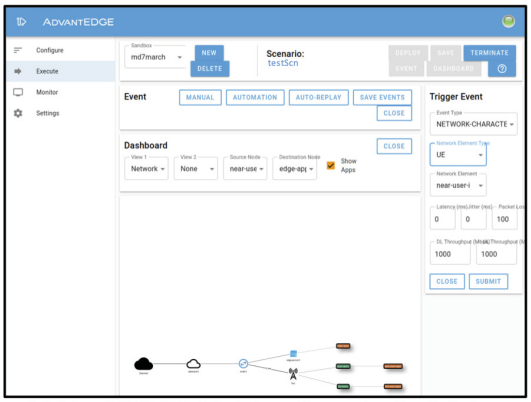
Since Eq. 30 is a unconstrained quadratic convex problem, we derive a low complexity algorithm by simply setting the gradients of $\mathcal{T}$ and K to zeros, and the iteration results are given as in Eq. 31.

$$\begin{aligned} \mathcal{T}_n^{\gamma+1} &= x_n^{\gamma+1} + \frac{\alpha_n^\gamma}{v}, \quad \forall n \in N \\ K_n^{\gamma+1} &= y_n^{\gamma+1} + \frac{\beta_n^\gamma}{v}, \quad \forall n \in N \end{aligned} \tag{31}$$

Step-3: Given $\{x^{\gamma+1}, y^{\gamma+1}\}$ and $\{\mathcal{T}^{\gamma+1}, K^{\gamma+1}\}$, we we optimize Eq. 27 with respect to $\{\alpha^\gamma, \beta^\gamma\}$, which is achieved by updating $\{\alpha, \beta\}$ as in Eq. 32.

$$\begin{aligned} \alpha_n^{\gamma+1} &= \alpha_n^\gamma + v\left(x_n^{\gamma+1} - \mathcal{T}_n^{\gamma+1}\right), \quad \forall n \in N \\ \beta_n^{\gamma+1} &= \beta_n^\gamma + v\left(y_n^{\gamma+1} - K_n^{\gamma+1}\right), \quad \forall n \in N \end{aligned} \tag{32}$$

The above three steps are conducted alternatively until convergence. As depicted the algorithm 1, the distributed nature of



(a) AdvantEDGE GUI: full window with sample network design.

(b) Sample network with BS, near- and far users, edge server, and their app settings.

Fig. 2. Sample AdvantEDGE setup with BS, two users, edge server and their corresponding apps.

this algorithm allows for a very efficient parallel implementation which can be implemented in parallel mobile devices.

IV. Simulation Setup

After getting the initial results using the formulation stated in above sections, we port these results, e.g., number of tasks sent to the BS, task transmission error rates, transmit power etc., on each node to AdvantEDGE [12] for further simulation. AdvantEDGE is a mobile edge emulation platform that allows the connection of real cloudlet and UE applications so that simulation can capture the impact of network design on application performance [12]; this makes it a very useful platform for edge network simulation. As it is not feasible to implement, connect and deploy edge node servers in a real mobile network infrastructure, our work relies instead on the realistic emulations using AdvantEDGE. We use it to emulate our mobile wireless network with one BS and two users and then compare the results obtained from this platform with our analysis results. Fig. 2 shows our sample AdvantEDGE setup with one BS, two users (near and far users), an edge server, and their corresponding apps. Detailed instructions on designing such a network can be found in [12].

V. Analysis and Simulation Results

This section presents the performance of NOMA-based offloading scheme in MEC for two users – a near user (relatively near to the BS) and a far user (relatively far than the near user from the BS). Below we present analysis and simulation results which we achieved by using the formulation and simulation setup as described in previous sections. The distances between the BS, and the two users are 200 and 400 meters. Each task contains $10,000$ bits and the path loss exponent is set to 3.5. The system bandwidth for computation offloading is set as $2\,\mathrm{MHz}$ and the noise power at the BS

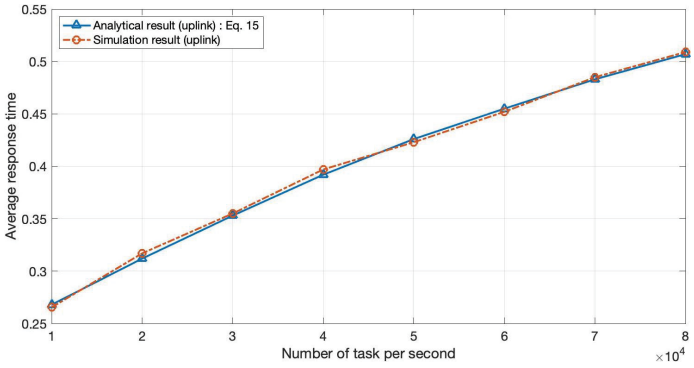


Fig. 3. Avgerage response time vs. number of tasks per second at each user: analysis and simulation results.

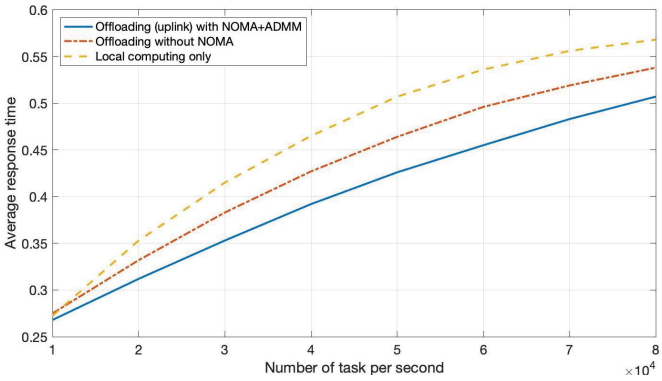


Fig. 4. Average response time vs. number of task per second at each user.

receiver is set to $-174 + 10log_{10}(BW); BW = bandwidth$. Except for Fig. 7, we use $N = 64$ no. of antennas for all of the other figures; in Fig. 7, we vary N from 32 to 128.

A. Verification of ADMM-based NOMA offloading method

Fig. 3 shows the analytical and simulation results for the average response time versus number of tasks per second at each user. Here the simulation results are obtained using AdvantEdge [12]. Here, we can observe the curve for the analytical result in Eq. 15 match with the simulation result's curve, which verifies the accuracy of our analysis.

Fig. 4 shows that the average response time increases as the as *number of task per second* increases. Here, we use three base schemes – i) offloading (uplink) with NOMA+ADMM, ii) offloading without NOMA, and iii) local computing only. The NOMA+ADMM offloading scheme is observed to achieve the smallest response time among all the schemes. Compared with the non-NOMA-based offloading scheme, significantly less response time is required by the NOMA+ADMM based one, especially when *number of task per second* becomes large. It is also observed that the non-NOMA-based scheme out-performs the local-computing-only scheme.

Fig. 5 demonstrates that the achievable rate becomes saturated after a transmit power of $10dBm$ for the far user which is a typical characteristic for all NOMA networks. This saturation is caused due to the interference experienced by the far user.

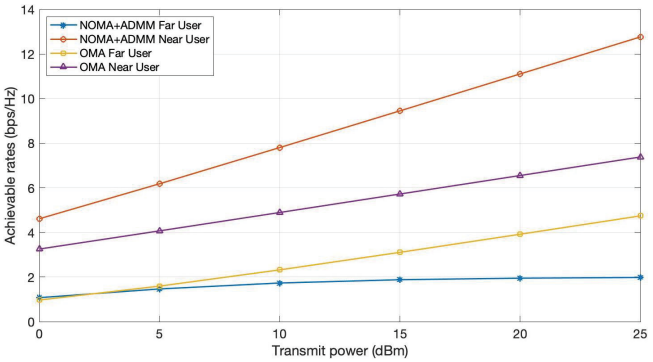


Fig. 5. Power consumption vs. achievable rates for far- and near users.

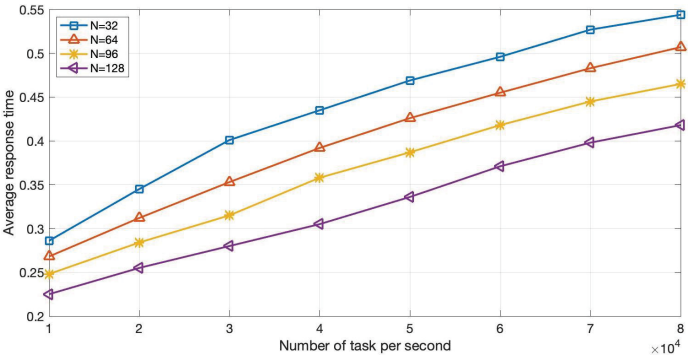


Fig. 7. Average response time vs. number of antennas at each user.

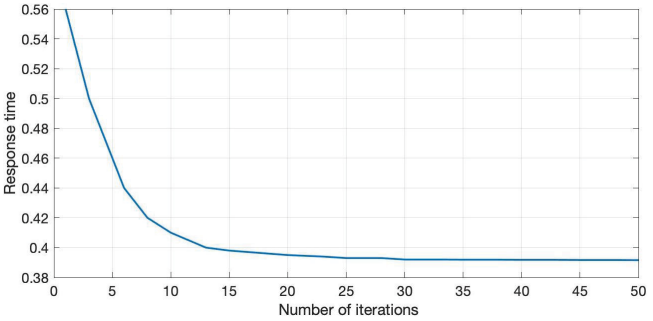


Fig. 6. Convergence rate of the distributed ADMM algorithm.

It would not occur if the required data rate of the far user is less than the saturation limit. OMA does not suffer from such problems, due to its simultaneous transmission capability.

B. Characteristics of ADMM-based NOMA offloading method

Fig. 6 shows the convergence performance of our distributed ADMM algorithm 1 for our proposed network. Here, we see that our proposed algorithm can converge to the global optimal solution within the first few iterations (less than 30 iterations).

Fig. 7 shows a depiction of the average response time of our NOMA-based network by varying the number of antennas; here we use $N = [32, 64, 96, 128]$ antennas respectively. Here, we observe the behavior of the network in terms of the average response time on how it varies as the number of antennas in the network changes. The result shows that, when the number of antennas becomes larger, there will be higher antenna gains and consequently, the achievable data rate would also be higher. Hence, the average response time would be lower.

VI. Conclusion and Future Works

This paper presents a time- and energy-efficient MEC design, by using a distributed ADMM technique, which considers three baseline schemes: NOMA offloading, non-NOMA offloading and local computation only. This design minimizes the response time at all users while ensuring the successful task execution at each user, by jointly optimizing the users' offloading decision, transmission powers and rates for offloading. Here, the offloading delay for NOMA-MEC is optimized by deriving the offloading delay minimization problem into a fractional programming using an iterative distributed ADMM algorithm. The proposed system- and network model consists of a near and a far user to the BS where the near user employs an energy harvesting technique to amplify and forward the signal to the far user assuming a power splitting ratio method. Our numerical analysis results demonstrate that the proposed technique efficiently minimizes offloading delay and optimize the energy for an uplink NOMA-based MEC network. Moreover, our simulation results, and the rate of convergence of the proposed algorithm further prove its efficiency. In future, we would like to incorporate multiple near users in our design and analyse its impact on the effectiveness and performance.

References

[1] Pavel Mach and Zdenek Becvar. Mobile edge computing: A survey on architecture and computation offloading. *IEEE Communications Surveys & Tutorials*, 19(3):1628–1656, 2017.

[2] Hong Wang, Chen Liu, Zheng Shi, Yaru Fu, and Rongfang Song. On the design of high power effic. uplink mimo-noma systems: A stbc and joint detec. perspective. *IEEE Tran. on Vehi. Tech.*, 70(1):627–638, 2021.

[3] Changsheng You, Kaibin Huang, Hyukjin Chae, and Byoung-Hoon Kim. Energy-efficient resource allocation for mobile-edge computation offloading. *IEEE Tran. on Wireless Comm.*, 16(3):1397–1411, 2016.

[4] Xu Chen, Lei Jiao, Wenzhong Li, and Xiaoming Fu. Efficient multi-user computation offloading for mobile-edge cloud computing. *IEEE/ACM Transactions on Networking*, 24(5):2795–2808, 2015.

[5] Suzhi Bi and Ying Jun Zhang. Computation rate maximization for wireless powered mobile-edge computing with binary computation offloading. *IEEE Tran. on Wireless Comm.*, 17(6):4177–4190, 2018.

[6] Dixiao Wu, Feng Wang, Xiaowen Cao, and Jie Xu. Wireless powered user cooperative computation in mobile edge computing systems. In *2018 IEEE Globecom Workshops (GC Wkshps)*, pages 1–7. IEEE, 2018.

[7] Feng Wang, Jie Xu, and Zhiguo Ding. Multi-antenna noma for computation offloading in multiuser mobile edge computing systems. *IEEE Transactions on Communications*, 67(3):2450–2463, 2019.

[8] Admoon Andrawes, Rosdiadee Nordin, and Nor Fadzilah Abdullah. Energy-efficient downlink for non-orthogonal multiple access with swipt under constrained throughput. *Energies*, 13(1), 2020.

[9] Sanjay K Bose. *An introduction to queueing systems*. Springer Science & Business Media, 2013.

[10] Zhiguo Ding, Linglong Dai, Robert Schober, and H. Vincent Poor. Noma meets finite resolution analog beamforming in massive mimo and millimeter-wave networks. *IEEE Commu. Let.*, 21(8):1879–1882, 2017.

[11] Stephen Boyd, Neal Parikh, and Eric Chu. *Distributed optimization and statistical learning via the alternating direction method of multipliers*. Now Publishers Inc, 2011.

[12] Michel Roy, Kevin Di Lallo, and Robert Gazda. Advantedge: A mobile edge emulation platform (meep). https://github.com/InterDigitalInc/AdvantEDGE, 2022. [Online; accessed 20-May-2022].