Bayesian Optimization for Task Offloading and Resource Allocation in Mobile Edge Computing

Jia Yan, Qin Lu, and Georgios B. Giannakis
Dept. of Electrical and Computer Engineering, University of Minnesota, USA
Emails: {yanj,qlu,georgios}@umn.edu

2022 56th Asilomar Conference on Signals, Systems, and Computers | 978-1-6654-5906-8/22/$31.00 ©2022 IEEE | DOI: 10.1109/IEEECONF56349.2022.10051868

***Abstract*—Recent years have witnessed the emergence of mobile edge computing (MEC), on the premise of a cost-effective enhancement in the computational ability of hardware-constrained wireless devices (WDs) comprising the Internet of Things (IoT). In a general multi-server multi-user MEC system, each WD has a computational task to execute and has to select binary (off)loading decisions, along with the analog-amplitude resource allocation variables in an online manner, with the goal of minimizing the overall energy-delay cost (EDC) with dynamic system states. While past works typically rely on the explicit expression of the EDC function, the present contribution considers a practical setting, where in lieu of system state information, the EDC function is not available in analytical form, and instead only the function values at queried points are revealed. Towards tackling such a challenging online combinatorial problem with only bandit information, novel Bayesian optimization (BO) based approach is put forth by leveraging the multi-armed bandit (MAB) framework. Per time slot, by exploiting temporal information, the discrete offloading decisions are first obtained via the MAB method, and the analog resource allocation variables are subsequently optimized using the BO selection rule. Numerical tests validate the effectiveness of the proposed BO approach.**

***Index Terms*—Mobile edge computing, Bayesian optimization, online learning, task offloading, resource allocation, Internet of Things.**

I. Introduction

Capitalizing on the mobile edge computing (MEC) architecture, wireless devices (WDs) equipped with low-power on-chip computing units in the Internet of Things (IoT), carry out high-performance computation by offloading tasks to the servers located at the network edge [1]. Due to the time-varying wireless channel conditions and the dynamic computing capacities at the edge servers, judiciously offloading computations can afford major performance enhancement. Prior works on offloading computations typically focus on offline algorithms, which assume that the system states are known a priori [2], [3], even though such knowledge is challenging to acquire beforehand.

Besides unknown system dynamics, the unpredictable WD preferences (e.g., service latency, reliability or privacy) render it prohibitive to model the objective function analytically in dynamic IoT environment. In fact, the IoT controller can only have available objective function values at queried points. In this context, the bandit convex optimization (BCO) [4], [5] leverages only point-wise values of objective functions for the gradient estimations. Tailored for partial task offloading strategies among multiple edge servers, BCO with both time-varying costs and constraints was studied in [6]. On the other hand, aiming at binary computational offloading strategies with such a bandit feedback, multi-armed bandit (MAB) based methods have been popular in MEC systems [7]–[10].

Although achieving promising results, the aforementioned BCO or MAB based works deal only with either continuous or discrete decision variables. In many practical settings though, the analog-amplitude communication and computation resource allocation variables (e.g., transmit power and local computing speed) need to be jointly optimized with discrete variables that capture offloading decisions for optimum MEC performance. Finely discretizing the analog action space (or relaxing the discrete task offloading decisions), renders the existing MAB methods (or the BCO approaches) inaccurate and computationally prohibitive. In addition, the convexity of objective functions commonly assumed in BCO algorithms may not hold in practice [1]–[3], [11]. Although dealing with arbitrary objective functions, MAB methods require to explore every single arm at least once to accumulate sufficient statistics, which may incur sudden performance drops and slow down the learning processes for large MEC networks [7]–[10].

Alleviating these limitations, we advocate a novel approach based on Bayesian optimization (BO) [12], [13] in conjunction with the MAB in order to solve this combinatorial optimization of discrete task offloading decisions and analog-amplitude resource allocation strategies in time-varying multi-server multi-user MEC systems with bandit feedback. Building on the BO framework for online bandit optimization of categorical and continuous decision variables, a Gaussian process (GP) [14]–[17] based surrogate model is adopted for the sought objective function with novel kernel design by incorporating temporal information. With the GP-based surrogate model, an innovative acquisition rule is developed in the time-varying BO scheme to select new optimization variables per iteration. Specifically, given the categorical offloading decisions obtained by the MAB-based method, the analog-amplitude resource allocation variables are determined using the conventional BO-based selection rule. Numerical tests demonstrate that our proposed BO approach outperforms the existing benchmarks.

Notation: $(\cdot)^\top$ and $(\cdot)^{-1}$ denote transpose and matrix inverse, respectively, and $\|\mathbf{x}\|$ stands for the l_2-norm of a vector

This work was supported by NSF grants 2102312, 2103256, 1901134, 2126052, 2128593, and 2220292.

$\mathbf{x}$. Besides, $\mathbf{0}_t$, $\mathbf{1}_t$ and $\mathbf{I}_t$ denote the $t \times 1$ all-zero vector, the $t \times 1$ all-one vector and the $t \times t$ identity matrix, respectively. Inequalities for vector $\mathbf{x} > \mathbf{0}$ are entry-wise. $\mathbb{I}(x = x')$ denotes the indicator function taking the value of 1 if $x = x'$, and 0 otherwise. $\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \mathbf{K})$ stands for the probability density function (pdf) of a Gaussian random vector $\mathbf{x}$ with mean $\boldsymbol{\mu}$ and covariance $\mathbf{K}$.

II. System Model and Problem Formulation

Consider a MEC system with M WDs, and N base stations (BSs). Each BS $n \in \mathcal{N} := \{1, \ldots, N\}$ is the gateway of edge servers to provide MEC services to the power-limited WDs indexed by $m \in \mathcal{M} := \{1, \ldots, M\}$. Per slot $t \in \mathcal{T} := \{1, ..., T\}$, the m-th WD has a computational task characterized by the pair (I_t^m, L_t^m), where I_t^m denotes the size of input data in bits, and L_t^m represents the workload in terms of the total number of CPU cycles to execute the aforementioned task. This WD could either execute its task locally or offload it to one of the BSs, a choice that is henceforth captured by the categorical variable $c_t^m \in \{0, 1, ..., N\}$. Specifically, $c_t^m = 0$ indexes local computing, and $c_t^m = n, n \in \mathcal{N}$, stands for offloading task to BS n, i.e.,

$$c_t^m = \begin{cases} 0, & \text{local computing} \\ n, & \text{offloading to BS } n \end{cases} \quad \forall m \in \mathcal{M}, n \in \mathcal{N}, t \in \mathcal{T}.$$

For both scenarios, the computational overhead per task consists of the execution delay and energy consumption, which will be elaborated as follows.

A. Local Computing

If WD m chooses to execute its task locally (i.e., $c_t^m = 0$) per slot t, it has to select the local CPU frequency f_t^m, based on which the task computing time is given by $\tau_{l,t}^m = \frac{L_t^m}{f_t^m}$ and the corresponding energy consumption is

$$\epsilon_{l,t}^m = \xi L_t^m (f_t^m)^2 \tag{1}$$

where ξ denotes the effective switched capacitance parameter.

B. Edge Computing

If WD m alternatively goes for edge computing at BS n per slot t, that is, $c_t^m = n$, it must first offload the task using transmit power p_t^m. Suppose that the wireless channel coefficient between WD m and BS n for task offloading is $h_t^{m,n}$, and the receiver is corrupted by additive white Gaussian noise (AWGN) with mean zero and variance σ^2. Here, the wireless channel is assumed to be invariant within each slot and may change across different slots. Then, the uplink transmission data rate for the sought offloading task is $R_t^{m,n} = W \log_2(1 + p_t^m |h_t^{m,n}|^2 / \sigma^2)$, where W is the identical bandwidth of the dedicated spectral resource block allocated to each WD. Accordingly, the offloading transmission time is $\tau_{u,t}^m = \sum_{n=1}^N \mathbb{I}(c_t^m = n) I_t^m / R_t^{m,n}$ and the transmission energy consumption of WD m is $\epsilon_{u,t}^m = p_t^m \tau_{u,t}^m$.

For edge computing at BS n, the total computation resource per slot t is signified by the CPU frequency $f_{c,t}^n$. Upon receiving all the offloaded tasks, the edge server generates multiple virtual machines (VMs) to execute the tasks in parallel, and equally partitions $f_{c,t}^n$ to yield $f_{c,t}^n / (1 + \sum_{m' \in \mathcal{M}/m} \mathbb{I}(c_t^{m'} = n))$ per task. The edge execution time for WD m's task is thus

$$\tau_{c,t}^m = \sum_{n=1}^N \mathbb{I}(c_t^m = n) \frac{L_t^m (1 + \sum_{m' \in \mathcal{M}/m} \mathbb{I}(c_t^{m'} = n))}{f_{c,t}^n}. \tag{2}$$

C. Problem Formulation

Accounting for both local and edge computing, the total time delay and energy consumption for executing the task at WD m per slot t are given by $D_t^m = \mathbb{I}(c_t^m = 0) \tau_{l,t}^m + \mathbb{I}(c_t^m \neq 0)(\tau_{u,t}^m + \tau_{c,t}^m)$ and $E_t^m = \mathbb{I}(c_t^m = 0) \epsilon_{l,t}^m + \mathbb{I}(c_t^m \neq 0) \epsilon_{u,t}^m$, respectively. Taking a weighted sum of task execution time delay D_t^m and energy consumption E_t^m yields the energy-delay cost (EDC) per WD m as

$$EDC_t^m(c_t^m, f_t^m, p_t^m) = \beta_d D_t^m + \beta_e E_t^m \tag{3}$$

where β_d, β_e are positive scalars that balance these two costs. For notational brevity, collect the optimization variables in $\mathbf{c}_t := [c_t^1, \ldots, c_t^M]^\top$, $\mathbf{p}_t := [p_t^1, \ldots, p_t^M]^\top$, and $\mathbf{f}_t := [f_t^1, \ldots, f_t^M]^\top$. The objective is to choose online (at the beginning of each slot t) the categorical task offloading decisions (i.e., $\mathbf{c}_t$) and analog-amplitude resource allocation strategies (i.e., $\mathbf{p}_t, \mathbf{f}_t$) minimizing the accumulated EDC across all WDs, that is

$$\text{(P1)} \quad \min_{\{\mathbf{c}_t, \mathbf{p}_t, \mathbf{f}_t\}_t} \quad \sum_{t=1}^T \sum_{m=1}^M EDC_t^m(c_t^m, f_t^m, p_t^m),$$
$$\text{s.t.} \quad c_t^m \in \{0, 1, 2, ..., N\}, \ 0 < p_t^m \leq P_{peak}, \ 0 < f_t^m \leq f_{peak}, \ \forall m \in \mathcal{M}, t \in \mathcal{T}$$

where f_{peak} and P_{peak} are the peak local CPU frequency and transmit power of the WDs, respectively. By further introducing $\mathbf{x}_t := [\mathbf{p}_t^\top, \mathbf{f}_t^\top]^\top$ and the reward function $\varphi_t(\mathbf{c}_t, \mathbf{x}_t) := -\sum_{m=1}^M EDC_t^m$ per slot t, (P1) can be equivalently expressed as

$$\text{(P2)} \quad \max_{\{\mathbf{c}_t, \mathbf{x}_t\}_t} \quad \sum_{t=1}^T \varphi_t(\mathbf{c}_t, \mathbf{x}_t),$$
$$\text{s.t.} \quad \mathbf{c}_t \in \{0, 1, 2, ..., N\}^M, \ \mathbf{0} < \mathbf{x}_t \leq \mathbf{x}_{peak}, \forall t \in \mathcal{T}$$

where $\mathbf{x}_{peak} := [P_{peak} \mathbf{1}_M^\top, f_{peak} \mathbf{1}_M^\top]^\top$, and $\mathbf{1}_M$ is the M-dimensional all-one column vector.

A major challenge facing (P2) (equivalently (P1)) is that the wireless channels $\{h_t^{m,n}\}$, the edge computing capacities $\{f_{c,t}^n\}$, the computational task characterization $\{I_t^m, L_t^m\}$ are not available; thus, the explicit form of the time-varying EDC function is unknown when making the task offloading and resource allocation decisions $\{\mathbf{c}_t, \mathbf{x}_t\}$ per slot. After performing $\{\mathbf{c}_t, \mathbf{x}_t\}$, only noisy EDC function value (equivalently the realization of $\varphi_t(\mathbf{c}_t, \mathbf{x}_t)$) at that queried point can be acquired at the end of slot t. The difficulty of such a bandit setup is further exacerbated by its combinatorial nature that calls for the joint optimization of the categorical $\mathbf{c}_t$ and continuous $\mathbf{x}_t$. To tackle this bandit mix-integer program, novel BO-based approach will be pursued in the following section.

III. Time-Varying BO for Dynamic MEC Management

BO has well-documented merits in optimizing black-box functions that arise in several settings [12]. To account for the temporal variation arising from unknown system dynamics (e.g., changing channel conditions and computing capacities of the edge servers), the slot index t is augmented as an additional input of the sought black-box function, i.e., $\varphi(\mathbf{c}_t, \mathbf{x}_t, t) := \varphi_t(\mathbf{c}_t, \mathbf{x}_t)$. In short, BO seeks to maximize the black-box $\varphi(\mathbf{z}_t)$ with $\mathbf{z}_t := [\mathbf{c}_t^\top, \mathbf{x}_t^\top, t]^\top$ by sequentially acquiring function observations using a surrogate model. Collect all the acquired data up to slot t in $\mathcal{D}_t := \{(\mathbf{z}_\tau, y_\tau)\}_{\tau=1}^t$ with y_τ denoting the possibly noisy observation of $\varphi(\mathbf{z}_\tau)$. Each BO iteration consists of i) obtaining the function posterior pdf $p(\varphi(\mathbf{z})|\mathcal{D}_t)$ based on the chosen surrogate model using $\mathcal{D}_t$; and, ii) selecting $\mathbf{z}_{t+1}$ to evaluate at the beginning of slot $t+1$, whose observation y_{t+1} will be acquired at the end of slot $t+1$. In the following, we will introduce the GP-based surrogate model and the acquisition rule for $\mathbf{z}_{t+1}$, respectively.

A. GP-based Surrogate Model for Time-Varying Function φ and Kernel Design

As an established Bayesian nonparametric approach, the GP can learn black-box functions with quantifiable uncertainty and sample efficiency, making it suitable for surrogate modeling in BO. Specifically, given data $\mathcal{D}_t$, the goal is to learn the function $\varphi(\cdot)$ that links the input $\mathbf{z}_\tau$ with the scalar output y_τ as $\mathbf{z}_\tau \to \varphi(\mathbf{z}_\tau) \to y_\tau$. Towards this, a GP prior is assumed on the unknown φ as $\varphi \sim \mathcal{GP}(0, \kappa(\mathbf{z}, \mathbf{z}'))$, where $\kappa(\cdot,\cdot)$ is a kernel (covariance) function measuring pairwise similarity of any two inputs. Then, the joint prior pdf of any t function evaluations $\boldsymbol{\varphi}_t := [\varphi(\mathbf{z}_1), ..., \varphi(\mathbf{z}_t)]^\top$ at inputs $\mathbf{Z}_t := [\mathbf{z}_1, ..., \mathbf{z}_t]^\top$ is jointly Gaussian distributed as [14]

$$p(\boldsymbol{\varphi}_t|\mathbf{Z}_t) = \mathcal{N}(\boldsymbol{\varphi}_t; \mathbf{0}_t, \mathbf{K}_t), \forall t \tag{4}$$

where $\mathbf{K}_t$ is a $t \times t$ covariance matrix with (τ, τ')-th entry $[\mathbf{K}_t]_{\tau,\tau'} = \text{cov}(\varphi(\mathbf{z}_\tau), \varphi(\mathbf{z}_{\tau'})) := \kappa(\mathbf{z}_\tau, \mathbf{z}_{\tau'})$. The estimation of φ relies on the observed outputs $\mathbf{y}_t := [y_1, ..., y_t]^\top$ that are linked with $\boldsymbol{\varphi}_t$ through the Gaussian conditional likelihood $p(\mathbf{y}_t|\boldsymbol{\varphi}_t, \mathbf{Z}_t) = \mathcal{N}(\mathbf{y}_t; \boldsymbol{\varphi}_t, \sigma_o^2\mathbf{I}_t)$, where σ_o^2 is the noise variance. Along with the GP prior in (4), one can readily obtain the function posterior pdf $p(\varphi(\mathbf{z})|\mathcal{D}_t)$ via Bayes' rule as

$$p(\varphi(\mathbf{z})|\mathcal{D}_t) = \mathcal{N}(\varphi(\mathbf{z}); \mu_t(\mathbf{z}), \sigma_t^2(\mathbf{z})) \tag{5}$$

where its mean and variance have the following closed-form expressions

$$\mu_t(\mathbf{z}) = \mathbf{k}_t^\top(\mathbf{z})(\mathbf{K}_t + \sigma_o^2\mathbf{I}_t)^{-1}\mathbf{y}_t \tag{6}$$

$$\sigma_t^2(\mathbf{z}) = \kappa(\mathbf{z}, \mathbf{z}) - \mathbf{k}_t^\top(\mathbf{z})(\mathbf{K}_t + \sigma_o^2\mathbf{I}_t)^{-1}\mathbf{k}_t(\mathbf{z}) \tag{7}$$

where $\mathbf{k}_t(\mathbf{z}) := [\kappa(\mathbf{z}_1, \mathbf{z}), ..., \kappa(\mathbf{z}_t, \mathbf{z})]^\top$.

Clearly, the performance of this GP predictor (6)-(7) highly hinges on the design of the kernel function $\kappa(\cdot,\cdot)$ over the input space. Accounting for both the continuous $\mathbf{x}_\tau$ for resource allocation and the categorical $\mathbf{c}_\tau$ for task offloading in the function input $\mathbf{z}_\tau$, as well as temporal variations across slots, three separate kernels are considered, which are $\kappa_x(\mathbf{x}_\tau, \mathbf{x}_{\tau'})$ over continuous inputs, $\kappa_c(\mathbf{c}_\tau, \mathbf{c}_{\tau'})$ over categorical inputs, and the temporal kernel $\kappa_{temp}(\tau, \tau')$.

Various kernel functions are available for continuous inputs; see [14]. A popular choice is the class of $Matérn$ kernels

$$\kappa_x^{MT}(\mathbf{x}_\tau, \mathbf{x}_{\tau'}) = \frac{2^{1-\nu}}{\Gamma(\nu)}\left(\frac{\sqrt{2\nu}\|\mathbf{x}_\tau - \mathbf{x}_{\tau'}\|}{l}\right)^\nu B_\nu\left(\frac{\sqrt{2\nu}\|\mathbf{x}_\tau - \mathbf{x}_{\tau'}\|}{l}\right) \tag{8}$$

with parameter $\nu > 0$ controlling the smoothness of the learning function. The smaller ν is, the less smooth the sought function is assumed to be. In (8), l is the characteristic lengthscale, B_ν is a modified Bessel function, and Γ is the gamma function. As for categorical variables, we follow [18] to adopt the kernel function $\kappa_c(\mathbf{c}_\tau, \mathbf{c}_{\tau'})$ as

$$\kappa_c(\mathbf{c}_\tau, \mathbf{c}_{\tau'}) = \frac{\omega}{M}\sum_{m=1}^{M}\mathbb{I}(c_\tau^m = c_{\tau'}^m) \tag{9}$$

where ω is the categorical kernel variance. To allow for a richer set of couplings between the continuous and categorical domains, a mixture of the sum and product compositions of the two kernels κ_x and κ_c is proposed for the kernel function $\kappa_{x,c}$ over continuous and categorical variables [18], i.e.,

$$\begin{aligned}\kappa_{x,c}([\mathbf{x}_\tau^\top, \mathbf{c}_\tau^\top]^\top, [\mathbf{x}_{\tau'}^\top, \mathbf{c}_{\tau'}^\top]^\top) =& (1-\lambda)[\kappa_c(\mathbf{c}_\tau, \mathbf{c}_{\tau'}) + \kappa_x(\mathbf{x}_\tau, \mathbf{x}_{\tau'})] \\ &+ \lambda\kappa_c(\mathbf{c}_\tau, \mathbf{c}_{\tau'})\kappa_x(\mathbf{x}_\tau, \mathbf{x}_{\tau'})\end{aligned} \tag{10}$$

where $\lambda \in [0, 1]$ weighs the contributions from the sum and product compositions of κ_c and κ_x.

To further capture the temporal variation of the black-box function φ due to the unknown system dynamics, the following temporal kernel function $\kappa_{temp}(\tau, \tau')$ is adopted based on [19]

$$\kappa_{temp}(\tau, \tau') = (1-\rho)^{\frac{|\tau-\tau'|}{2}} \tag{11}$$

where $\rho \in [0, 1]$ is the hyperparameter that controls the level of temporal dynamics in the learning function φ. The larger the value of ρ, the more frequently φ varies over time. In particular, when $\rho = 0$, $\kappa_{temp}(\tau, \tau') = 1$ for any (τ, τ'), thus inducing no dynamics in φ.

Henceforth, applying the product composition of $\kappa_{x,c}$ (10) and κ_{temp} (11) yields the overall kernel function given by

$$\kappa(\mathbf{z}_\tau, \mathbf{z}_{\tau'}) = \kappa_{temp}(\tau, \tau')\kappa_{x,c}([\mathbf{x}_\tau^\top, \mathbf{c}_\tau^\top]^\top, [\mathbf{x}_{\tau'}^\top, \mathbf{c}_{\tau'}^\top]^\top). \tag{12}$$

It can be observed that the temporal kernel imposes a scaling factor on $\kappa_{x,c}$ based on the time separation of any pair of inputs. This agrees well with intuition that inputs that are well separated in time (i.e., large $|\tau - \tau'|$) yield less correlated function values for $\rho \neq 0$.

B. Acquisition for $\mathbf{z}_{t+1}$ Based on GP Surrogate Model

Having available GP-based posterior function model (5) with the form of kernel function specified by (12) at slot t, one is ready to select the next decisions $\mathbf{z}_{t+1}$. Coping with both categorical and continuous variables, this is certainly a nontrivial task, but can fortunately be handled by relying on the MAB framework. Since the cardinality of the categorical variables is exponential with respect to the number M of

WDs, a scalable multi-agent MAB approach will be leveraged with each WD m acting as an agent simultaneously and independently determining its local task offloading decision $c_t^m \in \{0, 1, ..., N\}$. As the overall reward function in the resultant MAB framework does not follow any statistical distribution, it is more sensible to rely on the adversarial MAB framework and adopt as the action selection rule the well-known exponential-weight algorithm for exploration and exploitation (EXP3) [20]. Per slot t, EXP3 maintains an unnormalized weight vector $\mathbf{w}_t^m := [w_t^m(0), w_t^m(1), ..., w_t^m(N)]^\top$ for each WD m to guide the selection of its action. Next, we will delineate how each acquisition step of the time-varying BO selects categorical $\mathbf{c}_{t+1}$ and continuous $\mathbf{x}_{t+1}$ with the help of EXP3.

1) Acquisition for Categorical Task Offloading Decisions: Given $\mathbf{w}_t^m$ from the end of slot t, each agent m in EXP3 draws its action c_{t+1}^m randomly according to the probability vector $\mathbf{q}_t^m := [q_t^m(0), q_t^m(1), ..., q_t^m(N)]^\top$ with [20]

$$q_t^m(k) = \frac{(1-\gamma)w_t^m(k)}{\sum_{k'=0}^N w_t^m(k')} + \frac{\gamma}{N+1}, \forall k \in \{0, 1, ..., N\} \quad (13)$$

where $\gamma \in (0, 1]$ is the coefficient that balances *exploitation* given by the normalized weight in the first factor and *exploration* from the uniform probability in the second term. Specifically, by including the uniform distribution, EXP3 allows all $N + 1$ decisions to be explored per agent (WD) so as to get good reward estimates.

2) Acquisition for Analog-Amplitude Resource Allocation Decisions: With the categorical task offloading decisions $\mathbf{c}_{t+1}$ at hand, the analog-amplitude resource allocation decisions $\mathbf{x}_{t+1}$ are selected by finding the maximizer of the celebrated upper confidence bound (UCB)-based acquisition function as [21]

$$\mathbf{x}_{t+1} = \underset{0<\mathbf{x}\leq\mathbf{x}_{peak}}{\arg\max}\ u_{t+1}(\mathbf{x}|\mathcal{D}_t, \mathbf{c}_{t+1}, t+1) := \mu_t(\mathbf{x}, \mathbf{c}_{t+1}, t+1) + \sqrt{\zeta_{t+1}}\sigma_t^2(\mathbf{x}, \mathbf{c}_{t+1}, t+1) \quad (14)$$

where the coefficient $\zeta_{t+1} \geq 0$ nicely balances the exploitation and exploration that are signified by the posterior mean μ_t (6) and variance σ_t^2 (7), respectively. With closed-form expressions of μ_t and σ_t^2 at hand, one can readily solve (14) via off-the-shelf gradient-based solvers.

3) Weight Update in EXP3: Upon deploying $(\mathbf{c}_{t+1}, \mathbf{x}_{t+1})$ into the MEC system to yield the observed reward y_{t+1}, EXP3 capitalizes on the importance sampling rule to obtain an unbiased estimate of the reward value as

$$\hat{\varphi}_{t+1}^m(k) = \frac{y_{t+1}\mathbb{I}(c_{t+1}^m = k)}{q_t^m(k)}, \forall k \in \{0, 1, ..., N\}, m \in \mathcal{M}$$

based on which the corresponding weight is updated using the exponential rule as

$$w_{t+1}^m(k) = w_t^m(k) \exp\left(\frac{\gamma\hat{\varphi}_{t+1}^m(k)}{N+1}\right) = w_0^m(k) \exp\left(\frac{\gamma\sum_{\tau=1}^{t+1}\hat{\varphi}_\tau^m(k)}{N+1}\right), \forall k \in \{0, 1, ..., N\}, m \in \mathcal{M}.$$

It is evident that $w_{t+1}^m(k)$ summarizes the cumulative rewards up to slot $t+1$ for action k under WD m, and thus represents the effect of exploitation in (13).

IV. Simulation Results

In this section, numerical tests were conducted to evaluate the performance of the proposed BO approach for dynamic MEC management. In the multi-user multi-server MEC system with M WDs and N BSs, the time-varying wireless channel $h_t^{m,n}$ from WD m to BS n is modelled as Rician fading channel, where $K \geq 0$ is the Rician factor representing the ratio of the power in the LoS component to the power in the non-LoS component. The total average channel gain follows the free-space path loss model $|\bar{h}_t^{m,n}|^2 = A_d(\frac{3\times10^8}{4\pi\phi d_{m,n}})^{PL}, \forall t$, where $A_d = 4.11$ denotes the antenna gain, $\phi = 915$ MHz is the carrier frequency, $d_{m,n}$ represents the distance (measured by meters) between WD m and BS n, and $PL = 3$ signifies the pass loss exponent. In addition, the means of time-varying edge CPU frequencies $\{f_{c,t}^n\}_{n,t}$, task computational workloads $\{L_t^m\}_{m,t}$, and task input data sizes $\{I_t^m\}_{m,t}$ are 26 GHz, 125 Mcycles, and 1250 KBytes, respectively [2], [11]. Specifically, the generation rules follow [19] with parameter $\eta = 0.2$ adjusting the level of temporal dynamics in these system state variables.

Besides, the peak transmit power P_{peak} and computational frequency f_{peak} of each WD are equal to 100 mW and 10^8 Hz, respectively. To be aligned with commercial practise, the computing efficiency coefficient ξ of the WDs in (1) is chosen as $\xi = 10^{-26}$. We set the channel additive white Gaussian noise power $\sigma^2 = 10^{-10}$ W, and the bandwidth $W = 2$ MHz. The prior weights of the time delay and energy consumption cost of the WDs in (3) are set as $\beta_d = \beta_e = 0.5$.

For the proposed time-varying BO approach, the $Matérn$ kernel (8) with parameter $\nu = 5/2$ is adopted for the kernel κ_x over continuous variables. The weight λ regarding the sum and product kernel compositions in (10) is set to 0.5. The coefficients $\zeta_t = 2, \forall t$, in UCB-based acquisition rule (14). Unless otherwise stated, the other kernel hyperparameters are optimized by maximizing the log marginal likelihood every $\delta = 10$ slots via multi-started gradient descent. The performance measure of the competing methods is given by the notion of regret. By denoting the maximizer of φ_t as $(\mathbf{c}_t^*, \mathbf{x}_t^*)$, the instantaneous regret per slot t is $g_t := \varphi_t(\mathbf{c}_t^*, \mathbf{x}_t^*) - \varphi_t(\mathbf{c}_t, \mathbf{x}_t)$, based on which the cumulative and average regrets are denoted as $G_T := \sum_{t=1}^T g_t$ and $\bar{G}_T := G_T/T$, respectively. It is worth mentioning that $(\mathbf{c}_t^*, \mathbf{x}_t^*)$ are obtained by relying on explicit cost function in (P2) with known system state information. All the methods are run for 200 time slots and the average performances over 100 random repetitions are reported.

For performance comparison, three existing schemes are employed as baselines, namely, the MAB [20], bandit convex optimization (BCO) [4], and the conventional time-invariant BO approach [12]. Since MAB can only cope with discrete decision variables, we discretized the analog-amplitude resource allocation variables into 5 levels and then adopted the multi-agent EXP3 method [20] for learning. In BCO, the analog-amplitude resource allocation variables are obtained by constructing gradient estimates using evaluated function values, while the discrete offloading variables are still sought

based on MAB as in the proposed BO approach. Besides, time-invariant BO method neglects temporal information in MEC systems.

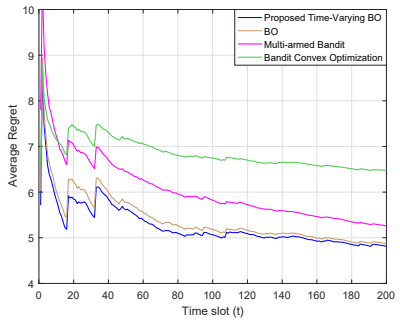

Fig. 1: Comparison of average regret under the 2-BS and 2-WD MEC system.

With properly selected temporal kernel hyperparameter, the average regret curves of all the competing approaches are presented in Fig. 1 for the 2-BS and 2-WD MEC system with $[d_{1,1}, d_{1,2}, d_{2,1}, d_{2,2}] = [20, 13, 15, 18]$ and $K = 4$. Specifically, the temporal kernel hyperparameter in the time-varying BO approach is chosen as $\rho = 0.048$. As shown in Fig. 1, our proposed time-varying BO approach outperforms the three benchmarks, namely, time-invariant BO, MAB, and BCO, by around 1.21%, 8.51% and 25.72% in average regret after 200 time slots. This suggests the benefits of adapting temporal information-aided Bayesian approach to the black-box optimization with both categorical (i.e., task offloading) and analog-amplitude (i.e., resource allocation) variables.

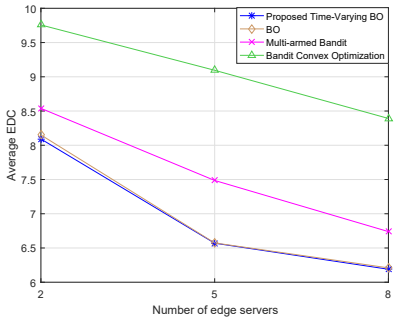

Fig. 2: Impact of MEC network size on average energy-delay cost.

Moreover, fixing the number M of WDs as 2, the average EDC over slots is plotted as a function of the number N of BSs for all the competing methods in Fig. 2. Apparently, the proposed BO approach achieves lower average EDC than the other three baselines. Additionally, the average EDC of all the methods decreases as the network size grows by better exploiting the diverse computing capacities and channel conditions of the edge servers.

V. Conclusion

BO for dynamic MEC management was studied in this paper. Different from prior works in time-varying MEC systems, the focus was online joint optimization of discrete task offloading decisions and analog-amplitude resource allocation strategies by minimizing the EDC using only bandit observations at queried points. Specifically, by exploiting temporal information, we developed novel BO approach that incorporates the strength of the MAB framework. Numerical tests under different MEC network sizes demonstrated the effectiveness of the proposed BO approach.

References

[1] 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, Fourthquarter 2017.

[2] C. You, K. Huang, and H. Chae, "Energy efficient mobile cloud computing powered by wireless energy transfer," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 5, pp. 1757–1771, May 2016.

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

[4] A. D. Flaxman, A. T. Kalai, and H. B. McMahan, "Online convex optimization in the bandit setting: Gradient descent without a gradient," in *Proc. ACM SODA, Vancouver, BC, Canada*, Jan. 2005, pp. 385–394.

[5] A. Agarwal, O. Dekel, and L. Xiao, "Optimal algorithms for online convex optimization with multi-point bandit feedback." *Proc. Annual Conf. Learning Theory*, pp. 28–40, 2010.

[6] T. Chen and G. B. Giannakis, "Bandit convex optimization for scalable and dynamic IoT management," *IEEE Internet of Things Journal*, vol. 6, no. 1, pp. 1276–1286, 2019.

[7] B. Wu, T. Chen, W. Ni, and X. Wang, "Multi-agent multi-armed bandit learning for online management of edge-assisted computing," *IEEE Trans. Commun.*, vol. 69, no. 12, pp. 8188–8199, 2021.

[8] B. Li, T. Chen, and G. B. Giannakis, "Secure mobile edge computing in IoT via collaborative online learning," *IEEE Trans. Signal Process.*, vol. 67, no. 23, pp. 5922–5935, 2019.

[9] Y. Sun, X. Guo, J. Song, S. Zhou, Z. Jiang, X. Liu, and Z. Niu, "Adaptive learning-based task offloading for vehicular edge computing systems," *IEEE Trans. Veh. Technol.*, vol. 68, no. 4, pp. 3061–3074, 2019.

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

[11] J. Yan, S. Bi, Y. J. Zhang, and M. Tao, "Optimal task offloading and resource allocation in mobile-edge computing with inter-user task dependency," *IEEE Trans. Wireless Commun.*, vol. 19, no. 1, pp. 235–250, 2019.

[12] P. I. Frazier, "A tutorial on Bayesian optimization," *arXiv:1807.02811. [Online]. Available: http://arxiv.org/abs/1807.02811*, 2018.

[13] Q. Lu, K. D. Polyzos, B. Li, and G. B. Giannakis, "Surrogate modeling for Bayesian optimization beyond a single Gaussian process," *arXiv preprint arXiv:2205.14090*, 2022.

[14] C. E. Rasmussen and C. K. Williams, *Gaussian processes for Machine Learning*. MIT press Cambridge, MA, 2006.

[15] Q. Lu, G. Karanikolas, Y. Shen, and G. B. Giannakis, "Ensemble Gaussian processes with spectral features for online interactive learning with scalability," *Proc. Int. Conf. Artif. Intel. and Stats.*, pp. 1910–1920, 2020.

[16] Q. Lu, G. V. Karanikolas, and G. B. Giannakis, "Incremental ensemble Gaussian processes," *IEEE Trans. Pattern Anal. Mach. Intel.*, 2022.

[17] K. D. Polyzos, Q. Lu, and G. B. Giannakis, "Ensemble Gaussian processes for online learning over graphs with adaptivity and scalability," *IEEE Trans. Sig. Process.*, 2021.

[18] B. Ru, A. S. Alvi, V. Nguyen, M. A. Osborne, and S. J. Roberts, "Bayesian optimisation over multiple continuous and categorical inputs," *Proc. Int. Conf. Mach. Learn.*, 2020.

[19] I. Bogunovic, J. Scarlett, and V. Cevher, "Time-varying Gaussian process bandit optimization," *Proc. Int. Conf. Artif. Intel. and Stats.*, pp. 314–323, 2016.

[20] A. Peter, C.-B. Nicolo, F. Yoav, and R. E. Schapire, "The nonstochastic multiarmed bandit problem," *SIAM J. on Computing*, pp. 48–77, 2002b.

[21] N. Srinivas, A. Krause, S. M. Kakade, and M. W. Seeger, "Information-theoretic regret bounds for Gaussian process optimization in the bandit setting," *IEEE Trans. Inf. Theory*, vol. 58, no. 5, pp. 3250–3265, 2012.