Energy-Constrained Online Matching for Satellite-Terrestrial Integrated Networks

Jingye Wang, Xin Gao, Xi Huang, Qiuyu Leng, Ziyu Shao*, Yang Yang
School of Information Science and Technology, ShanghaiTech University
Email: {wangjy5, gaoxin, huangxi, lengqy, shaozy, yangyang}@shanghaitech.edu.cn

***Abstract*—In satellite-terrestrial integrated networks, it is a common practice to distribute real-time tasks from low Earth orbit (LEO) satellites to ground stations (GSs) for data processing. However, it remains an open problem how to match tasks with proper GSs in an online fashion with unknown dynamics, *e.g.,* transmission latency. Moreover, such a problem is further complicated by the non-trivial interaction between the decision-making procedure and long-term constraints on time-averaged energy consumptions. In this paper, by formulating the energy-constrained online matching problem with unknown transmission latency as a constrained Combinatorial Multi-Armed Bandit (CMAB) problem, we adopt bandit learning methods and virtual queue techniques to deal with the exploration-exploitation tradeoff and long-term constraints, respectively. With an effective integration of online learning and online control, we propose a *Task-matching and Resource-allocation with Data-driven Bandit Learning (TRDBL)* scheme. Our theoretical analysis shows that TRDBL achieves a sublinear regret bound with a time-averaged energy constraints guarantee in the long run. Through simulation results we not only verify our theoretical analysis but also demonstrate the outperformance of TRDBL in terms of both task latency reduction and energy efficiency.**

I. Introduction

In recent years, satellite-terrestrial integrated networks have attracted great attention owing to the rapid development of low Earth orbit (LEO) satellite systems [1]. By far, more and more LEO satellites have been launched for real-time tasks such as target surveillance [2] and environment monitoring [3]. In such scenarios, each LEO satellite collects data with onboard sensors and distributes tasks to the ground stations (GSs) for further data processing [4]. Considering that each LEO satellite may have access to more than one GS at a time [1], a key issue for each LEO satellite is how to match the tasks with proper GSs in an online fashion to achieve a minimum task latency.

However, the design of such matching remains an open problem. Specifically, the wireless channel dynamics between LEO satellites and GSs are usually unknown *a priori* for decision making [5]. Instead, the statistics of such uncertainty could only be inferred implicitly based on the subsequent feedback from the system. To this end, online learning should be integrated into the decision-making procedure. Moreover, the decision making for such matching often involves the concern of energy efficiency [6], which requires online control on the energy consumption to aid the matching procedure.

There are three challenges for such an integrated design. The *first* is regrading the exploration-exploitation dilemma in the learning procedure. Specifically, when an LEO satellite distributes tasks to the ground, the LEO satellite can either *exploit* the current knowledge by selecting the GS with the empirically lowest estimated transmission latency, or *explore* new knowledge by selecting the GS with less collected feedback. Besides, although the LEO satellite retains history information about the transmission latency, it is still unknown how to leverage such information to improve the learning efficiency. The *second* one is concerning the non-trivial tradeoff between energy consumption and task latency. When a GS receives a task, the GS has to allocate computing resources to process the task, which would incur energy consumption on the GS. In particular, the more energy the GS consumes, the lower the processing latency. Nonetheless, due to the scarcity of energy, the energy consumption should also be subject to a time-averaged constraint along with the objective of minimizing the task latency. The *third* lies in the interplay between online learning and online control procedures. Specifically, the learning procedure, if conducted ineffectively, may vitiate the control procedure and incur performance loss; meanwhile, the control procedure, if carried out improperly, would lead to low learning efficiency and excessive energy consumption.

A number of existing works have proposed various solutions for related problems. Specifically, Wang *et al.* [3] and Zhang *et al.* [7] utilized offline control techniques for task scheduling. Wang *et al.* [8] and He *et al.* [9] adopted online control techniques like Lyapunov optimization techniques to deal with dynamic arrivals. Nonetheless, they all assumed that instantaneous environment dynamics are available in the decision-making procedure. However, in practice, such information is usually unknown *a priori*. To deal with the uncertainty, later works adopted online learning techniques. For example, Cheng *et al.* [10] employed deep reinforcement learning techniques to propose a scheme with online learning. However, their design provided neither theoretical performance analysis nor consideration of time-averaged constraints.

In this paper, we address all the aforementioned challenges. In particular, we focus on the energy-constrained online matching between the tasks of one LEO satellite and GSs with unknown transmission latency. By investigating such a problem from the perspective of Combinatorial Multi-Armed Bandit (CMAB) [11], we propose an online scheme that integrates online control and online learning with a performance guarantee. The comparison between our work and the most related works is shown in Table I. Below we summarize the contributions and key results

This research was supported in part by the Nature Science Foundation of Shanghai under Grant 19ZR1433900. (*Corresponding author: Ziyu Shao)

TABLE I. Comparison between our work and related works

	Optimization Metrics	Dynamic Arrivals	Offline Control	Online Control	Online Learning	Offline History Information
[3]	Sum of the priorities of successfully scheduled tasks		•			
[7]	Throughput & energy consumption & transmission latency		•			
[8]	Throughput	•		•		
[9]	Number of completed tasks	•		•		
[10]	Task latency & energy consumption & server usage cost				•	
Our Work	Task latency & energy consumption	•		•	•	•

of this paper.

- **Problem Formulation:** We formulate the online matching problem as a CMAB problem with time-averaged constraints. Specifically, our formulation aims to minimize the performance loss (*i.e.,* regret) due to the decision making under uncertainty, while subject to the time-averaged energy constraints to ensure energy efficiency in the long run.
- **Algorithm Design:** We propose an integrated scheme called *TRDBL* (Task-matching and Resource-allocation with Data-driven Bandit Learning) to solve the formulated problem. In particular, TRDBL leverages data-driven bandit learning techniques to estimate the unknown transmission latency in the online learning procedure. In the online control procedure, we adopt Lyapunov optimization techniques to conduct the matching and resource allocation based on the estimates provided by the online learning procedure.
- **Theoretical Analysis:** Our theoretical analysis shows that TRDBL achieves a time-averaged regret bound of order $O(1/V+1/T+\sqrt{(\log T)/(T+H_{\min})})$ over a finite time horizon of T time slots, subject to the time-averaged energy constraints. The positive constant V is a tunable parameter, and the non-negative integer $H_{\min}$ is the minimal number of offline historical observations of each GS.
- **Numerical Evaluation:** We conduct extensive simulations to evaluate the performance of TRDBL and its variants. The simulation results demonstrate the effectiveness of TRDBL in achieving a low task latency under time-averaged energy constraints.

The rest of this paper is organized as follows. Section II introduces our system model and problem formulation. Section III elaborates the design of our algorithm, followed by its performance analysis in Section IV. Section V evaluates the performance of our approach through simulations. Finally, we conclude this paper in Section VI. Proofs and more results are delegated to our technical report [12].

II. System Model and Problem Formulation

A. Basic Model

We consider a satellite-terrestrial system which operates on a finite time horizon of T time slots, indexed by $\{0,1,\ldots,T-1\}$. The system consists of one LEO satellite and M GSs which are denoted by set $\mathcal{M} \triangleq \{1,2,\ldots,M\}$. As the set of accessible ground stations for the LEO satellite changes over time [3] [5], we denote such a ground station set in time slot t as $\mathcal{M}(t)$.

B. System Workflow

The workflow of the system during each time slot t proceeds as follows. At the beginning of the time slot, a set of tasks $\mathcal{A}(t) \triangleq \{A_1(t), A_2(t), \ldots, A_{N(t)}(t)\}$ arrive, where $N(t)$ ($0 \leq N(t) \leq n_{\max}$) is the number of tasks and varies over time slots. For each task $A_n(t)$, an observation of data size $S_n(t)$ ($0 < S_n(t) \leq s_{\max}$) is collected by the onboard sensors of the LEO satellite. To process the observation data, the LEO satellite needs to *match* the task with one GS of its accessible GS set $\mathcal{M}(t)$. Then the GS *allocates* its computing resources to process the task. At the end of the time slot, the information about the task latency and the incurred energy consumption is sent back to the LEO satellite.

C. Decision Variables

Matching Decision: For each task $A_n(t) \in \mathcal{A}(t)$ and GS $m \in \mathcal{M}(t)$, we denote the matching decision by variable $I_{n,m}(t) \in \{0,1\}$. If task $A_n(t)$ is matched with GS m ($I_{n,m}(t)=1$), the LEO satellite will distribute task $A_n(t)$ to GS m for processing. Since each task can only be assigned to one GS of its accessible GS set, the following constraints should be satisfied:

$$\sum_{m \in \mathcal{M}(t)} I_{n,m}(t) = 1, \ \forall n \in \mathcal{N}(t), t, \tag{1}$$

$$I_{n,m}(t) \in \{0,1\}, \ \forall n \in \mathcal{N}(t), m \in \mathcal{M}(t), t, \tag{2}$$

$$I_{n,m}(t) = 0, \ \forall n \in \mathcal{N}(t), m \notin \mathcal{M}(t), t, \tag{3}$$

where $\mathcal{N}(t) \triangleq \{1,2,\ldots,N(t)\}$.

Resource Allocation Decision: We adopt the CPU cycle frequency as the dominant computing resource[1]. Specifically, we consider the case where each GS can dynamically adjust its CPU cycle frequency [13]. We denote the CPU cycle frequency allocated to task $A_n(t)$ on GS m as $F_{n,m}(t)$ such that

$$f_{\min} \leq F_{n,m}(t) \leq f_{\max}, \quad \forall n \in \mathcal{N}(t), m \in \{m' | I_{n,m'}(t)=1, m' \in \mathcal{M}\}, t, \tag{4}$$

$$F_{n,m}(t) = 0, \quad \forall n \in \mathcal{N}(t), m \in \{m' | I_{n,m'}(t)=0, m' \in \mathcal{M}\}, t. \tag{5}$$

D. Performance Metrics

1) Task Latency: For each task $A_n(t) \in \mathcal{A}(t)$, its latency is composed of the transmission latency from the LEO satellite to its matched GS and the processing latency on the GS.

Transmission Latency: If the task $A_n(t)$ is distributed to GS m, it would experience a transmission latency of

[1]Our scheme can be also extended to take other kinds of resources into consideration.

$S_n(t)W_m(t)$, where $W_m(t)$ ($w_{\min} \leq W_m(t) \leq w_{\max}$) is the unit transmission latency from the LEO satellite to GS m during time slot t. Accordingly, given matching decision $\boldsymbol{I}(t) \triangleq (I_{n,m}(t))_{n\in\mathcal{N}(t),m\in\mathcal{M}}$, the transmission latency of task $A_n(t)$ is given by

$$D_n^{tr}(t) \triangleq \hat{D}_{n,t}^{tr}(\boldsymbol{I}(t)) = \sum_{m\in\mathcal{M}(t)} S_n(t) W_m(t) I_{n,m}(t). \quad (6)$$

Note that during each time slot t, the unit transmission latency $W_m(t)$ is a random variable with an unknown mean w_m. Moreover, $W_m(t)$ is assumed to be *i.i.d.* across time slots.

Offline History Information about Transmission Latency: As the LEO satellite retains offline historical observations about transmission latency, we denote such information for GS $m \in \mathcal{M}$ as a set $\{W_m^h(0), W_m^h(1), \ldots, W_m^h(H_m - 1)\}$, where $H_m > 0$ is the number of the offline historical observations. In particular, each observation $W_m^h(k)$ is assumed to have the same distribution as the unit transmission latency $W_m(t)$.

Processing Latency: To characterize the processing latency of task $A_n(t)$ on GS m, we define $L_{n,m}(t)(0 \leq L_{n,m}(t) \leq l_{\max})$ as the number of CPU cycles the GS needs to process per *bit* of the task. Accordingly, the processing latency is $L_{n,m}(t)S_n(t)/F_{n,m}(t)$. Then given matching decision $\boldsymbol{I}(t)$ and CPU cycle frequency allocation $\boldsymbol{F}(t) \triangleq (F_{n,m}(t))_{n\in\mathcal{N}(t),m\in\mathcal{M}}$, the processing latency of task $A_n(t)$ is

$$\begin{aligned} D_n^{pr}(t) &\triangleq \hat{D}_{n,t}^{pr}(\boldsymbol{I}(t), \boldsymbol{F}(t)) \\ &= \sum_{m\in\mathcal{M}(t)} L_{n,m}(t) S_n(t) I_{n,m}(t) / F_{n,m}(t). \end{aligned} \quad (7)$$

Therefore, the task latency of $A_n(t)$ is

$$\begin{aligned} D_n(t) &\triangleq \hat{D}_{n,t}(\boldsymbol{I}(t), \boldsymbol{F}(t)) \\ &= \hat{D}_{n,t}^{tr}(\boldsymbol{I}(t)) + \hat{D}_{n,t}^{pr}(\boldsymbol{I}(t), \boldsymbol{F}(t)). \end{aligned} \quad (8)$$

2) Energy Consumption: Similar to the task latency, the energy consumption for each task is composed of the transmission energy consumption on the LEO satellite and the processing energy consumption on the GS that receives the task.

Transmission Energy: For the transmission of tasks from the LEO satellite to GS m in time slot t, we denote the per-bit energy consumption as $C_m(t)$ ($0 < C_m(t) \leq c_{\max}$). Then given matching decision $\boldsymbol{I}(t)$, the total energy consumption for the transmission of task $A_n(t)$ is given by

$$\begin{aligned} E^{tr}(t) &\triangleq \hat{E}_t^{tr}(\boldsymbol{I}(t)) \\ &= \sum_{n\in\mathcal{N}(t)} \sum_{m\in\mathcal{M}(t)} C_m(t) S_n(t) I_{n,m}(t). \end{aligned} \quad (9)$$

As LEO satellites can only harvest energy from the solar and store the energy in the rechargeable battery [14], the energy consumption on the LEO satellite should be carefully controlled. To this end, we consider the following long-term energy constraints on the LEO satellite.

$$\limsup_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}\left[E^{tr}(t)\right] \leq \gamma_0, \quad (10)$$

where the energy budget γ_0 is predefined for the LEO satellite.

Processing Energy: For the processing of task $A_n(t)$ on GS m in time slot t, the incurred energy consumption is given by

$$\begin{aligned} E_m^{pr}(t) &\triangleq \hat{E}_{m,t}^{pr}(\boldsymbol{I}(t), \boldsymbol{F}(t)) \\ &= \sum_{n\in\mathcal{N}(t)} \kappa_m L_{n,m}(t) S_n(t) I_{n,m}(t) (F_{n,m}(t))^2, \end{aligned} \quad (11)$$

where parameter $\kappa_m > 0$ is a positive constant that is measurable in practice [15]. Regarding the scarcity of energy and the avenue for service providers, we consider the following constraints on the processing energy for each GS:

$$\limsup_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}\left[E_m^{pr}(t)\right] \leq \gamma_m, \ \forall m \in \mathcal{M}. \quad (12)$$

Here $\gamma_m > 0$ is the processing energy budget on GS m.

E. Problem Formulation

Our goal is to minimize the total task latency under long-term energy constraints on the LEO satellite and GSs over T time slots. Specifically, our problem formulation is given as follows:

$$\begin{aligned} \underset{\{\boldsymbol{I}(t),\boldsymbol{F}(t)\}_t}{\text{minimize}} \quad & \sum_{t=0}^{T-1} \sum_{n\in\mathcal{N}(t)} \mathbb{E}\left[D_n(t)\right] \\ \text{subject to} \quad & (1)(2)(3)(4)(5)(10)(12). \end{aligned} \quad (13)$$

III. Algorithm Design

For problem (13), given full knowledge about the transmission latency, it can be solved asymptotically optimally by Lyapunov optimization techniques [16]. However, such prior information is usually not available in practice. To characterize such a decision-making problem under uncertainty, we consider problem (13) from the perspective of constrained Combinatorial Multi-Armed Bandit (CMAB) [17]. Below we first introduce our reformulation. Then we present the details of our algorithm design and performance analysis.

A. Problem Reformulation

The classical CMAB [11] considers a sequential game between an agent and a bandit with multiple arms. The game is played over a finite number of time slots. In each time slot, the agent selects a subset of arms to play, then the bandit reveals a reward with respect to each played arm. The objective of the agent is to maximize the expected cumulative reward over time slots.

To reformulate problem (13), we view the LEO satellite as the agent and GSs as arms. In each time slot t, the LEO satellite (agent) matches each task with a GS (arm) from the available set $\mathcal{M}(t)$. Note that each GS may be matched with more than one task during the time slot. If GS m is matched with task $A_n(t)$ (arm m is played), a reward of $R_n(t) \triangleq -D_n(t)$ would be returned to the LEO satellite. Note that the task latency $D_n(t)$ defined in (8) is determined by not only the matching decision $I_{n,m}(t)$ but also the resource allocation decision $F_{n,m}(t)$. With

such a reformulation, problem (13) can be rewritten as the following equivalent problem.

$$\begin{aligned} \underset{\{\boldsymbol{I}(t),\boldsymbol{F}(t)\}_t}{\text{maximize}} \quad & \sum_{t=0}^{T-1} \sum_{n\in\mathcal{N}(t)} \mathbb{E}\left[R_n\left(t\right)\right] \\ \text{subject to} \quad & (1)(2)(3)(4)(5)(10)(12). \end{aligned} \tag{14}$$

To characterize the performance loss due to the decision making under uncertainty, we define the *regret* as

$$Reg\left(T\right) \triangleq R^*(T) - \frac{1}{T}\sum_{t=0}^{T-1} \sum_{n\in\mathcal{N}(t)} \mathbb{E}\left[R_n\left(t\right)\right], \tag{15}$$

where $R^*(T)$ is the optimal expected reward that can be achieved. The reward maximization problem (14) is then equivalent to the following regret minimization problem:

$$\begin{aligned} \underset{\{\boldsymbol{I}(t),\boldsymbol{F}(t)\}_t}{\text{minimize}} \quad & Reg\left(T\right) \\ \text{subject to} \quad & (1)(2)(3)(4)(5)(10)(12). \end{aligned} \tag{16}$$

To solve problem (16), we leverage both data-driven bandit learning and Lyapunov optimization. The detailed design is given in the following subsections.

B. Data-Driven Bandit Learning for Uncertainty

Based on the collected online feedback and the offline history information retained on the LEO satellite, for each GS $m \in \mathcal{M}$, we adopt UCB1 method [18] to estimate the mean transmission latency w_m as follows:

$$\begin{aligned} &\tilde{w}_m\left(t\right) \\ &\quad = \max\left\{\bar{w}_m\left(t\right) - \omega_0\sqrt{\frac{3\log t}{2\left(h_m\left(t\right)+H_m\right)}}, w_{\min}\right\}, \end{aligned} \tag{17}$$

where $\omega_0 \triangleq w_{\max} - w_{\min}$, and $h_m(t)$ is the counter that keeps track of the number of times GS m is selected by time slot t. In (17), the term $\bar{w}_m(t)$ is the empirical mean of the transmission latency over both collected online feedback and offline history information. Such a term reflects the acquired knowledge about the mean transmission latency and thus it is known as the *exploitation* term. Meanwhile, the value of the term $\sqrt{(3\log t)/(2(h_m(t)+H_m))}$ is inversely proportional to the selection times of GS m. The fewer the times GS m has been selected, the more chances for GS m to be selected in the subsequent time slots. Therefore the term is also called the *exploration* term. Accordingly, constant w_0 measures the relative weight of the exploitation to the exploration. Other learning methods such as ε-greedy [18] can also be applied in our scheme, which is discussed in Section V.

C. Online Control for Regret Minimization under Constraints

We adopt virtual queue techniques [16] to handle the time-averaged energy constraints in (10) and (12). Specifically, we introduce a virtual queue $\{Q_0(t)\}_t$ for the LEO satellite and a virtual queue $\{Q_m(t)\}_t$ for each GS $m \in \mathcal{M}$. The backlog size of each virtual queue is initialized as zero and then updated at the end of each time slot t as follows:

$$Q_0\left(t+1\right) = \left[Q_0\left(t\right) - \gamma_0\right]^+ + E^{tr}\left(t\right), \tag{18}$$

$$Q_m\left(t+1\right) = \left[Q_m\left(t\right) - \gamma_m\right]^+ + E_m^{pr}\left(t\right), \ \forall m \in \mathcal{M}, \tag{19}$$

in which $[\cdot]^+ \triangleq \max\{\cdot, 0\}$. When the stability of each queueing process $\{Q_m(t)\}_t, m \in \{0\} \cup \mathcal{M}$ is ensured, the energy consumption constraints (10) and (12) will be guaranteed [16].

To ensure the queue stability while minimizing the task latency, we adopt Lyapunov optimization techniques [16] to transform the stochastic optimization problem (16) into a series of sub-problems over time slots, which is given by

$$\begin{aligned} \underset{\boldsymbol{I}(t),\boldsymbol{F}(t)}{\text{minimize}} \quad & \sum_{n\in\mathcal{N}(t)} \sum_{m\in\mathcal{M}} \tilde{p}_{n,m}^t\left(F_{n,m}\left(t\right)\right) I_{n,m}\left(t\right) \\ \text{subject to} \quad & (1)(2)(3)(4)(5). \end{aligned} \tag{20}$$

In (20), for each task $A_n(t)$ and each GS m, we define $\tilde{p}_{n,m}^t(F_{n,m}(t))$ as

$$\begin{aligned} &\tilde{p}_{n,m}^t(F_{n,m}(t)) \triangleq VS_n(t)(\tilde{w}_m(t) + L_{n,m}(t)/F_{n,m}(t)) \\ &\quad + S_n(t)(Q_0(t)C_m(t) + Q_m(t)\kappa_m L_{n,m}(t)(F_{n,m}(t))^2), \end{aligned} \tag{21}$$

which represents the price of matching task $A_n(t)$ with GS m. Such a price is a weighted sum of task latency (the first term) and energy consumption (the second term). The positive tunable parameter V determines the relative importance of regret minimization to the energy consumption.

Note that problem (20) is a mixed integer programming problem, the optimal solution for such a problem is given as follows[2]. For each $n \in \mathcal{N}(t)$ and $m \in \mathcal{M}$, we have

$$I_{n,m}\left(t\right) = \begin{cases} 1, & \text{if } m = \arg\min_{m'\in\mathcal{M}(t)} \tilde{p}_{n,m'}^t(\check{F}_{n,m'}(t)); \\ 0, & \text{otherwise;} \end{cases} \tag{22}$$

$$F_{n,m}\left(t\right) = \begin{cases} \check{F}_{n,m}\left(t\right), & \text{if } I_{n,m}\left(t\right) = 1; \\ 0, & \text{otherwise;} \end{cases} \tag{23}$$

where the optimal resource allocation $\check{F}_{n,m}(t)$ is defined as

$$\check{F}_{n,m}(t) = \max\left\{\min\left\{\sqrt[3]{V/(2\kappa_m Q_m(t))}, f_{\max}\right\}, f_{\min}\right\}. \tag{24}$$

In summary, with an effective integration of data-driven bandit learning and online control, we propose an online scheme for joint task matching and resource allocation called TRDBL. The pseudocode of TRDBL is shown in Algorithm 1.

D. Computational Complexity of TRDBL

There are four procedures in TRDBL: data-driven online learning procedure (lines 2-6), task matching procedure (lines 7-12), computing resource allocation procedure (lines 13-17), and updating procedure (lines 18-22). The computational complexity of TRDBL is mainly incurred by the task matching procedure. Specifically, in the worst case, the for-loop (lines 8-11) would involve $n_{\max}$ iterations. Note that the computational complexity of line 9 is $O(M)$. Accordingly, the computational complexity of TRDBL is $O(n_{\max}M)$.

[2]If more than one GS achieves $\min_{m'\in\mathcal{M}(t)} \tilde{p}_{n,m'}(\check{F}_{n,m'}(t))$, the LEO satellite would select one of such GSs uniformly at random.

Algorithm 1 Task-matching and Resource-allocation with Data-driven Bandit Learning (TRDBL)

Input: Initialize $h_m(0)=0$, $\bar{w}_m(0)=\frac{1}{H_m}\sum_{k=0}^{H_m-1}W_m^h(k)$, and $\tilde{w}_m(0)=w_{\min}$ for each GS $m\in\mathcal{M}$.
Output: Matching decision and resource allocation decision over T time slots, *i.e.*,$\{\boldsymbol{I}(t),\boldsymbol{F}(t)\}_{t=0}^{T-1}$.
for $t=0,1,2,\ldots,T-1$ **do**:
%Data-driven Online Learning
  for each GS $m\in\mathcal{M}$ **do**
    if $h_m(t)>0$ **then**
      $\tilde{w}_m(t)\leftarrow\max\left\{\bar{w}_m(t)-\omega_0\sqrt{\frac{3\log t}{2(h_m(t)+H_m)}},w_{\min}\right\}$.
    end if
  end for
%Task Matching
  Initialize $I_{n,m}(t)=0$ for each $n\in\mathcal{N}(t)$ and $m\in\mathcal{M}$.
  for each task $A_n(t)\in\mathcal{A}(t)$ **do**
    $\mathcal{M}_{\min}(t)\leftarrow\{m|m\in\arg\min_{m'\in\mathcal{M}(t)}\tilde{p}_{n,m'}^t(\check{F}_{n,m'}(t))\}$.
    Uniformly randomly select GS m from set $\mathcal{M}_{\min}(t)$ and conduct matching $I_{n,m}(t)\leftarrow 1$.
  end for
  The LEO satellite distributes tasks in set $\mathcal{A}(t)$ to GSs according to matching decision $\boldsymbol{I}(t)$.
%Computing Resource Allocation
  Initialize computing resource allocation $F_{n,m}(t)=0$ for each $n\in\mathcal{N}(t)$ and $m\in\mathcal{M}$.
  for each GS $m\in\mathcal{M}$ **do**
    Set $F_{n,m}(t)\leftarrow\check{F}_{n,m}(t)$ for each task $n\in\mathcal{N}(t)$ distributed to GS m, *i.e.*, task n with $I_{n,m}(t)$=1.
  end for
  Each GS $m\in\mathcal{M}$ processes received tasks according to computing resource allocation $\{F_{n,m}(t)\}_{n\in\mathcal{N}(t)}$ on it.
%Update of Virtual Queues and Statistics
  Update virtual queues $\boldsymbol{Q}(t)$ according to (18) and (19).
  for each GS $m\in\mathcal{M}$ **do**
    $h_m(t+1)\leftarrow h_m(t)+\prod_{n\in\mathcal{N}(t)}I_{n,m}(t)$.
    $\bar{w}_m(t+1)\leftarrow\frac{h_m(t)+H_m}{h_m(t+1)+H_m}\bar{w}_m(t)+\frac{W_n(t)\prod_{n\in\mathcal{N}(t)}I_{n,m}(t)}{h_m(t+1)+H_m}$.
  end for
end for

IV. Performance Analysis

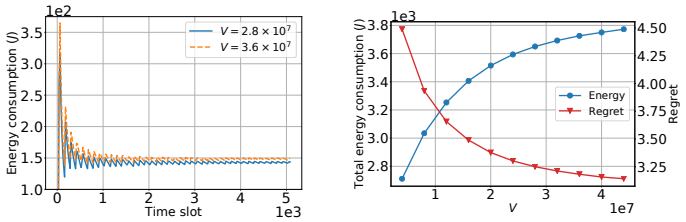
A. Energy Consumption Guarantee

An energy budget vector $\boldsymbol{\gamma}\triangleq(\gamma_0,\gamma_1,\ldots,\gamma_M)$ is said to be *feasible* if there exists a feasible scheme to problem (13). The set of all feasible energy budget vector is defined as the *maximal feasibility region*. Based on such definitions, we provide the following theorem to show the performance of TRDBL.

Theorem 1: Suppose that the energy budget vector $\boldsymbol{\gamma}$ lies in the interior of the maximal feasibility region, then TRDBL satisfies the energy constraints in (10) and (12). Furthermore, the virtual queues $\{Q_m(t)\}_t, m\in\{0\}\cup\mathcal{M}$, defined in (18) and (19) are strongly stable and there exists a positive constant ϵ such that

$$\limsup_{T\to\infty}\frac{1}{T}\sum_{t=0}^{T-1}\sum_{m\in\{0\}\cup\mathcal{M}}\mathbb{E}[Q_m(t)] \le\Gamma/\epsilon+Vs_{\max}n_{\max}(\omega_0+w_{\max}+l_{\max}/f_{\min})/\epsilon, \quad (25)$$

where $\Gamma\triangleq n_{\max}^2s_{\max}^2(c_{\max}^2+l_{\max}^2f_{\max}^4\sum_{m\in\mathcal{M}}\kappa_m^2)/2+\sum_{m\in\mathcal{M}\cup\{0\}}\gamma_m^2/2$.



(a) Time-averaged processing energy consumption on GS 10.

(b) The energy consumption and regret under various values of V.

Fig. 1. Performance of TRDBL on energy consumption.

Remark: Theorem 1 shows that TRDBL can effectively guarantee the stability of each virtual queue, thereby satisfying the time-averaged energy consumption constraints.

B. Regret Bound

The theoretical upper bound for the regret achieved by TRDBL is characterized by the following theorem.

Theorem 2: Under TRDBL, the regret defined in (15) is upper bounded by

$$Reg\,(T)\le\frac{\Gamma}{V}+\frac{4\omega_0n_{\max}s_{\max}}{T}+2\omega_0n_{\max}s_{\max}\sqrt{\frac{6n_{\max}M\log T}{T+H_{\min}}}, \quad (26)$$

where the non-negative integer $H_{\min}\triangleq\min_{m\in\mathcal{M}}H_m$.

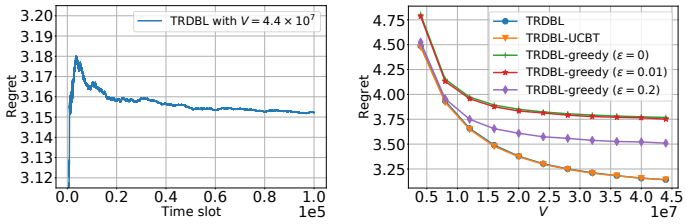
Remark: As shown in Theorem 2, the regret upper bound depends on the time horizon length T and the value of parameter V. As T increases to infinity, the regret bound decreases to Γ/V since the last two terms in (26) is in the order of $O(1/T+\sqrt{(\log T)/(T+H_{\min})})$. On the other hand, the increase of the value of V would conduce a lower regret upper bound for TRDBL. In practice, the choice of the value of V depends on the design of the real system.

V. Numerical Results

A. Simulation Settings

We consider a satellite-terrestrial system with one LEO satellite and 25 GSs. In each time slot, due to the system dynamics, only three GSs are accessible to the LEO satellite.

The parameter settings are based on the common settings in satellite-terrestrial systems [3] [5]. In each time slot t, the number $N(t)$ of task arrivals is sampled from a Poisson distribution with parameter $\lambda=6$, while the size of each task (in the unit of *bits*) is generated from a uniform distribution over $(3\times10^6,5\times10^6)$. The transmission latency (in the unit of *s/bit*) for GS m is generated from a uniform distribution over $(R_{\min}(m),R_{\max}(m))$, where the minimum and the maximum transmission latencies, $R_{\min}(m)$ and $R_{\max}(m)$, are sampled uniformly at random from intervals $(1\times10^{-7},2\times10^{-7})$ and $(5\times10^{-7},6\times10^{-7})$, respectively. For the energy constraints, referring to the settings in [13], we set $\gamma_0=80$ *J* for the LEO satellite and $\gamma_m=150$ *J*, $m\in\{1,2,\ldots,25\}$, for each GS.



(a) Regret of TRDBL with $H_{\min} = 100$.

(b) Comparison of TRDBL and its variants.

Fig. 2. Performance of TRDBL on regret.

B. Performance Evaluation

1) Energy Consumption with Different Values of V: In Figure 1(a), by taking 10th GS (GS 10) as an example, we show how the time-averaged energy consumption changes across time slots. We see the fluctuation of the time-averaged energy consumption over time under different values of V. Such fluctuation is caused by the fact that when the LEO satellite has no access to the GS, the GS would receive no tasks to be processed, thereby resulting the drop with respect to the time-averaged energy consumption. However, in the long run, the time-averaged energy constraint is satisfied under TRDBL.

2) Tradeoff between Regret and Total Energy Consumption: In Figure 1(b), the results show that the larger the value of V, the lower the regret and the larger the total energy consumption. Such results verify the tradeoff between the task latency and the energy consumption as suggested in our theoretical analysis.

3) Regret of TRDBL over Time Slot: Taking the case where $V = 4.4 \times 10^7$ and $H_{\min} = 100$ as an example, we show how the regret of TRDBL evolves over time slot in Figure 2(a). The sharp increase of the regret during about the first 10^4 time slots is induced by the control for energy constraints. Specifically, under the control of TRDBL, GSs tend to process tasks with a relatively high latency to satisfy energy constraints, which causes the increase in the regret. However, in the long run, the regret decreases as the time horizon length T increases, which is consistent with our theoretical analysis.

4) Regret under Different Exploration Strategies: To investigate the regret under different exploration strategies, we propose two variants of TRDBL: one leveraging ε-greedy method and the other derived from UCB-Tuned (UCBT) [19]. The comparison between TRDBL and its variants, TRDBL-greedy with $\varepsilon \in \{0, 0.01, 0.2\}$ and TRDBL-UCBT, is presented in Figure 2(b). The results show that when the value of V increases, all the schemes have the same trends with respect to the regret. Among these schemes, both TRDBL and TRDBL-UCBT take advantage of adaptive exploration and achieve relatively low regrets.

VI. Conclusion

In this paper, we studied the problem of the energy-constrained online matching between tasks of one LEO satellite and GSs with unknown transmission latency from the perspective of constrained CMAB. With an effective integration of online learning and online control, we proposed an online scheme for joint task matching and resource allocation. Our theoretical analysis and simulation results demonstrated the effectiveness of our proposed scheme in achieving a sublinear regret bound while subject to the long-term constraints. In addition, we would like to investigate how to apply TRDBL in the case with multiple LEO satellites, which can be an interesting future work.

References

[1] B. Di, L. Song, Y. Li, and H. V. Poor, "Ultra-dense leo: Integration of satellite access networks into 5g and beyond," *IEEE Wireless Communications*, vol. 26, no. 2, pp. 62–69, 2019.

[2] X. Jia, T. Lv, F. He, and H. Huang, "Collaborative data downloading by using inter-satellite links in leo satellite networks," *IEEE Transactions on Wireless Communications*, vol. 16, no. 3, pp. 1523–1532, 2017.

[3] Y. Wang, M. Sheng, W. Zhuang, S. Zhang, N. Zhang, R. Liu, and J. Li, "Multi-resource coordinate scheduling for earth observation in space information networks," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 2, pp. 268–279, 2018.

[4] D. Zhou, M. Sheng, J. Luo, R. Liu, J. Li, and Z. Han, "Collaborative data scheduling with joint forward and backward induction in small satellite networks," *IEEE Transactions on Communications*, vol. 67, no. 5, pp. 3443–3456, 2019.

[5] C. Niephaus, M. Kretschmer, and G. Ghinea, "Qos provisioning in converged satellite and terrestrial networks: A survey of the state-of-the-art," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 4, pp. 2415–2441, 2016.

[6] J. Li, K. Xue, D. S. Wei, J. Liu, and Y. Zhang, "Energy efficiency and traffic offloading optimization in integrated satellite/terrestrial radio access networks," *IEEE Transactions on Wireless Communications*, vol. 19, no. 4, pp. 2367–2381, 2020.

[7] M. Zhang and W. Zhou, "Energy-efficient collaborative data downloading by using inter-satellite offloading," in *Proceedings of IEEE GLOBECOM*, 2019.

[8] Y. Wang, M. Sheng, J. Li, X. Wang, R. Liu, and D. Zhou, "Dynamic contact plan design in broadband satellite networks with varying contact capacity," *IEEE Communications Letters*, vol. 20, no. 12, pp. 2410–2413, 2016.

[9] L. He, J. Li, M. Sheng, R. Liu, K. Guo, and D. Zhou, "Dynamic scheduling of hybrid tasks with time windows in data relay satellite networks," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 5, pp. 4989–5004, 2019.

[10] N. Cheng, F. Lyu, W. Quan, C. Zhou, H. He, W. Shi, and X. Shen, "Space/aerial-assisted computing offloading for iot applications: A learning-based approach," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 5, pp. 1117–1129, 2019.

[11] W. Chen, Y. Wang, and Y. Yuan, "Combinatorial multi-armed bandit: General framework and applications," in *Proceedings of ICML*, 2013.

[12] J. Wang, X. Gao, X. Huang, Q. Leng, Z. Shao, and Y. Yang, "Energy-constrained online matching for satellite-terrestrial integrated networks," ShangheiTech University, Tech. Rep., 2020. [Online]. Available: http://faculty.sist.shanghaitech.edu.cn/faculty/shaozy/matching.pdf

[13] N. Budhdev, M. C. Chan, and T. Mitra, "Pr^3: Power efficient and low latency baseband processing for lte femtocells," in *Proceedings of IEEE INFOCOM*, 2018.

[14] D. Zhou, M. Sheng, B. Li, J. Li, and Z. Han, "Distributionally robust planning for data delivery in distributed satellite cluster network," *IEEE Transactions on Wireless Communications*, vol. 18, no. 7, pp. 3642–3657, 2019.

[15] W. Zhang, Y. Wen, K. Guan, D. Kilper, H. Luo, and D. O. Wu, "Energy-optimal mobile cloud computing under stochastic wireless channel," *IEEE Transactions on Wireless Communications*, vol. 12, no. 9, pp. 4569–4581, 2013.

[16] M. J. Neely, "Stochastic network optimization with application to communication and queueing systems," *Synthesis Lectures on Communication Networks*, vol. 3, no. 1, pp. 1–211, 2010.

[17] F. Li, J. Liu, and B. Ji, "Combinatorial sleeping bandits with fairness constraints," in *Proceedings of IEEE INFOCOM*, 2019.

[18] A. Slivkins *et al.*, "Introduction to multi-armed bandits," *Foundations and Trends® in Machine Learning*, vol. 12, no. 1-2, pp. 1–286, 2019.

[19] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multiarmed bandit problem," *Machine learning*, vol. 47, no. 2-3, pp. 235–256, 2002.