Online Learning of Image Recognition Task Offloading Policies over Wireless Links

Iordanis Koutsopoulos
Department of Informatics
Athens University of Economics and Business
Athens, Greece

***Abstract*—We study the problem of online learning of optimal offloading policies for image processing tasks, for minimizing a cost that is weighted sum of transmit energy and object recognition error rate. A mobile node generates image processing tasks that involve object recognition. There exist three options: (*i*) transmit the image to a remote server for processing with a deep-learning (DL) model, (*ii*) process locally with a simpler model, (*iii*) apply a lightweight, error-prone technique for object detection, and if objects are detected, then send image to the server. The proper offloading decision requires knowledge of the transmit energy cost and object recognition error rate for each option. However, these processes are non-stationary due to unpredictable object occurrence, mobility and propagation dynamics, and they depend on the object inference result which is unknown at decision time. We cast the problem as an adversarial multi-armed bandit, in which the EXP3 algorithm achieves sublinear regret. For the constrained problem, we propose an algorithm that extends EXP3 and achieves good regret in the objective and constraint, thus asymptotically learning the optimal static randomized offloading policy, while satisfying the error constraint. Performance is validated via numerical experiments informed by real-life object recognition measurements and models.**

I. Introduction

Real-time image processing arises in services such as video surveillance, augmented reality (AR), and unmanned-air-vehicle (UAV) video processing for infrastructure monitoring, agriculture, security or smart-city services. In these scenarios, a video stream is captured through a camera mounted on a mobile phone, drone or other device [1]. The goal is to process video frames and *recognize* one or more objects by solving a classification problem. For example, when a drone performs infrastructure (e.g railway) monitoring, certain obstacles on rails need to be recognized [2]. In crowd surveillance, individuals carrying weapons or moving suspiciously should be identified. In AR applications, buildings, monuments or other sites should be identified so that appropriate meta-data are superimposed on the image.

The consumed energy at the node that captures the video is critical and depends on the task offloading policy, among other factors. A first option is to offload the image to a remote edge or cloud server for processing through a sophisticated deep-learning (DL) model, e.g. a deep Convolutional Neural Network (CNN). Energy is consumed for image transmission, depending on the instantaneous distance between the mobile transmitter and the edge server receiver, and on channel conditions. The latter dictate the required transmit rate, which affects transmission duration and power, and thus energy.

A second option is to process the image *locally* through DL models tailored to mobile devices such as Tiny-YOLO [3] or Tensorflow Lite [4]. For example, Tiny-YOLO uses a shallow convolutional structure that allows inference with affordable computational burden and energy consumption, at the cost of reduced inference accuracy. Finally, a third option is to run a light-weight brute-force but error-prone technique for object detection, and if objects of interest are detected with some confidence, then the image is offloaded to the remote server for processing. In this option, the image is transmitted remotely only when needed, however errors may sometimes occur.

There exist two sources of *uncertainty*. The first one concerns the *transmit energy* for image transmission. The transmit energy random process may be hard or impossible to characterize statistically. Unpredictable channel dynamics and the surrounding environment may lead to non-stationarity. For instance, for a flying drone that captures a video stream, a video frame may need different amounts of transmit energy in consecutive slots, due to arbitrary drone movements, rapidly time-varying wireless propagation conditions, and different receiving access points. The second type of uncertainty concerns the *presence or absence of objects of interest* in an image. Indeed, it is not possible to know at the time of image offloading decision whether an object of interest will be inferred to be present or absent in the image. Objects of interest may appear in consecutive frames in an arbitrary manner, and it is not reasonable to model object inference (absence/presence) as a stationary binary random process.

If instantaneous values of transmit energy and object inference *were* available at decision time, the instantaneous energy costs and error rates would be known for the three options above, so that the best option could be chosen. Further, if the processes of transmit energy and object inference could be characterized statistically, for example through stationary distributions, a dynamic policy based on Lyapunov stochastic optimization would use their instantaneous realizations to find an optimal online offloading policy [5]. However, these processes may be non-stationary, and in fact the object inference result is not even known at the time of the offloading decision.

In this paper, we consider the scenario where a mobile node generates image processing tasks for recognizing objects of interest. We study the problem of *learning* the optimal

offloading policy for image processing tasks, in terms of minimizing a cost that is a weighted sum of transmit energy and object recognition error. The difficulty stems from the unknown, arbitrary, non-stationary dynamics about object occurrence and about transmit energy. The dynamics are assumed to be arbitrarily chosen by an adversary and are revealed to the learner *after* the offloading decision. This setting is unique to offloading of image processing tasks. We seek an online learning algorithm that learns an offloading policy whose cost is close to that of the optimal static offloading policy that has knowledge of the entire dynamics path. The contributions of our work to the literature are as follows:

- We cast the problem of learning a policy that minimizes the weighted cost above, as an adversarial bandit, where arms correspond to the options: (*i*) transmit the image to a remote server for processing with a DL model, (*ii*) process the image locally with a simpler DL model, (*iii*) use a lightweight, error-prone technique for object detection, and if objects are detected, then send the image to the server. The EXP3 algorithm solves this problem.
- We propose an algorithm that extends EXP3 for the learning problem of minimizing transmit energy, and a constraint on total expected error rate. The algorithm relies on Lagrangian gradient ascent and constraint violation to influence the probabilities of choosing among the three options, and it achieves desirable regret.

In section II, we discuss related work. In section III, we describe the model and state the problem and algorithm for the combined-cost problem. In section IV, we consider the constrained problem and present our algorithm. Section V presents numerical results, and section VI concludes the paper.

II. Related work

Offloading. Computation offloading has been an active topic in recent years, and Lyapunov optimization is one of the mathematical tools to tackle it. In [6], each user may either perform the computation locally or send the task remotely through a channel and receive interference from other users. This work does not include the image processing aspect, which changes the setting. The authors in [7] study joint dynamic offloading, transmit power, and CPU cycle control in the presence of wireless channel and energy harvesting dynamics, so as to minimize execution latency and task drop cost. The work [8] studies offloading of computer vision tasks and decides whether to initiate task pre-processing prior to task offloading so as to save energy to the expense of accuracy. The authors in [9] formulate the optimal computation offloading problem as a Markov Decision Process so as to minimize a convex combination of energy and latency.

Multi-armed bandits and the EXP3 algorithm. In online learning with a discrete set of choices, the learner at each round makes a choice out of available ones, and it receives some feedback in response. When the learner gets to see the resulting losses (costs) for *all* choices after making a certain choice, the problem is called *d-expert selection*. At each time t, we pick an expert and observe the costs of all experts. The problem is to find an expert selection policy that has sublinear regret over the time horizon, where the regret measures the difference between the cumulative cost of our policy and that of the policy that always selects the best action (expert) in hindsight. The randomized *Exponentially Weighted Average* (EWA) (or *Exponentiated Gradient*, EG) algorithm achieves $O(\sqrt{T \log d})$ regret [10]. EWA selects at time t an expert with probability inversely proportional to an exponential function of the cumulative loss of the expert up to time $t-1$. Thus, it penalizes experts according to observed losses by reducing the probability of selection for experts with high losses.

In multi-armed bandits, there exists again a finite set of decision options, the arms. At each round, the learner chooses one arm, possibly through a randomized policy and gets to see *only* the cost of that arm and not costs of other arms. The cost may be stochastic and drawn from an unknown probability distribution, or it may be arbitrarily chosen by an adversary. The goal is to achieve sublinear regret with time, for not pulling the best arm. For stochastic bandits, the Universal Confidence Bound (UCB) algorithm attains $O(\log T)$ regret [11]. For non-stochastic (adversarial) bandits, the EXP3 algorithm achieves $O(\sqrt{T})$ regret [12]. EXP3 is based on EWA and selects at each time t an expert a^t with probability $p^t_{a^t}$ inversely proportional to an exponential function of the cumulative loss of the expert up to t. However, there are two differences in EXP3 compared to EWA. First, an unbiased estimate of the loss vector is formed, via vector $\tilde{\boldsymbol{\ell}}^t$ with components equal to $\tilde{\ell}^t_j = \ell^t_j / p^t_j$, if $j = a^t$, and 0 otherwise. This is an unbiased estimate of the loss at time t, since for each j, it is $\mathbb{E}[\tilde{\ell}^t_j \,|\, \mathbf{p}^t] = p^t_j \cdot \frac{\ell^t_j}{p^t_j} + (1 - p^t_j) \cdot 0 = \ell^t_j$. Second, *only* the cumulative loss of the selected expert is updated each time, based on the loss estimate above. Multiarmed bandits are used in various settings, see e.g. [13] and references therein.

Online learning under constraints. A related thread is regret minimization under constraints that need to be satisfied in the long run and may be chosen by an adversary. If constraint violation is sublinear in T, constraints are satisfied as $T \to \infty$. The work [14] proposes a modification of Online Gradient Descent that attains $O(\sqrt{T})$ regret and $O(T^{3/4})$ constraint violation, while the work [15] achieves a cumulative regret plus constraint violation of $O(T^{2/3})$. The work [16] achieves similar bounds for adversarial contextual bandits, where the learner observes some context, it takes a decision and then observes a loss conditioned on decision and context.

Our main differentiating point from the state of the art is the adversarial bandit formulation for the offloading problem, specialized to image processing tasks. The new twist emerges because of the non-stationary dynamics of object occurrence in the image, and of energy consumption, which are hitherto not addressed, and they are unique to image processing tasks.

III. Model and problem statement

A. Model

We assume that the mission of the learner is to recognize at each image one or more objects (labels) from a set $\mathcal{K} =$

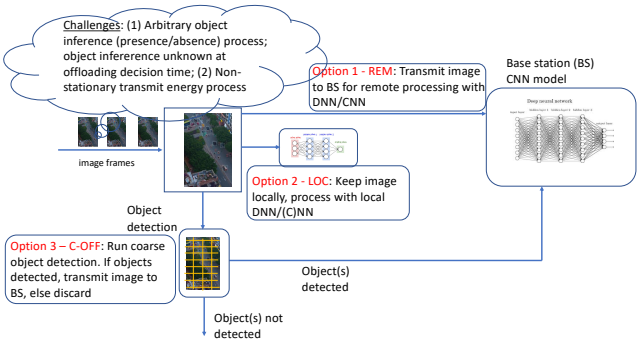

Figure 1. The three offloading options in our setting.

$\{1, \ldots, K\}$ of K labels, where a label refers to a distinct type of object. Time is slotted. At each time slot t, a node generates a video frame for processing, and each frame constitutes an image processing task. For each task at time t, the node should choose among the following options shown in Fig. 1.

Option 1: Remote processing (REM). The first option is to offload the task to a remote edge server by transmitting the image through the wireless channel to the access point (AP) that hosts the server. Given an instantaneous transmitter-receiver link gain G_t at time t, which depends on wireless propagation, the surrounding environment, and the distance to the AP receiver, a minimum amount of transmit power, P_t is needed for successful image reception. Assuming noise-limited channel, the transmit rate is $r_t = \log(1+\tilde{G}_t P_t)$, where $\tilde{G}_t = G_t/\sigma^2$, and σ^2 is the noise variance. If the size of the video frame is S bits, transmit energy is $E_t^R = E_t^{tr} = P_t \cdot \frac{S}{r_t}$.

Let p_k^R, q_k^R denote the known values of false positive and false negative rates for object k, for $k = 1, \ldots, K$, as a result of a sophisticated deep-learning (DL) inference model at the server. These are the misclassification probabilities for label k, and they are assumed to be known if the DL inference model is applied on a test dataset.

Define the binary parameter $b_{k,t} \in \{1, 0\}$ that denotes whether label k is found to occur in the image at time t or not, based on the DL model. The probability of error at time t for label k at the remote server is defined as:

$$\beta_{k,t}^R = (1 - b_{k,t}) q_k^R + b_{k,t} p_k^R \,. \tag{1}$$

We stress that $b_{k,t}$ shows the observed inference outcome of the DL model, and not the ground truth i.e. whether object k truly exists in the image, which cannot be assumed to be known. Based on the observed value of $b_{k,t}$, and the known values of p_k^R, q_k^R, one can compute the probability of error as above. However, the challenge is that the value of $b_{k,t}$ cannot be observed at the time of the offloading decision, as explained in the sequel. The overall average probability of error at time t at the remote server is $\beta_t^R = \frac{1}{K}\sum_{k=1}^K \beta_{k,t}^R$.

Option 2: Local Processing (LOC). The second option is to process the image locally at the mobile node with a local, computationally "light" version of a Machine Learning (ML) model and framework, such as Tiny-YOLO or Tensorflow Lite. Clearly, the lower computational burden of such a model comes with lower accuracy.

Let p_k^L, q_k^L denote the known values of false positive and false negative rates for object k for the local ML model. The energy consumed for local model execution is known and fixed, $E_t^L = E^L$. Similarly to REM, the probability of of error at time t for label k at the remote server is

$$\beta_{k,t}^L = (1 - b_{k,t}) q_k^L + b_{k,t} p_k^L \,, \tag{2}$$

while the average probability of error at time t over all labels with the local ML model is $\beta_t^L = \frac{1}{K}\sum_{k=1}^K \beta_{k,t}^L$.

Option 3: Conditional offloading (C-OFF). The third alternative lies in between REM and LOC. The mobile node may use a lightweight, fast-track technique to check whether objects exist. If it infers existence of labels of interest, the image is offloaded to the server for further processing, otherwise it is discarded.

There exist various techniques that use on-device processing and video frame filtering so that only frames with detected objects are transmitted. For example, pixel-based techniques are discussed in [17], through which blurry frames are dropped before image offloading. The work [18] proposes a system that performs a coarse visual pre-processing at very low power for each frame in order to detect an object. Further, in [19], the authors propose image preprocessing that performs early frame discard based on deep neural networks for object detection.

We assume that the lightweight object detection technique consumes energy $E^0 < E^L$, thus it is more energy-efficient than the LOC option. Option 3 saves energy since only images with detected objects are offloaded to the remote server. Let p_k^C, q_k^C denote the known false positive and false negative rates for label (object) k for the lightweight object detection algorithm. The probability of error for label k at time t is

$$\beta_{k,t}^C = b_{k,t}(1 - p_k^C)\beta_{k,t}^R + b_{k,t} p_k^C + (1 - b_{k,t}) q_k^C \,, \tag{3}$$

where the first term refers to the case when an object of interest is correctly inferred to be present by the object detection algorithm, and thus the frame is sent to the remote server and is processed with the DL model there. The overall average probability of error is $\beta_t^C = \frac{1}{K}\sum_{k=1}^K \beta_{k,t}^C$.

The consumed energy at time t for the C-OFF option is

$$E_t^C = E^0 + \min\left\{1, \sum_{k=1}^K b_{k,t}\right\} E_t^{tr}, \tag{4}$$

where the second term means that the image is offloaded if at least one object of interest is inferred to be present in it.

In terms of error, we assume that the ML algorithm in REM is more sophisticated than that in LOC. Further, C-OFF is more error-prone than LOC. Thus, $p_k^C > p_k^L > p_k^R$ and $q_k^C >$

$q_k^L > q_k^R$, and thus $\beta_{k,t}^C > \beta_{k,t}^L > \beta_{k,t}^R$ for $k = 1, \ldots, K$, so that REM is overall the most accurate option, while C-OFF is the less accurate one for $b_{k,t} \in \{0, 1\}$.

In terms of energy consumption, depending on channel conditions, E_t^R may be higher or lower than E^L, and E_t^C may be higher or lower than E^L.

B. Problem statement and the EXP3 algorithm

The learner needs to choose among the following options: (*i*) offloading the image to a remote server, which is the most accurate option but may also be the most energy-consuming one, depending on channel conditions; (*ii*) local computation, which has moderate accuracy and may be more energy-efficient than the first option; (*iii*) conditional offloading, which could be the most energy-efficient but is also the most error-prone option. The tradeoff is that C-OFF saves transmit energy by discarding frames in which no objects are inferred as present; however it does so at the expense of possible errors.

The relative ordering of the three options in terms of energy cost depends on instantaneous values of energy consumption $\{E_t^{tr}\}$ and object presence/absence inference $\{b_{k,t}\}$, for $k = 1, \ldots, K$. Each of these options comes with an error cost as well. The main challenge is that the process $\{b_{k,t}\}$ is non-stationary, and it is not possible to know $b_{k,t}$ (i.e. whether an object will be inferred to be present or absent) at the time of the offloading decision, since this refers to the inference result of the DL model in the image. Further, the transmit energy process E_t^{tr} may be non-stationary as well, due to non-stationarity of the wireless channel gain and node mobility, and E_t^{tr} may also be unknown at decision time.

Since processes $\{E_t^{tr}\}$ and $\{b_{k,t}\}$, $k = 1, \ldots, K$ are non-stationary, we cast the problem of selecting the appropriate offloading policy at each time t as an adversarial bandit.

Define the real-valued variables $x_t(1)$, $x_t(2)$, $x_t(3)$ that denote the probability that the controller chooses REM, LOC or C-OFF respectively at time t, with $0 \leq x_t(1), x_t(2), x_t(3) \leq 1$. Let $\mathbf{x}_t = (x_t(1), x_t(2), x_t(3))$. At each time slot t, it is $\sum_{i=1}^3 x_t(i) = 1$.

For each time t, we define the energy cost parameters

$$c_t(i) = \begin{cases} E_t^R & \text{if } i = 1, \\ E_t^L & \text{if } i = 2, \\ E_t^C & \text{if } i = 3\,. \end{cases} \tag{5}$$

and let $\mathbf{c}_t = (c_t(1), c_t(2), c_t(3))$. We normalize $c_t(i)$ in $[0, 1]$ for $i = 1, 2, 3$. Further, we define the error cost parameters

$$\beta_t(i) = \begin{cases} \beta_t^R & \text{if } i = 1, \\ \beta_t^L & \text{if } i = 2, \\ \beta_t^C & \text{if } i = 3\,, \end{cases} \tag{6}$$

and let $\boldsymbol{\beta}_t = (\beta_t(1), \beta_t(2), \beta_t(3))$. The expected consumed energy at time slot t is

$$E_t(\mathbf{x}_t) = \mathbf{c}_t^T \mathbf{x}_t\,, \tag{7}$$

while the expected probability of error at time t is

$$B_t(\mathbf{x}_t) = \boldsymbol{\beta}_t^T \mathbf{x}_t\,. \tag{8}$$

We define the weighted sum of energy and error costs as

$$F(\mathbf{x}_t) = E_t(\mathbf{x}_t) + \omega B_t(\mathbf{x}_t) = \boldsymbol{\gamma}_t^T \mathbf{x}_t\,, \tag{9}$$

where $\omega > 0$ is a fixed weight factor, and $\boldsymbol{\gamma}_t = \boldsymbol{\beta}_t + \omega \mathbf{c}_t$.

Given a time horizon T, an online offloading policy $(\mathbf{x}_1, \ldots, \mathbf{x}_T)$ decides at each slot t the probability distribution $\mathbf{x}_t$ with which to select each option (REM, LOC or C-OFF).

Let $\mathbf{x}^*$ be the optimal offline static policy i.e., the one which minimizes the total weighted cost by having full knowledge of processes $\{\mathbf{c}_t\}$ and $\{\boldsymbol{\beta}_t\}$ for all $t = 1, \ldots, T$. Namely,

$$\mathbf{x}^* = \arg\min_{\mathbf{x} \in \mathcal{X}} \sum_{t=1}^T \boldsymbol{\gamma}_t^T \mathbf{x}\,, \tag{10}$$

where $\mathcal{X} = \{\mathbf{x} : \sum_{i=1}^3 x(i) = 1 \,\forall\, t\}$. The learning problem is:

$$\min_{\mathbf{x}_1, \ldots, \mathbf{x}_T} \frac{1}{T} \mathrm{Reg}_T(\mathbf{x}_1, \ldots, \mathbf{x}_T) = \min_{\mathbf{x}_1, \ldots, \mathbf{x}_T} \frac{1}{T} \sum_{t=1}^T (F_t(\mathbf{x}_t) - \boldsymbol{\gamma}_t^T \mathbf{x}^*)\,. \tag{11}$$

At each time t, the energy $\{c_t(i)\}$ and error rate $\{\beta_t(i)\}$ for a specific offloading action $i \in \{R, L, C\}$ are revealed to the learner as the result of instantaneous realizations of the energy process $\{E_t^{tr}\}$ and the object presence/absence inference process $\{b_{k,t}\}$ for objects $k \in \mathcal{K}$, *after* the action is selected. The problem can be mapped to an adversarial bandit with three arms, R (REM), L (LOC) or C (C-OFF). The EXP3 algorithm solves the problem with the cumulative cost of arm $i = 1, 2, 3$ given by $\hat{\gamma}_t(i) = \hat{c}_t(i) + \omega \hat{\beta}_t(i)$, and it achieves a regret upper bound of $O(\sqrt{6T \log 3})$.

IV. Constrained problem

Consider now the problem of minimizing the energy cost, subject to a constraint on long-term average error rate:

$$\frac{1}{T} \sum_{t=1}^T B_t(\mathbf{x}_t) \leq \theta \tag{12}$$

where $\theta \in (0, 1)$ is a pre-specified threshold.

Let $\mathbf{x}^*$ be the optimal offline static policy i.e., the one which minimizes the total energy cost by having full knowledge of processes $\{\mathbf{c}_t\}$ and $\{\boldsymbol{\beta}_t\}$ for all $t = 1, \ldots, T$. Namely,

$$\mathbf{x}^* = \arg\min_{\mathbf{x} \in \mathcal{Y}} \sum_{t=1}^T \mathbf{c}_t^T \mathbf{x}\,, \tag{13}$$

where $\mathcal{Y} = \{\mathbf{x} : \boldsymbol{\beta}_t^T \mathbf{x} \leq \theta,\, \forall\, t,\ \text{and}\ \sum_{i=1}^3 x_t(i) = 1\}$.

Formally, the learning problem to solve is:

$$\min_{\mathbf{x}_1, \ldots, \mathbf{x}_T} \frac{1}{T} \mathrm{Reg}_T(\mathbf{x}_1, \ldots, \mathbf{x}_T) = \min_{\mathbf{x}_1, \ldots, \mathbf{x}_T} \frac{1}{T} \sum_{t=1}^T (E_t(\mathbf{x}_t) - \mathbf{c}_t^T \mathbf{x}^*) \tag{14}$$

subject to (12).We are interested in a policy $\mathbf{x}_1, \ldots, \mathbf{x}_T$ so that

$$\mathrm{Reg}_T^* = \min_{\mathbf{x}_1, \ldots, \mathbf{x}_T} \mathrm{Reg}_T(\mathbf{x}_1, \ldots, \mathbf{x}_T) = o(T)\,, \tag{15}$$

so that $\lim_{T \to \infty} \frac{1}{T} \mathrm{Reg}_T^* = 0$.

We seek a no-regret policy for which the difference (on average) from the optimal static policy that knows in advance the energy cost sequence $\{\mathbf{c}_t\}_{t=1,\ldots,T}$ and error cost sequence $\{\boldsymbol{\beta}_t\}_{t=1,\ldots,T}$ goes to zero as $T \to \infty$. The policy should fulfill constraint (12) on average and $\sum_{i=1}^{3} x_t(i) = 1$ at each t.

A remark is in place here. The optimal static policy is computed over $\mathcal{Y}$, which requires that the constraint is satisfied at each slot, rather than over set $\mathcal{Y}' = \{\mathbf{x} : \sum_{t=1}^{T} B_t(\mathbf{x}) \leq T\theta\}$, which would require that the constraint is satisfied on average. According to [16, Proposition 2.1], if the learner competes against the optimal policy computed over set $\mathcal{Y}'$, the regret grows at least linearly. Hence, we assume that the learner competes against an optimal policy computed over $\mathcal{Y}$.

The amount of average constraint violation is

$$\frac{1}{T}\text{Viol}_T(\mathbf{x}_1, \ldots, \mathbf{x}_T) = \frac{1}{T}\sum_{t=1}^{T}(B_t(\mathbf{x}_t) - T\theta)\,. \tag{16}$$

We would also like to have,

$$\text{Viol}_T^* = \min_{\mathbf{x}_1,\ldots,\mathbf{x}_T} \text{Viol}_T(\mathbf{x}_1, \ldots, \mathbf{x}_T) = o(T)\,, \tag{17}$$

so that $\lim_{T\to\infty} \frac{1}{T}\text{Viol}_T^* = 0$.

A. Proposed algorithm

We propose an algorithm to address the constrained problem. First, note that a constrained optimization problem of the form

$$\min_{\mathbf{x}} \sum_{t=1}^{T} f_t(\mathbf{x}) \text{ subject to: } \sum_{t=1}^{T} g_t(\mathbf{x}) \leq 0\,, \tag{18}$$

is equivalent to the convex-concave optimization problem:

$$\min_{\mathbf{x}} \max_{\lambda>0} \left(\sum_{t=1}^{T} f_t(\mathbf{x}) + \lambda g_t(\mathbf{x}) \right)\,, \tag{19}$$

where λ is the Lagrange multiplier associated with the constraint. For each t, define the regularized Lagrangian [14],[15]:

$$L_t(\mathbf{x}, \lambda) = E_t(\mathbf{x}) + \lambda(B_t(\mathbf{x}) - \theta)) - \frac{\eta\delta}{2}\lambda^2\,, \tag{20}$$

where $\eta > 0$ is the learning rate, and $\delta > 0$ is a constant whose role will be detailed shortly. Regularization prevents large values of the Lagrange multiplier.

We propose the *Constrained-EXP3* (ConEXP3) algorithm which we refer to as Algorithm 1. ConEXP3 extends EXP3 to a constrained problem. The main difference from EXP3 is the Lagrange multiplier update in Step 11, that is, the dual update of gradient ascent with respect to dual variable λ on the sequence of functions $\{L_t(\mathbf{x}_t, \lambda)\}$. A large value of Lagrange multiplier, and large values of energy cost and error rate estimates for action i will result in reduced probability of selection of i in the next round. Steps 8-9 stand for the EWA descent update with respect to primal variables $\mathbf{x}$ on the sequence of Lagrangian functions $\{L_t(\mathbf{x}, \lambda_t)\}$. The policy update in Step 8 resembles that in EXP3, but the Lagrange multiplier λ_t (which captures the amount of penalty for violating the constraint) and the error rate estimate are factored in the update as well. The policy $\mathbf{y}_{t+1}(i)$ is discounted by $\exp\left[-\eta\left(\hat{c}_t(i) + \lambda_t\hat{\beta}_t(i)\right)\right]$ here, while in EXP3, it is discounted by $\exp\left[-\eta\hat{\gamma}_t(i)\right]$, where $\hat{\gamma}_t(i) = \gamma_t(a_t)/x_t(a_t)$ if $i = a_t$, and 0 else, and $\boldsymbol{\gamma}_t$ is defined in subsection III.B.

In the algorithm above, the ℓ_1 norm of vector $x = (x_1, \ldots, x_d)$ is $\|\mathbf{x}\|_1 = \sum_{i=1}^{d} |x_i|$, while $\mathbf{1}$ is the vector of ones. In the sequel, we show numerically that ConEXP3 performs well in regret and constraint violation.

Algorithm 1 Algorithm ConEXP3.

1: **input:** Parameter $\eta > 0$, the learning rate of the algorithm. Parameter $\delta > 0$.
2: **output:** A sequence of vectors $\mathbf{x}_1, \ldots, \mathbf{x}_T$.
3: **Initialization:** $\mathbf{y}_1 = \mathbf{1}$, $\mathbf{x}_1 = \frac{\mathbf{y}_1}{\|\mathbf{y}_1\|_1} = \frac{1}{3}\mathbf{1}$.
4: **for** $t = 1, \ldots, T$
5: Draw an arm $a_t \in \{1, 2, 3\}$ according to prob. distr. $\mathbf{x}_t$. Pull arm a_t.
6: Observe $b_{k,t}$, $k \in \mathcal{K}$, and E_t^{tr}, and thus find energy cost $c_t(a_t)$. Estimate energy cost gradient $\nabla_{\mathbf{x}_t} E_t(\mathbf{x}_t))$ as the vector with components,

$$\hat{c}_t(i) = \begin{cases} \frac{c_t(a_t)}{x_t(a_t)} & \text{if } i = a_t, \\ 0 & \text{else.} \end{cases} \tag{21}$$

7: Compute incurred error rate $\beta_t(a_t)$. Estimate error rate gradient $\nabla_{\mathbf{x}_t} B_t(\mathbf{x}_t))$ as the vector with components,

$$\hat{\beta}_t(i) = \begin{cases} \frac{\beta_t(a_t)}{x_t(a_t)} & \text{if } i = a_t, \\ 0 & \text{else.} \end{cases} \tag{22}$$

8: **if** $a_t = i$, **then**

$$\mathbf{y}_{t+1}(i) = \mathbf{y}_t(i) \exp\left[-\eta\left(\hat{c}_t(i) + \lambda_t\hat{\beta}_t(i)\right)\right], \tag{23}$$

9: **else if** $a_t \neq i$, **then** $\mathbf{y}_{t+1}(i) = \mathbf{y}_t(i)$.
10: **Project** $\mathbf{x}_{t+1} = \frac{\mathbf{y}_{t+1}}{\|\mathbf{y}_{t+1}\|}$.
11: **Update** Lagrange multiplier λ_{t+1} according to:

$$\lambda_{t+1} = \max\left\{0, \lambda_t + \eta[\hat{\beta}_t(a_t)x_t(a_t) - \theta - \delta\eta\lambda_t]\right\} \tag{24}$$

12: **end for**

V. Numerical evaluation

We consider the scenario of a flying drone that captures video and has a classification problem with $K = 1$ object. The DL model for REM is in the VGG family of DL models with typical accuracy values between $92-94\%$ [20], hence we take error rates $p^R = 0.06$ and $q^R = 0.08$. The DL model for LOC is AlexNet, with a typical accuracy between $80 - 82\%$ [21], hence we take $p^L = 0.18$ and $q^L = 0.2$.

For C-OFF, we consider a method similar in flavor to the Early Discard "weak" object detector proposed in [19]. We take a conservative view and set $p^C = q^C = 0.35$.

The transmitter moves so that instantaneous distance to an edge server varies continuously between $40 - 600$ meters, and transmit rate is between 48 and 1 Mbps. Transmit power is

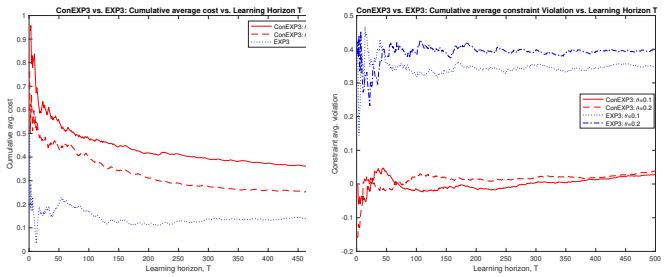

Figure 2. (Left): Cumulative average cost for ConEXP3 and EXP3. (Right):Average amount of constraint violation for ConEXP3 and EXP3.

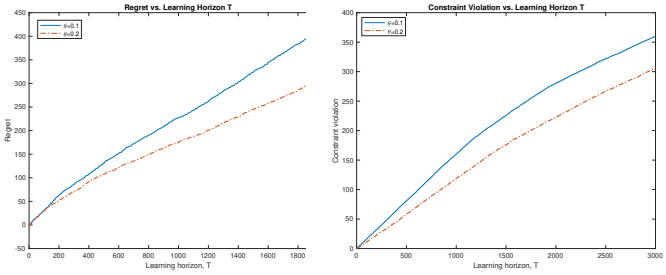

Figure 3. (Left): ConEXP3 Expected regret as a function of T. (Right): ConEXP3 Expected amount of constraint violation as a function of T.

based on the technical specs of a Raspberry RPi3B+ device, with maximum value 0.138 Watt. We have measured the energy consumption per inference instance for AlexNet on RPi3B+, and set $E^L = 3$ Joules. We also set $E^0 = 0.4 \times E^L$.

The non-stationary processes $\{b_t\}$ and $\{E_t^{tr}\}$ are taken to be

$$b_t = u_t \cdot |\sin t| \text{ and } E_t^{tr} = P_t \frac{S}{r_t} u_t |\cos(3t)|, \qquad (25)$$

where u_t is uniformly distributed in $[0, 1]$, S is the size of the image, and P_t, r_t are the transmit power and rate at time t. Process $\{b_t\}$ models a generic pedestrian appearance process. Energy costs are normalized in $[0, 1]$. Results are the average of 100 experiments.

In Fig. 2, we plot the average cumulative cost and average amount of constraint violation versus T, for $\theta = 0.1$ and $\theta = 0.2$. ConEXP3 asymptotically achieves zero violation as opposed to the EXP3 algorithm which does not cater for error constraints. In Fig. 3, we plot the average regret Reg_T^* and the average amount of constraint violation, Viol_T^* as functions of T for $\theta = 0.1$ and for $\theta = 0.2$. Both the regret and the amount of constraint violation increase sub-linearly with T, and both of these quantities are better off for $\theta = 0.2$.

VI. Conclusion

We studied the problem of learning the optimal offloading policy for image processing tasks in terms of a weighted sum of energy cost and object inference error rate. We mapped the problem to an adversarial bandit. For the constrained version of the problem, we proposed the ConEXP3 algorithm. ConEXP3 achieves satisfactory regret in the presence of error constraints, which is better than the one of the EXP3 algorithm that is constraint-agnostic. The model could be enhanced in other aspects such as delayed feedback.

Acknowledgment

This work was supported by the CHIST-ERA grant CHIST-ERA-18-SDCDN-004 (project LeadingEdge, grant number T11EPA4-00056) through the General Secretariat for Research and Innovation (GSRI).

References

[1] T. Y.-H. Chen, L. Ravindranath, S. Deng, P. Bahl, and H. Balakrishnan, "Glimpse: Continuous, real-time object recognition on mobile devices", *in Proc. ACM Conf. on Embedded Networked Sensor Sys. (SenSys)*, 2015.

[2] D. Kafetzis, I. Fourfouris, S. Argyropoulos, and I. Koutsopoulos, "UAV-assisted aerial survey of railways using deep learning", *in Proc 2020 Int. Conf. on Unmanned Aircraft Systems (ICUAS)*, 2020.

[3] J. Redmon and A. Farhadi. Yolov3: An incremental improvement, arXiv, abs/1804.02767 2018.

[4] TensorFlow. Tensorflow lite: https://www.tensorflow.org/lite.

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

[6] X. Chen, L. Jiao, W. Li, and X. Fu, "Efficient multi-user computation offloading for mobile-edge cloud computing", *IEEE/ACM Trans. Networking*, vol. 24, no.5, pp.2795–2808, Oct. 2016.

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

[8] J. Li, Z. Peng, B. Xiao, and Y. Hua, "Make smartphones last a day: Pre-processing based computer vision application offloading", *in Proc. IEEE Int. Conf. Sensing, Commun. and Networking (SECON)*, 2015.

[9] D. Callegaro and M. Levorato, "Optimal computation offloading in edge-assisted uav systems", *in Proc. IEEE Global Commun. Conf. (GLOBECOM)*, 2018.

[10] E. Hazan, Introduction to online convex optimization. Foundations and Trends in Optimization, 2(3-4):157–325, Aug. 2016.

[11] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multiarmed bandit problem", *Mach. Learning*, vol. 47, pp.235–256, May 2002.

[12] P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire, "The non-stochastic multiarmed bandit problem", *SIAM J. Comput.*, vol.32, no.1, pp.48–77, Jan. 2003.

[13] E. V. Belmega, P. Mertikopoulos, R. Negrel, and L. Sanguinetti, "Online convex optimization and no-regret learning: Algorithms, guarantees and applications", CoRR, abs/1804.04529, 2018.

[14] M. Mahdavi, R. Jin, and T. Yang, "Trading regret for efficiency: Online convex optimization with long term constraints", *J. Mach. Learn. Research*, vol.13, no.1, pp.2503–2528, Sept. 2012.

[15] R. Jenatton, J. C. Huang, and C. Archambeau, "Adaptive algorithms for online convex optimization with long-term constraints", *in Proc. Int. Conf. on Machine Learning (ICML)*, 2016.

[16] W. Sun, D. Dey, and A. Kapoor, "Safety-aware algorithms for adversarial contextual bandit", *in Proc. Int. Conf. on Machine Learning (ICML)*, 2020.

[17] W. Hu, B. Amos, Z. Chen, K. Ha, W. Richter, P. Pillai, B. Gilbert, J. Harkes, and M. Satyanarayanan, "The case for offload shaping", *in Proc. ACM Int. Workshop on Mobile Comp. Systems and Applications (HotMobile)*, 2015.

[18] S. Naderiparizi, P. Zhang, M. Philipose, B. Priyantha, J. Liu, and D. Ganesan, "Glimpse: A programmable early-discard camera architecture for continuous mobile vision", *in Proc. ACM Int. Conf. on Mobile Systems, Appl. and Services (MobiSys)*, 2017.

[19] J. Wang, Z. Feng, Z. Chen, S. George, M. Bala, P. Pillai, S.-W. Yang and M. Satyanarayanan, "Bandwidth-efficient live video analytics for drones via edge computing", *in Proc. IEEE/ACM Symp. on Edge Computing (SEC)*, 2018.

[20] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition" *in Proc. Int. Conf. on Learning Representations (ICLR)*, 2015.

[21] F. N. Iandola, M. W. Moskewicz, K. Ashraf, S. Han, W. Dally, and K. Keutzer, "Squeezenet: Alexnet-level accuracy with 50x fewer parameters and ¡0.5MB model size" ArXiv, abs/1602.07360, 2017.