Joint Resource Allocation for Latency-Constrained Dynamic Computation Offloading with MEC

Conference Paper · April 2019

DOI: 10.1109/WCNCW.2019.8902904

4 authors:

Mattia Merluzzi
Sapienza University of Rome
12 PUBLICATIONS **83** CITATIONS
SEE PROFILE

Paolo Di Lorenzo
Università degli Studi di Perugia
65 PUBLICATIONS **1,669** CITATIONS
SEE PROFILE

S. Barbarossa
Sapienza University of Rome
279 PUBLICATIONS **10,582** CITATIONS
SEE PROFILE

Valerio Frascolla
Intel Deutschland GmbH
58 PUBLICATIONS **420** CITATIONS
SEE PROFILE

Joint Resource Allocation for Latency-Constrained Dynamic Computation Offloading with MEC

Mattia Merluzzi[1], Paolo Di Lorenzo[1], Sergio Barbarossa[1], and Valerio Frascolla[2]
[1]Sapienza Univ. of Rome, DIET Dept, Via Eudossiana 18, 00184, Rome, Italy
[2]Intel Deutschland GmbH, Am Campeon 10-12, 85579 Neubiberg Germany
e-mail: {mattia.merluzzi,paolo.dilorenzo, sergio.barbarossa}@uniroma1.it, valerio.frascolla@intel.com

***Abstract*—In this paper, we address the problem of dynamic computation offloading with Multi-Access Edge Computing (MEC), where new requests for computations are continuously generated at each user equipment (UE), and are handled through dynamic queue systems. Building on stochastic optimization tools, we provide a dynamic algorithm that jointly optimize radio (i.e., power, bandwidth) and computation (i.e., CPU cycles) resources, while guaranteeing a target performance in terms of average latency and out of service probability, i.e., the probability that the (sum of) computation queues exceeds a predefined value. The method requires the solution of a convex optimization problem at each time slot, and does not need any apriori knowledge of channel and task arrival distributions. Finally, numerical results corroborate the potential benefits of our strategy.**

***Index Terms*—Computation offloading, Mobile Edge computing, 5G networks, queues, stochastic optimization.**

I. Introduction

Future 5G networks are foreseen to radically evolve the concept of mobile systems, enabling a plethora of new services for a wide range of sectors (verticals) with very different requirements. The growth of the mobile data traffic is foreseen to reach 100 exabytes by 2023 [1], and 5G is expected to cover 20 percent of this traffic. Differently from previous generations of mobile systems [2], the aim of 5G is not a mere set of enhancements, but a real revolution towards new applications and services such as Industry 4.0, virtual and augmented reality, automated driving, etc. This requires a flexible design of the network, which builds on some key technology thrusts, like network function virtualization, millimeterwave (mmWave) communications [3], Multi-Access Edge Computing [4], etc. MEC aims at bringing cloud computing functionalitites at the edge of the network, typically in the Access Points (AP) or at an aggregation point of the core network. The combination mmWave communications and MEC, which is the main idea of the H2020 EU/JP funded project 5G-Miedge [5], provides high capacity access (mmWave) to nearby Mobile Edge Hosts (MEH), enabling low latency services with possibly high data consuming applications. In particular, one of the services provided by MEC is the offloading of resource hungry applications from tiny devices (smartphones, sensors, etc.), to nearby MEHs, with the purpose of saving energy or allowing resource poor devices to run sophisticated applications within low latency constraints. However, differently from cloud computing, computational resources in MEC networks are limited and need to be managed properly in order to achieve a good Quality of Service (QoS) for the end users. In particular, since the end-to-end delay comprises a communication time and a computation time, in a user/application centric architecture it is important to manage these resources jointly [6].

This work was supported by H2020 EUJ Project 5G-MiEdge, Nr. 723171.

Related works. Resource allocation for computation offloading with MEC is a challenging problem, and represented a pivotal topic in several research communities during last years [6]–[15]. In [16], communication, computation and storage resources are optimized jointly in order to make an effective usage of the limited (with respect to the cloud) resources available at the MEC hosts. In [17], a recent survey related to MEC and computation offloading is provided. A possible classification for computation offloading separates between *static* and *dynamic* strategies. In static strategies, users request to run an application, having a well defined computational demand, within a very short Time To Leave (TTL); whereas, in dynamic scenarios, the application continuously generates data to be processed, possibly without knowing the statistics of the the data. The dynamic formulation of the problem is also effective to handle users' mobility, which is a crucial problem in mobile networks. Concerning the static formulation, work [6] introduces the concept of joint optimization of radio and computation resources for computation offloading. In recent works [7], [8], overprovisioning of radio and computation resource and multi-link communications are investigated as a way to counteract the intermittent behavior of mmwave links due to blockages. In particular, in [7], statistical independent blocking event are considered, while in [8] the investigation is extended to the statistical dependent events due to large obstacles. In [18], a novel analysis of block erasure coding for multi-link mmmWave communications is provided. In [14], the joint assignment of mobile users to mmWave AP and MEH and the joint allocation of radio and computation resources is studied. In particular, the problem of the assignment is handled through two different algorithms: one based on a penalized Successive Convex Approximation strategy, and one based on a many-to-one matching game. In [19], the authors propose a strategy to minimize the energy consumption, in TDMA and OFDMA systems, while in [20] a joint optimization of offloading decision and allocation of radio and computation resources is investigated. The dynamic formulation is investigated in [11], where the authors aim to minimize the long-term average power consumption under constraints on the

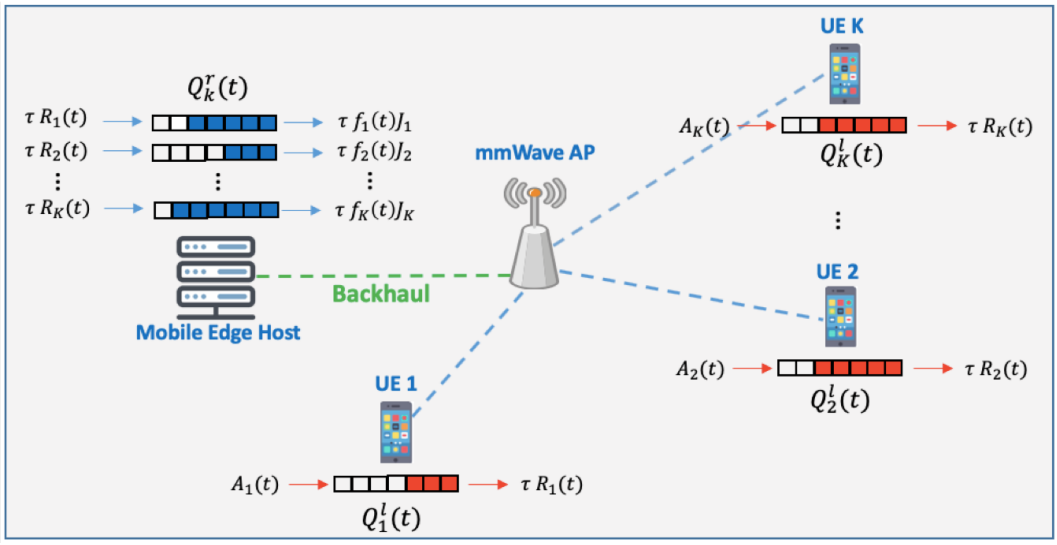

Fig. 1: Network scenario.

mean rate stability of the computation queues with a single MEH. In [10], the authors introduce energy harvesting devices formulating the problem as the minimization of an offloading cost incorporating the delay and the task dropping rate. In [12] the authors consider a fog-enabled D2D scenario and propose a strategy to associate mobile devices and offload tasks among each other. The authors of [13] address the problem of user assignment, with the aim of minimizing the average delay under energy constraints, while introducing a penalty function that discourages frequent handovers, and using Multi-armed bandit to learn the optimal penalty parameter.

All the aforementioned works do not address the problem of dynamic computation offloading while keeping the computation queues under a certain threshold in order to limit the service delay. To the best of our knowledge, there are only a few works that deal with this problem. In fact, latency-constrained dynamic computation offloading was first addressed in [15], where the authors introduce a probabilistic constraint on the computation queues, written as a bound on the probability of exceeding a certain value, handling it with extreme value theory. A similar approach was also used in the context of vehicular communications in [21]. Finally, [22] extends [15] by considering a scenario with multiple APs and MEHs, and introducing a UE's assignment strategy based on matching theory.

Contribution. In this paper, we propose a novel algorithm for dynamic computation offloading, aimed at minimizing the long-term average power consumption under an average latency constraint and a bound on the out-of-service probability, defined as the probability that the overall service time (including communication and computation times) exceed a certain value. We consider the scenario where UEs offload all their computations to a MEH and there is no concurrent (UE/MEH) computation, to avoid continuous back and forth exchange of program status from UE and MEH. We impose constraints on the sum of the local queues (data to be transmitted from the UEs) and the remote queues at the MEH (computations to be performed). This sum represents a proper measure of the overall service delay. Our approach differs from what is proposed in [15], [22], where constraints on local and remote queues are imposed separately, and not *jointly* as in our case. In our case, we provide a truly joint optimization of radio and computation resources in a dynamic fashion and we are able to satisfy a constraint on the overall out-of-service probability. The proposed method requires the solution of a convex problem in each time slot, so that it can be implemented through efficient numerical tools [23]. Numerical results assess the performance of our solution, illustrating how, in its simplicity, it guarantees out of service probability and average delay constraints.

II. Problem Formulation

Let us consider a scenario where K UEs wish to offload computations to a MEH, connected to a mmWave AP via a high capacity backhaul, as in the example shown in figure 1. Since we deal with a dynamic problem, time is divided in slots of equal duration τ. In each time slot t, new computation requests are randomly generated at the UE side; the radio channel, denoted by $h_k(t)$, can also vary over time. Then, letting $p_k(t)$ be the transmit power of UE k, and considering a frequency division multiple access, the maximum data rate between UE k and the AP is given by:

$$R_k(t) = \beta_k(t) B \log_2 \left(1 + \frac{h_k(t) p_k(t)}{N_0 \beta_k(t) B}\right), \qquad (1)$$

where $\beta_k(t)$ is the portion of the bandwidth allocated to UE k, B is the total available bandwidth, and N_0 is the noise power spectral density.

We consider a local (at the mobile handset) queue of bits to be transmitted and a remote (at the MEH) computation queue for each UE (cf. Fig. 1). The local data queue of UE k, say,

$Q_k^l(t)$, takes on input the new data arrivals $A_k(t)$, randomly generated, and it is drained by transferring data to the MEH via the mmWave AP, thus evolving as:

$$Q_k^l(t+1) = \max\left(Q_k^l(t) - \tau R_k(t), 0\right) + A_k(t). \quad (2)$$

Similarly, the remote computation queue, say, $Q_k^r(t)$, is fed by the data arriving from the UEs and drained by the computation power of the MEH, and it evolves as follows:

$$\begin{aligned} Q_k^r(t+1) \; = \; & \max\left(Q_k^r(t) - \tau f_k(t) J_k, 0\right) \\ & + \min\left(Q_k^l(t), \tau R_k(t)\right) \end{aligned} \quad (3)$$

where $f_k(t)$ is the total computation power (in CPU cycles/s) assigned to UE k during time slot t; and J_k denotes the number of bits per CPU cycle, a parameter that depends on the specific application required by UE k. The overall delay is then associated to the sum of the time needed to send the data in the local data and the time to run all computation requests associated to the remote computation queue. The overall delay is then linked to the sum of local and remote queue lengths:

$$Q_k^{\mathrm{tot}}(t) = Q_k^l(t) + Q_k^r(t). \quad (4)$$

The goal of this work is to find an optimal resource allocation strategy in order to minimize the long-term average power consumption at each UE, under constraints on the maximum average queue length (which can be directly related to the average delay by Little's law [24]) and the out of service probability, i.e. the probability that $Q_k^{\mathrm{tot}}(t)$ in (4) exceeds a certain value. The problem can be formulated as follows:

$$\begin{aligned} & \min_{\mathbf{\Psi}(t)} \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \sum_{k=1}^{K} \mathbb{E}\left\{p_k(t)\right\} \\ & \text{subject to} \\ & (a) \quad \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}\big[Q_k^{\mathrm{tot}}(t)\big] \leq Q_k^{\mathrm{avg}}, \;\; \forall k; \\ & (b) \quad \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \Pr\left\{\mathrm{Q_k^{tot}(t) > Q_k^{max}}\right\} \leq \epsilon_{\mathrm{k}}, \;\; \forall k; \\ & (c) \quad 0 \leq p_k(t) \leq P_k, \qquad \forall k, t; \\ & (d) \quad 0 \leq \beta_k(t) \leq 1, \qquad \forall k, t; \\ & (e) \quad \sum_{k=1}^{K} \beta_k(t) \leq 1, \quad \forall t; \\ & (f) \quad 0 \leq f_k(t) \leq f_{\max}, \qquad \forall k, t; \\ & (g) \quad \sum_{k=1}^{K} f_k(t) \leq f_{\max}, \quad \forall t; \end{aligned} \quad (5)$$

where $\mathbf{\Psi}(t) = \big[\{p_k(t)\}_k, \{f_k(t)\}_k, \; \{\beta_k(t)\}_k\big]$; the expectation is taken with respect to the channel and arrival rate realizations, and it depends on the control policy; Q_k^{avg} and $Q_k^{\max}$ are the upper bounds on the average queue length and on the maximum queue length for the out of service probability, respectively; ϵ_k is the out-of-service probability, while P_k and $f_{\max}$ are the UE transmit power budget and the computational power of the MEH, respectively. The constraints have the following meaning: (a) imposes that the average queue length (i.e., the average delay) of each UE does not exceed a certain value; (b) ensures that the probability for the total queue in (4) to exceed a maximum value does not exceed the required out of service probability; (c) ensures that the transmit power of each user is non negative and does not exceed a maximum power budget; (d) ensures that the fraction of the bandwidth allocated to each user is non negative and is at most 1; (e) guarantees that the sum of the allocated bandwidth to all users does not exceed the available bandwidth; (f) forces the computation resources allocated to each user to be non negative and not greater than the computation power of the MEH $f_{\max}$; (g) guarantees that the sum of the computation resources allocated to each user is at most equal to the computational power of the MEH.

III. Algorithm development

We tackle problem (5) using tools from stochastic optimization [25]. To this aim, we introduce two *virtual queues* corresponding to constraints (a) and (b) in (5). Denoting by $Z_k(t)$ the virtual queue of UE k associated to the first constraint, we can write its evolution as follows:

$$Z_k(t+1) = \max\left[0, Z_k(t) + Q_k^{\mathrm{tot}}(t+1) - Q_k^{\mathrm{avg}}\right]. \quad (6)$$

To introduce the second virtual queue, we recast constraint (b) in the following equivalent form:

$$\lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}\left\{\mathbf{1}\left\{\mathrm{Q_k^{tot}(t) > Q_k^{max}}\right\}\right\} \leq \epsilon_{\mathrm{k}}, \quad (7)$$

where $\mathbf{1}\{\cdot\}$ is the indicator function. Since the indicator function in (7) can be rewritten as

$$\mathbf{1}\left\{Q_k^{\mathrm{tot}}(t) > Q_k^{\max}\right\} = u\left\{Q_k^{\mathrm{tot}}(t) - Q_k^{\max}\right\}, \quad (8)$$

where $u(\cdot)$ denotes the unitary step function, the virtual queue $Y_k(t)$ associated to the second constraint in (5) evolves as:

$$\begin{aligned} Y_k(t+1) = \max\big[0, & Y_k(t) \\ & + \mu\left(u\left\{Q_k^{\mathrm{tot}}(t+1) - Q_k^{\max}\right\} - \epsilon_k\right)\big], \end{aligned} \quad (9)$$

where μ is a step-size used to speed up the convergence of the algorithm. Note that the use of the step size does not change the problem, since it comes just from the scalar multiplication of both sides of constraint (7) by a factor μ. Having introduced the virtual queues $Z_k(t)$ and $Y_k(t)$ for each UE k, the constraints (a) and (b) in (5) can be substituted by mean-rate stability constraints of the virtual queues as follows:

$$(a) \quad \lim_{T\to\infty} \frac{\mathbb{E}[Z_k(T)]}{T} = 0, \;\; \forall k \quad (10)$$

$$(b) \quad \lim_{T\to\infty} \frac{\mathbb{E}[Y_k(T)]}{T} = 0, \;\; \forall k \quad (11)$$

The algorithmic solution passes through the definition of the Lyapunov function

$$L(\mathbf{\Theta}(t)) = \frac{1}{2} \sum_{k=1}^{K} \left[Z_k(t)^2 + Y_k(t)^2\right], \quad (12)$$

where $\mathbf{\Theta}(t) = [\mathbf{Z}(t), \mathbf{Y}(t)]$, and $\mathbf{Z}(t), \mathbf{Y}(t)$ are the vectors whose elements are the virtual queues of all UEs. Then, the Lyapunov drift is defined as [25]

$$\Delta(\mathbf{\Theta}(t)) \triangleq \mathbb{E}\{L(\mathbf{\Theta}(t+1)) - L(\mathbf{\Theta}(t))|\mathbf{\Theta}(t)\}, \tag{13}$$

The Lyapunov drift defined in (13) leads to the mean-rate stability of the virtual queues [i.e., (10) and (11)], but it can also lead to an unnecessary power consumption. To balance the mean-rate stability and the long-term average power consumption, we introduce the *drift-plus-penalty* function [25]:

$$\Delta_p(\mathbf{\Theta}(t)) = \Delta(\mathbf{\Theta}(t)) + V \cdot \mathbb{E}\left\{\sum_{k=1}^{K} p_k(t)\big|\mathbf{\Theta}(t)\right\} \tag{14}$$

where V is a control parameter used to balance the power consumption and the Lyapunov drift. Using a stochastic optimization approach, our algorithm is based on the concept of opportunistically minimizing an upper bound of the drift-plus-penalty function in a per slot fashion. It can be shown that an upper bound to (14) is given by [26]:

$$\begin{aligned}\Delta_p(\mathbf{\Theta}(t)) \leq C + \mathbb{E}\Bigg\{ \sum_{k=1}^{K} \Bigg[& Z_k(t)\Big(\max(0, Q_k^l(t) - \tau R_k(t)) \\ & + \max(0, Q_k^r(t) - \tau f_k(t) J_k)\Big) \\ & + \mu Y_k(t)\, u\Big(\max(0, Q_k^l(t) - \tau R_k(t)) \\ & + \max(0, Q_k^r(t) - \tau f_k(t) J_k) \\ & + \tau R_{k,\max}(t) + A_k(t) - Q_k^{\max}\Big\} + V \cdot p_k(t)\Bigg] \Bigg| \mathbf{\Theta}(t)\Bigg\},\end{aligned} \tag{15}$$

where C is a positive constant, and where $R_{k,\max}(t)$ is an upper bound on the data rate. Since the step function in (15) is non-convex, we exploit its closest convex upper bound

$$u(x) \leq \max(0, 1 + x), \tag{16}$$

which is reminiscent of the hinge loss used in support vector machines [27]. Then, using (16) in (15), we obtain:

$$\begin{aligned}\Delta_p(\mathbf{\Theta}(t)) \leq C + \mathbb{E}\Bigg\{ \sum_{k=1}^{K} \Bigg[& Z_k(t)\Big(\max\left(0, Q_k^l(t) - \tau R_k(t)\right) \\ & + \max\left(0, Q_k^r(t) - \tau f_k(t) J_k\right)\Big) \\ & + \mu Y_k(t) \max\Big(0, \max(0, Q_k^l(t) - \tau R_k(t)) \\ & + \max(0, Q_k^r(t) - \tau f_k(t) J_k) + \delta_k(t)\Big) + V p_k(t)\Bigg] \Bigg| \mathbf{\Theta}(t)\Bigg\},\end{aligned} \tag{17}$$

where $\delta_k(t) = \tau R_{k,\max}(t) + A_k(t) - Q_k^{\max} + 1$. Thus, the algorithm proceeds by greedily minimizing instantaneous values of the upper bound in (17), thus obtaining the following dynamic control policy:

$$\begin{aligned}\min_{\mathbf{\Psi}(t)} \quad & \sum_{k=1}^{K} \Big[Z_k(t)\big[\max(0, Q_k^l(t) - \tau R_k(t)) \\ & + \max(0, Q_k^r(t) - f_k(t) J_k)\big] \\ & + Y_k(t) \max\Big(0, \max(0, Q_k^l(t) - \tau R_k(t)) \\ & + \max(0, Q_k^r(t) - \tau f_k(t) J_k) + \delta_k(t)\Big) + V p_k(t)\Big] \\ & \text{subject to} \quad \mathbf{\Psi}(t) \in \mathcal{Z}(t)\end{aligned} \tag{18}$$

where $\mathcal{Z}(t)$ is the set of feasible actions according to the constraints $(c)-(g)$ of problem (5). It is easy to prove that (18) is a convex optimization problem, but having a non-differentiable objective function. To handle non-differentiability, we first perform a simple change of variable, in order to use the data rate $R_k(t)$ as a variable instead of the transmit power $p_k(t)$. In particular, the transmit power can be written as:

$$p_k(t) = \frac{\beta_k(t) B}{h_k(t)} \left[\exp\left(\frac{R_k(t) \log_e(2)}{\beta_k(t) B} \right) - 1 \right]. \tag{19}$$

Then, exploiting the equivalent epigraph form [23], it is possible to show that (18) can be equivalently recast as follows [26]:

$$\begin{aligned}\min_{\mathbf{\Omega}(t)} \quad & \sum_{k=1}^{K} \Bigg[V \cdot \frac{\beta_k(t) B}{h_k(t)} \left[\exp\left(\frac{R_k(t) \log_e(2)}{\beta_k(t) B} \right) - 1 \right] \\ & + Z_k(t)\Big(\xi_k(t) + \Gamma_k(t) \Big) + \mu Y_k(t) \Phi_k(t) \Bigg] \\ & \text{subject to} \\ & (a)\ 0 \leq \frac{\beta_k(t) B}{h_k(t)} \left[\exp\left(\frac{R_k(t) \log_e(2)}{\beta_k(t) B} \right) - 1 \right] \leq P_k,\ \forall k, t; \\ & (b)\ 0 \leq \beta_k(t) \leq 1, \quad \forall k, t; \\ & (c)\ \sum_{k=1}^{K} \beta_k(t) \leq 1, \quad \forall t; \\ & (d)\ 0 \leq f_k(t) \leq f_{\max}, \quad \forall k, t; \\ & (e)\ \sum_{k=1}^{K} f_k(t) \leq f_{\max}, \quad \forall t; \\ & (f)\ \xi_k(t) \geq 0, \quad \forall k, t; \\ & (g)\ \xi_k(t) \geq Q_k^l(t) - \tau R_k(t), \quad \forall k, t; \\ & (h)\ \Gamma_k(t) \geq 0, \quad \forall k, t; \\ & (i)\ \Gamma_k(t) \geq Q_k^r(t) - \tau f_k(t) J_k, \quad \forall k, t; \\ & (l)\ \Phi_k(t) \geq 0, \quad \forall k, t; \\ & (m)\ \Phi_k(t) \geq \delta_k(t), \quad \forall k, t; \\ & (n)\ \Phi_k(t) \geq Q_k^l(t) - \tau R_k(t) + \delta_k(t), \quad \forall k, t; \\ & (o)\ \Phi_k(t) \geq Q_k^r(t) - \tau f_k(t) J_k + \delta_k(t), \quad \forall k, t; \\ & (p)\ \Phi_k(t) \geq Q_k^l(t) - \tau R_k(t) + Q_k^r(t) \\ & \qquad -\tau f_k(t) J_k + \delta_k(t),\ \forall k, t;\end{aligned} \tag{$\mathcal{P}$}$$

where $\mathbf{\Omega}(t) = [\{R_k(t)\}_k, \{f_k(t)\}_k, \{\beta_k(t)\}_k, \{\xi_k(t)\}_k, \{\Gamma_k(t)\}_k, \{\Phi_k(t)\}_k]$. It is easy to see that problem $(\mathcal{P})$ is convex and differentiable, and can be solved using powerful numerical tools as interior point methods [23]. In fact, almost

all functions in $(\mathcal{P})$ are linear, except for (19), which is the perspective function that is known to be convex [23]. The overall dynamic procedure is described in Algorithm 1.

Algorithm 1 : Dynamic Latency-constrained Computation Offloading Algorithm

Data: K, N_{slot}, τ, J_k, V, P_t, B, $A_{k,\max}$, Q_k^{avg}, $Q_k^{\max}$, ϵ_k, $f_{\max}$. Set μ, $Z_k(0)$, $Y_k(0)$;
For $t = 1 : N_{\text{slot}}$
(S.1): Observe the radio channels $h_k(t)$ for all k;
(S.2): Solve problem $\mathcal{P}$;
(S.3): Update $Q_k^r(t)$ as in (3);
(S.4): Observe $A_k(t)$ and update $Q_k^l(t)$ as in (2);
(S.5): Update $Z_k(t)$ and $Y_k(t)$ as in (6) and (9) , respectively;
End

IV. Numerical results

In this section, we show the performance of our algorithm through numerical results obtained by MATLAB simulations, using the *fmincon* function from the optimization toolbox. Since problem $(\mathcal{P})$ is convex, *fmincon* converges to the global optimal solution very efficiently. We consider a pathloss mmWave link as in [28], an available bandwidth of 200 MHz, a noise power spectral density of -174 dBm/Hz, and a mmWave AP at the center of a square of size 100 m. The single MEH has a computational power $f_{\max} = 5 \times 10^9$ CPU cycles/s, and the parameter J_k is set to 10^{-1} bits/CPU cycle for all k. The maximum transmit power of each user is $P_k = 500$ mW, and each terminal is endowed with a planar array of 4 antennas. At the receive side, the AP has an array of 16 elements. In Fig. 2, we show the tradeoff between the average user queue length and the average user transmit power, comparing our algorithm with the algorithm proposed in [11], which requires only mean rate stability of the sum of the computation queues. In this evaluation we considered a scenario with 15 users with an arrival rate uniformly distributed between 0 and $A_{k,\max} = 6 \times 10^5$ bits in each time slot. The requirements are $Q_k^{\text{avg}} = 3 \times 10^6$ bits, $Q_k^{\max} = 6 \times 10^6$ bits and $\epsilon_k = 10^{-2}$. Simulations are run for 10000 slots with $\tau = 10$ ms, and are averaged over 100 channel realizations, given by different positions of the UEs. For the virtual queue $Y_k(t)$, we used a step-size $\mu = 1000$. The power/delay tradeoff is explored by letting the parameter V of (14) to vary along the curves reported in Fig. 2 (as V decreases, the average power increases). In particular, V increases going from rigth to left along the abscissa. As we can notice from Fig. 2, the proposed method obtains a considerable gain with respect to the strategy in [11] in terms of queue length/power tradeoff. In particular, with the proposed method, the average queue length approaches the maximum average requirement as V increases, whereas the algorithm in [11] incurs in a much longer total user queue length for a given power. Note that, since we imposed constraints on the average delay and on the maximum queue length, our strategy does not arbitrarily decrease the power consumption as V increases,

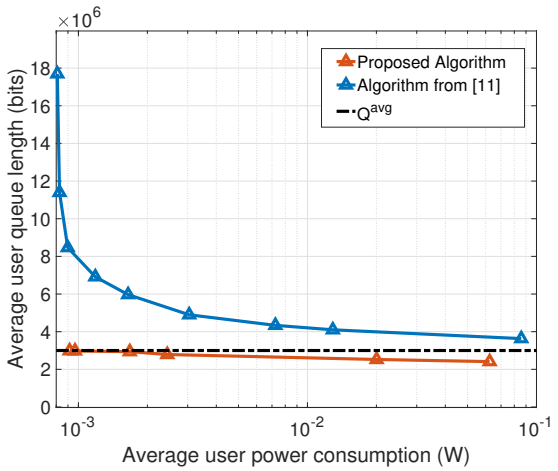

Fig. 2: Average user queue length vs average user power consumption, for different algorithms.

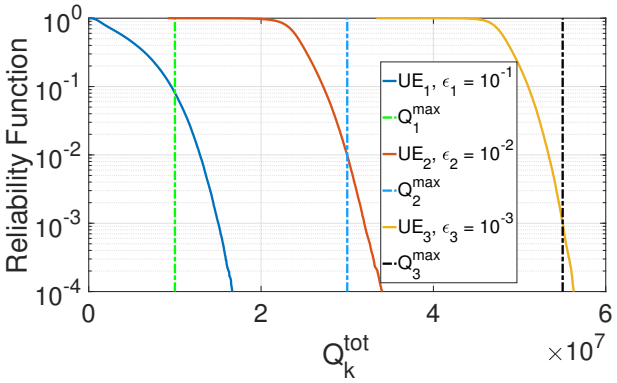

Fig. 3: Probability of exceeding the value on the abscissa.

but it reaches a minimum power such that these constraints are satisfied. The only drawback of increasing V, and thus finding the minimum power value, is the convergence time. On the contrary, the algorithm proposed in [11] can arbitrarily decrease the transmit power consumption at the cost of a larger average queue length.

As a further example, in Fig. 3, we show the behavior of the reliability function defined as $1 - \text{CDF}(Q_k^{\text{tot}}(t))$, where $\text{CDF}(\cdot)$ is the cumulative distribution function. We consider 3 users with different Q_k^{avg}, $Q_k^{\max}$, and ϵ_k, running the simulation for 300000 slots, averaging over the last 250000 slots, and considering $V = 4 \times 10^{16}$. Each curve shows the probability that Q_k^{tot} is greater than the value on the abscissa, while the vertical lines represent the maximum requirements $Q_k^{\max}$, $k = 1, 2, 3$. From Fig. 3, we can notice that all the users meet the required constraint on the out of service probability. Finally, in Fig. 4, we show the instant value of the sum queue length for the 3 UEs with the same simulation parameters of Fig. 3. In this figure we can notice the effectiveness of the algorithm in terms of average queue length and, at the same time, the effect of the bound on the out of service probability. Indeed, while the first UE requires an out of service probability $\epsilon_1 = 10^{-1}$ and its queues often exceeds the

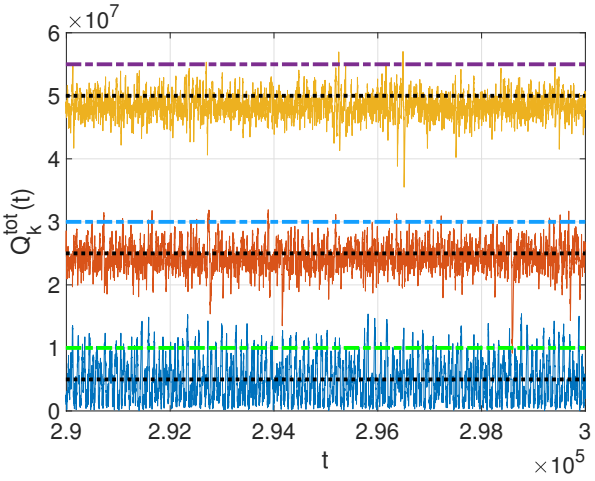

Fig. 4: Instantaneous sum queue length vs. iteration index

prescribed threshold $Q_1^{\max}$, UE 2 and UE 3 present much less peaks exceeding their thresholds, since they require a much lower value of ϵ_k. At the same time, the bound on the average queue length is always met by all UEs.

V. Conclusions and Future Work

In this paper we studied the problem of dynamic resource allocation for computation offloading with MEC, considering constraints on the average delay and out of service probability evaluated with respect to the sum of the local and remote queues. We proposed an algorithm that jointly optimize radio and computation resources based on stochastic optimization, in order to deal with the unknown statistics of the mmWave channel and the data arrival process. The formulation leads to a very simple and efficient (convex) algorithmic solution that, through several numerical results, is shown to effectively meet the constraints on the average queue length and on the out-of-service probability, while minimizing the average power spent for transmission. It is worth emphasizing that our approach handling the communication and computation queues jointly is fundamental to guarantee the out-of-service constraint, as opposed to alternative strategies available in the literature handling the two queues separately. Several open research directions can be investigated in the future. A first possibility is to extend the problem to a multiple AP and MEH case, also introducing mobility management. Also, the intermittent behavior of mmWave links due to blockages has to be addressed to avoid service interruptions.

References

[1] Ericsson, "Ericsson mobility report," *Available Online*, Jun. 2018.

[2] B. Raaf, M. Faerber, B. Badic, , and V. Frascolla, "Key technology advancements driving mobile communications from generation to generation," *Intel Technology Journal*, vol. 18, 2014.

[3] M. Tercero et al., "5G systems: The mmMAGIC project perspective on use cases and challenges between 6-100 Ghz," in *2016 IEEE Wireless Communications and Networking Conference*, April 2016, pp. 1–6.

[4] "ETSI multi-access edge computing," *https://www.etsi.org/technologies-clusters/technologies/multi-access-edge-computing*.

[5] "5G-MiEdge millimeter-wave edge cloud as an enabler for 5G ecosystem," *Available online at http://5g-miedge.eu*.

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

[7] S. Barbarossa, E. Ceci, M. Merluzzi, and E. Calvanese-Strinati, "Enabling effective mobile edge computing using millimeterwave links," in *Proc. of IEEE Int. Conf. Commun. (ICC) Work.*, May 2017, pp. 367–372.

[8] S. Barbarossa, E. Ceci, and M. Merluzzi, "Overbooking radio and computation resources in mmw-mobile edge computing to reduce vulnerability to channel intermittency," in *Proc. of 2017 Eur. Conf. Net. Commun. (EuCNC)*, Jun. 2017, pp. 1–5.

[9] W. Labidi, M. Sarkiss, and M. Kamoun, "Energy-optimal resource scheduling and computation offloading in small cell networks," in *Proc. of 2015 22nd International Conference on Telecommunications (ICT)*, Sydney, NSW, Australia 2015, pp. 313–318.

[10] 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.

[11] Y. Mao, J. Zhang, S. H. Song, and K. B. Letaief, "Stochastic joint radio and computational resource management for multi-user mobile-edge computing systems," *IEEE Trans. Wireless Commun.*, vol. 16, no. 9, pp. 5994–6009, Sept 2017.

[12] Y. Yang, S. Zhao, W. Zhang, Y. Chen, X. Luo, and J. Wang, "DEBTS: Delay energy balanced task scheduling in homogeneous fog networks," *IEEE Internet of Things Journal*, vol. 5, no. 3, pp. 2094–2106, 2018.

[13] Y. Sun, S. Zhou, and J. Xu, "EMM: Energy-aware mobility management for mobile edge computing in ultra dense networks," *IEEE Journal on Sel. Areas in Comm.*, vol. 35, no. 11, pp. 2637–2646, Nov 2017.

[14] S. Sardellitti, M. Merluzzi, and S. Barbarossa, "Optimal association of mobile users to multi-access edge computing resources," in *Proc. of 2018 IEEE International Conference on Communications Workshops (ICC Workshops)*, May 2018, pp. 1–6.

[15] L. Chen-Feng, M. Bennis, and H.V. Poor, "Latency and reliability-aware task offloading and resource allocation for mobile edge computing," in *Proc. of 2017 IEEE Globecom Workshops (GC Wkshps)*, Singapore 2017, pp. 1–7.

[16] S. Barbarossa, S. Sardellitti, E. Ceci, and M. Merluzzi, "The edge cloud: A holistic view of communication, computation, and caching," in *Chapter 16 of Cooperative and Graph Signal Processing*. 2018, pp. 419 – 444, Academic Press.

[17] P. Mach and Z. Becvar, "Mobile edge computing: A survey on architecture and computation offloading," *IEEE Communications Surveys Tutorials*, vol. 19, no. 3, pp. 1628–1656, 2017.

[18] N. di Pietro, M. Merluzzi, E. Calvanese Strinati, and S. Barbarossa, "Resilient design of 5G Mobile-Edge computing over Intermittent mmwave links," *Available online: http://arxiv.org/abs/1901.01894*, 2019.

[19] C. You, K. Huang, H. Chae, and B. H. Kim, "Energy-efficient resource allocation for mobile-edge computation offloading," *IEEE Trans. Wir. Commun.*, vol. 16, no. 3, pp. 1397–1411, Mar. 2017.

[20] P. Zhao, H. Tian, C. Qin, and G. Nie, "Energy-saving offloading by jointly allocating radio and computational resources for mobile edge computing," *IEEE Access*, vol. 5, pp. 11255–11268, 2017.

[21] M. I. Ashraf, C. Liu, M. Bennis, W. Saad, and C. S. Hong, "Dynamic resource allocation for optimized latency and reliability in vehicular networks," *IEEE Access*, vol. 6, 2018.

[22] L. Chen-Feng, M. Bennis, M. Debbah, and H.V. Poor, "Dynamic task offloading and resource allocation for ultra-reliable low-latency edge computing," *[Online]. Available: arXiv:1812.08076*.

[23] S. Boyd and L. Vandenberghe, *Convex optimization*, Cambridge university press, 2004.

[24] John DC Little, "Little's law as viewed on its 50th anniversary," *Operations research*, vol. 59, no. 3, pp. 536–549, 2011.

[25] Michael J. Neely, *Stochastic Network Optimization with Application to Communication and Queueing Systems*, Morgan and Claypool Publishers, 2010.

[26] M. Merluzzi, P. Di Lorenzo, S. Barbarossa, and V. Frascolla, "Dynamic computation offloading in mobile edge computing via ultra-reliable and low latency communications," *Submitted to IEEE Transactions on Mobile Computing*, 2019.

[27] Osvaldo Simeone, "A brief introduction to machine learning for engineers," *Foundations and Trends in Signal Processing*, vol. 12, no. 3-4, pp. 200–431, 2018.

[28] R. J. Weiler et al., "Outdoor millimeter-wave access for heterogeneous networks path loss and system performance," in *2014 IEEE 25th Annual International Symposium on Personal, Indoor, and Mobile Radio Communication (PIMRC)*, Sep. 2014, pp. 2189–2193.