Age-of-Information-Aware Mobile Crowdsensing for Uncertain Event Capture

Jinrui Zhou*, Yin Xu*, Haotian Xu*, Mingjun Xiao*, and Jie Wu†‡
*School of Computer Science and Technology & Suzhou Institute for Advanced Research,
University of Science and Technology of China
†China Telecom Cloud Computing Research Institute
‡Department of Computer and Information Sciences, Temple University
Email: *{zzkevin@mail, yinxu@, xht2002@mail, xiaomj@}.ustc.edu.cn †‡jiewu@temple.edu

***Abstract*—Mobile CrowdSensing (MCS) is a crowdsourcing-based paradigm that leverages mobile users to collect data from Points of Interest (PoIs) using their smart devices. As data freshness has become a crucial concern, Age of Information (AoI) is employed to measure data freshness for MCS systems. Existing AoI-aware MCS works mainly focus on direct data collection scenarios, where data in PoIs is always available. Unlike these works, this paper explores the AoI-aware MCS system for uncertain event capture applications, where events may occur frequently yet with uncertainty, making the AoI update hard to be estimated. First, we model this uncertain event capture problem as a constrained episodic restless bandit problem with unknown transition probability. Next, we propose a belief-DPP bandit policy by extending the Drift-Plus-Penalty (DPP) policy. By combining belief-DPP and the Thompson Sampling (TS) technique, we further propose the TS-DPP algorithm, so as to minimize the cumulative weighted AoI values of all events under a given budget constraint. We analyze the theoretical performance of the TS-DPP algorithm, and derive a sublinear Bayesian regret bound $\mathcal{O}(\sqrt{T\log T})$, where T is the size of the time horizon. Additionally, we conduct extensive simulations to demonstrate the significant performance of the TS-DPP algorithm.**

***Index Terms*—Crowdsensing, event capture, age of information, restless bandit.**

I. Introduction

A. Background and Motivation

Mobile CrowdSensing (MCS) is a crowdsourcing-based sensing paradigm that a platform can recruit a crowd of mobile users (a.k.a., workers) to collect data from some Points of Interest (PoIs) with carried smart devices [1]–[7]. By leveraging the mobility of users, MCS enables efficient acquisition of large-scale data, whose applications span various domains including traffic monitoring [4], environmental monitoring [1], and so on. Recently, in order to provide more valuable data for machine learning model training, the freshness of data has become a highly concerned issue. Age of Information (AoI), defined as the elapsed time since the latest update or generation of the data collected or transmitted from some PoI, is proposed to measure the freshness of data [8], [9], having gained significant attention in a variety of studies [10]–[15].

In this paper, we focus on the AoI-aware event capture issue in MCS, where the platform needs to schedule suitable workers to capture uncertain PoI-related events. Although much effort has been devoted to data collection [16], [17], most of them assume that data at each PoI is always available and can be collected when a worker arrives. However, these assumptions are unrealistic in practical applications. As illustrated in Fig. 1, some PoI-related events might occur within a large-scale sensing area frequently and uncertainly, such as traffic congestion events at P1, exhaust emissions events at P2, and so on. The platform aims to collect all event information with minimal AoI. On one hand, the platform does not know whether an event has occurred at a particular location or when it will happen. If the platform rashly recruits workers to a PoI (e.g., P3) where no event has occurred, it would lead to fruitless efforts and stale information. On the other hand, the total budget of the platform (e.g., recruitment salary) is limited. If the platform spends too much of its budget recruiting workers to capture events at distant PoIs (e.g., P2), it will deprive other PoIs of adequate exploration and updates, ultimately failing to minimize the average AoI values across the system. Hence, it is highly significant to explore efficient AoI-aware event capture strategies for the platform at each time step while facing an unknown environment and budget constraints.

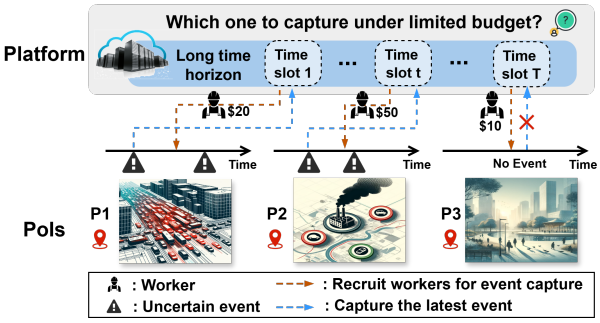


Fig. 1. Illustration of the crowdsensing for event capture.

B. Challenges

There are two major challenges that need to be overcome in addressing the AoI-aware event capture issue. The first challenge is that both the generation and capture of events are often probabilistic, which introduces a dual uncertainty that makes it difficult for the platform to make efficient decisions. Specifically, the platform not only lacks knowledge of whether each PoI has generated a new event, but also faces uncertainty about the exact occurrence time of an event. For example, as shown in Fig. 2, a PoI generates three events when $t=0, 6, 18$,

and the platform recruits five workers for the event capture. Unlike the traditional AoI model, the 3rd and 4th event captures (i.e., $t=12, 15$) in our AoI model cannot encounter an event and thus do not reset the AoI. Additionally, even when the 1st, 2nd, and 5th (i.e., $t=3, 9, 21$) captures are successful, the AoI is not reset to zero, but rather to the age of the captured event. Therefore, the dual uncertainty of event occurrence complicates the decision-making process, as the platform must continuously adapt its strategies to balance exploration and exploitation while minimizing AoI under uncertain conditions.

The second challenge stems from the temporal coupling introduced by both AoI dynamics and the long-term budget constraint, which together significantly complicate the planning and optimization of future decisions. Specifically, each event capture decision affects not only the current AoI reduction but also the future evolution of AoI and the remaining budget. This is due to the fact that capturing events at some PoIs resets their AoI, whereas the AoI values of the remaining PoIs continue to accumulate. Moreover, the presence of a long-term budget constraint introduces an additional dimension of complexity, since the platform must strategically manage budget consumption over time to maximize long-term freshness by balancing short-term AoI reduction and the ability to respond to future events. These temporal dependencies—in both AoI evolution and budget consumption—invalidate standard online learning approaches that assume static reward structures or independent decision rounds, thus requiring more sophisticated strategies that account for long-term trade-offs under uncertainty.

Several works leverage Lyapunov optimization (e.g., Drift-Plus-Penalty algorithm [18], [19]) to tackle the long-term constrained optimization problem [20], [21]. Although these methods are effective in many stochastic settings, they are generally designed for ideal scenarios where random events are observable before making decisions. On the other hand, a few studies adopt restless bandit models [22], [23] or Whittle index-based methods [24], [25] to address decision-making problems under uncertainty. However, these methods usually rely on the prior knowledge of state transition dynamics. Consequently, none of the previous research has comprehensively addressed the aforementioned challenges.

C. Solution and Contribution

To circumvent the above challenges, we model the event capture decision problem as an online constrained restless bandit problem with unknown states and transitions. In this model, the platform acts as the *learner*, each PoI represents an *arm*, the AoI value of each event corresponds to the *state* of the associated arm, and pulling an arm signifies capturing the corresponding event. Unlike existing constrained restless bandit issues, our model lacks knowledge of the state transition probability due to the dual uncertainty in event capture. To this end, we propose a TS-DPP algorithm by skillfully combining the Thompson Sampling (TS) and Drift-Plus-Penalty (DPP) techniques. Specifically, we first employ the TS technique to estimate the occurrence frequencies of events. Next, we extend the DPP algorithm to solve stochastic optimization problems,

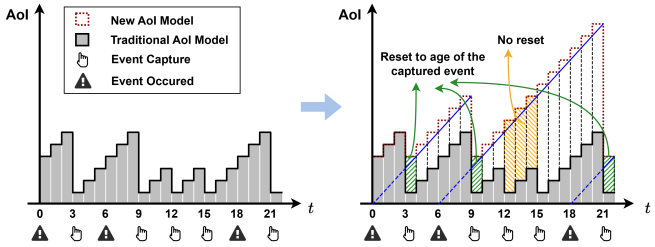


Fig. 2. Illustration of the deviation between two AoI models.

and design a belief-DPP bandit policy for allocating the budget and selecting PoIs. The TS-DPP algorithm dynamically adjusts event capture strategies to accommodate dual uncertainty and limited budget constraints, and continuously optimizes system performance via online learning. Overall, our main results and key contributions are as follows:

- We introduce AoI-aware MCS to event capture applications, involving a long-term stochastic optimization problem with unknown Markovian transition. To the best of our knowledge, this is the first AoI-aware MCS study that addresses uncertain event capture while simultaneously tackling dual uncertainty and temporal coupling.
- We propose a TS-DPP algorithm to solve the constrained restless bandit problem with unknown transition probability, in which a belief-DPP bandit policy is designed to make efficient AoI-aware event capture strategies.
- We analyze the theoretical performance of the TS-DPP algorithm, compared with a widely-used offline oracle, and derive a sublinear Bayesian regret bound $\mathcal{O}(\sqrt{T\log T})$, where T is the size of the time horizon.
- We conduct extensive simulations to verify the theoretical analysis results and demonstrate the significant performance of the TS-DPP algorithm.

II. System Model and Problem Formulation

A. System Overview and AoI Models

We consider an MCS system for uncertain event capture scenarios. Specifically, there are N PoIs, denoted by the set $\mathcal{N}=\{1,\ldots,N\}$, each of which might produce a PoI-related event in a certain probability. Time is divided into T equal-length time slots, represented by the set $\mathcal{T}=\{1,2,\ldots,T\}$. Moreover, we consider an episodic setting [16], wherein each episode runs over a finite time horizon L. We assume that there are m episodes in total, implying $T=mL$. At the end of each episode, the system resets to its initial state, which aligns with practical scheduling cycles in MCS systems.

In each time slot, the i-th PoI might produce an event e_i with an unknown probability, denoted by θ_i. The probabilities across all PoIs are represented as the vector $\theta=(\theta_1,\theta_2,\ldots,\theta_N)$. Meanwhile, the platform maintains a local copy of each PoI-related event. At the beginning of each time slot, it will make the event capture decision, i.e., selecting a PoI and recruiting workers to capture the related event. If the event happens during this time slot, it will be captured by the workers and transmitted to the platform. Consequently, the platform will update its local copy with the fresh event. Otherwise, if no event is captured, the local copies remain

TABLE I
DESCRIPTION OF KEY NOTATIONS

Variable	Description
$\mathcal{N}, N$	the set of PoIs and the number of PoIs.
$\mathcal{T}, T$	the set of time slots and the number of time slots.
m, L	the number and the length of episodes.
l, t	the index of episodes and the index of rounds.
θ_i, θ	the probability of event e_i.
θ^*, θ^l	the true and the estimation of probability.
$X_i(t), X_i^b(t)$	the AoI of event e_i in PoI i and its estimation.
$A_t, Z(t)$	the arm chosen at time t and the virtual queue.
$Y_i(t;\theta,\pi)$	the AoI of event copy in PoI i at time t.
$\widetilde{Y}_i(t;\theta,\pi)$	the approximation of $Y_i(t;\theta,\pi)$.
c_i, B	the cost to capture the event e_i and the budget.
$\omega^o(t), \omega^u(t)$	the observed (unobserved) AoI state vector.

unchanged. For ease of presentation, we assume the platform selects only one PoI for event capture in each time slot, which can be readily extended to the case of multiple PoIs. Here, both the event at each PoI and the event copy in the platform have the AoI metric, which can be modeled as follows.

Definition 1 (***AoI of Event***). *The AoI of an event is defined as the elapsed time since the time when the event occurs in the corresponding PoI. Let $X_i(t;\theta)$ denote the AoI of the event in the i-th PoI at time t. Then, it is formulated as*

$$X_i(t;\theta) \triangleq t - u_i(t;\theta), \tag{1}$$

where $u_i(t;\theta)$ denotes the latest occurrence time up to time t.

According to its definition, the AoI of an event is reset to zero when the event occurs; otherwise, it increases over time. In other words, the update of AoI for each event e_i can be formalized as a Markov chain over time t, where the state is denoted by $X_i(t;\theta)$ and the corresponding state transition probabilities are given by the following equations:

$$Pr[X_i(t+1;\theta)=0] = \theta_i, \tag{2}$$
$$Pr[X_i(t+1;\theta)=X_i(t;\theta)+1] = 1-\theta_i, \tag{3}$$

with initial state $X_i(0;\theta)=0$. This AoI update model indicates that the AoI state $X_i(t;\theta)$ can be viewed as a function of the event probability parameter θ. Moreover, we define the set $\omega^u(t;\theta) = (X_1(t;\theta), \ldots, X_N(t;\theta))$ as the *AoI state vector*, representing the AoI states of all events, which can also be regarded as a function of θ. For simplicity, we will use $X_i(t)$ and $\omega^u(t)$ hereafter, omitting the parameter θ when no ambiguity arises. Note that since the event probability parameter θ is unknown, the AoI state vector $\omega^u(t)$ is unobservable. Before introducing the AoI model of each event copy maintained in the platform, we define the event capture decision strategy.

Definition 2 (***Event Capture Strategy***). *We denote the event capture strategy decided by the platform at time t as a vector $\pi(t)=(\pi_1(t),\ldots,\pi_N(t)) \in \{0,1\}^N$, where $\pi_i(t)$ $(1\le i\le N)$ is an indicator, i.e., $\pi_i(t)=1$ means that the platform decides to capture the event e_i in the i-th PoI at time t; otherwise, $\pi_i(t)=0$. Additionally, let $\pi=\{\pi(1),\ldots,\pi(T)\}$.*

Having defined the event capture strategy, we now introduce the AoI model of the event copies maintained in the platform. Specifically, the AoI of each event copy depends not only on the occurrence of the corresponding event but also on the platform's capture decisions governed by the strategy.

Definition 3 (***AoI of Event Copy***). *The AoI of an event copy in the platform is defined as the elapsed time since the occurrence time of its source event up to the current time t, including the time required for capturing and transmitting the event to the platform. Let $Y_i(t;\theta,\pi)$ denote the AoI of the copy of event e_i in the platform. It is then formulated as:*

$$Y_i(t;\theta,\pi) = \begin{cases} X_i(t-1;\theta)+1 & \text{if } \pi_i(t)=1 \\ Y_i(t-1;\theta,\pi)+1 & \text{if } \pi_i(t)=0 \end{cases} \tag{4}$$

Here, we assume that once an event is captured, its copy can be transmitted to the platform within one time slot. Actually, if the transmission of the i-th event copy takes more than one time slot, e.g., τ time slots, we only need to set $Y_i(t+\tau;\theta,\pi)=X_i(t;\pi)+\tau$ for $\pi_i(t)=1$. Our solution can still accommodate this scenario with a simple extension. Additionally, we use a vector $\omega^o(t;\theta,\pi) = (Y_1(t;\theta,\pi),\ldots,Y_N(t;\theta,\pi))$ to denote the AoI states of all event copies. For simplicity, we directly use $Y_i(t)$ and $\omega^o(t)$ hereafter, omitting the parameters θ and π when no ambiguity arises. Note that since all event copies are maintained by the platform, the vector $\omega^o(t)$ is observable.

B. Problem Formulation

In the above MCS system, at the beginning of each time slot, the platform will make the event capture decision according to the strategy π and recruits workers to capture the event at the selected PoI. Let c_i denote the cost incurred by the platform for capturing event e_i, and let B denote the average budget of the platform across all time slots. Our goal is to determine an event capture strategy π that minimizes the cumulative weighted average AoI value under the budget constraint B. Then, our problem can be formulated as follows:

$$\textbf{P1:}\quad \min_\pi \quad \frac{1}{T}\sum_{l=1}^{m}\sum_{t=1}^{L}\mathbb{E}\big(\sum_{i=1}^{N}\gamma_i Y_i(t)\big), \tag{5}$$
$$s.t. \quad \sum_{i=1}^{N}\pi_i(t)=1, \text{for any } t\in\mathcal{T}, \tag{6}$$
$$\frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{N}\pi_i(t)c_i \le B, \tag{7}$$
$$Eq.\ (2)\sim Eq.\ (4). \tag{8}$$

Here, Eq. (5) is the cumulative weighted average AoI value of the whole system, where γ_i denotes the weight assigned to event e_i, satisfying $\gamma_i\ge 0$ and $\sum_{i=1}^{N}\gamma_i=1$. The expectation is taken over the potential randomness in the AoI updating processes (Eq. (8)). Eq. (6) means that only one PoI will be selected in each time slot. Eq. (7) represents a long-term average cost constraint on event capture, following traditional stochastic optimization formulations [20]. The key notations are summarized in Table I for reference.

Problem **P1** is a long-term stochastic optimization problem with an unknown Markovian transition. Existing studies on stochastic optimization, such as [20], [21], have primarily focused on applying Lyapunov optimization methods. Although these methods are effective in many stochastic settings, they are generally designed for scenarios where random events (i.e.,

states) are observable before decision-making. However, since this assumption does not hold in our problem, these methods cannot be directly applied without significant modification.

To tackle this limitation, we reformulate the problem as a constrained restless bandit problem to model the uncertainty in event capture under resource constraints. However, classical solutions for restless bandits, such as the Whittle index policy [24], [26], further rely on the prior knowledge of state transition dynamics, which are also unknown in our case. Consequently, both traditional Lyapunov and Whittle-based methods face inherent challenges when applied to our problem. This motivates the development of an online solution that can jointly learn the unknown dynamics and make near-optimal event capture decisions under uncertainty.

III. Algorithm Design

In this section, we propose the TS-DPP algorithm to tackle the uncertain event capture problem (i.e., **P1**). Firstly, we transform the problem into a constrained restless bandit problem with unknown transition probabilities. Next, we extend the classic DPP policy, which addresses stochastic optimization problems, and propose a belief-DPP bandit policy to handle the challenge of unknown Markovian states in **P1**. Finally, we develop the TS-DPP algorithm to solve this problem. Specifically, we integrate the Thompson sampling technique to estimate the frequency of events, which enhances the belief-DPP policy's capacity to handle unknown transition dynamics by incorporating real-time probabilistic inferences.

A. Problem Transformation

Since the uncertain event capture process involves an online learning issue, we transform Problem **P1** into a budget-constrained restless bandit problem, where the platform is treated as the learner, each PoI is seen as an arm, and pulling an arm means capturing the corresponding event. In particular, the AoI state of each PoI is the state of the related arm. The learner can only pull one arm at a time. We denote the arm chosen by the learner at time t as A_t. Then, pulling the arm A_t brings a *penalty*, which is defined as the weighted average AoI value of all events, denoted by:

$$r(t)=\sum_{i=1}^{N}\gamma_i Y_i(t)=\sum_{i=1}^{N}\gamma_i Y_i(t-1)+1+\gamma_{A_t}(X_{A_t}(t)-Y_{A_t}(t-1)-1). \quad (9)$$

At the same time, it causes a *cost*:

$$c(t)=\sum\nolimits_{i=1}^{N}\pi_i(t)c_i. \quad (10)$$

Therefore, the objective becomes to minimize the time-averaged penalty, subject to a time-averaged budget constraint.

B. Belief-DPP Bandit Policy

After transforming Problem **P1**, we propose a belief-DPP bandit policy to minimize the penalty under the budget constraint. First, we denote $\omega^b(t;\theta)=(X_1^b(t),\ldots,X_N^b(t))$ as the *belief state* with respect to the transition probability θ, where $X_i^b(t)$ represents the estimate of state $X_i(t)$. Based on the belief state, we get the expected penalty $\sum_{i=1}^{N}\gamma_i\widetilde{Y}_i(t)$, where we use $\widetilde{Y}_i(t)$ as an approximation of $Y_i(t)$, i.e.,

$$\widetilde{Y}_i(t)=\pi_i(t)X_i^b(t)+(1-\pi_i(t))(Y_i(t-1)+1). \quad (11)$$

To decompose the long-term budget constraint into per-slot decisions, we construct a virtual queue $Z(t)$ to track the deviation between the cumulative cost and the budget:

$$Z(t)=\max\{0,Z(t-1)+\sum\nolimits_{i=1}^{N}\pi_i(t-1)c_i-B\}. \quad (12)$$

Here, the initial value is $Z(0)=0$. Then, the belief-DPP bandit policy is designed as follows:

Belief-DPP Bandit Policy. At the start of each time slot t, the platform observes the AoI states of event copies $\omega^o(t)$ and virtual queue $Z(t)$. After computing the belief state $\omega^b(t;\theta)$, it performs the action $\pi^*(t)$ by solving the following problems:

$$\pi^*(t)=\arg\min V\sum\nolimits_{i=1}^{N}\gamma_i\widetilde{Y}_i(t)+Z(t)(\sum\nolimits_{i=1}^{N}\pi_i(t)c_i-B), \quad (13)$$

where V is a positive tuning parameter so that we can balance the penalty and the virtual queue length. We emphasize that the belief-DPP bandit policy is a deterministic policy mapping from the transition probability θ to the decision policy π. Therefore, having an accurate parameter estimation is crucial for the effectiveness of the belief-DPP bandit policy. In the following subsection, we present the estimation of transition parameters and belief states using Thompson sampling.

C. Thompson Sampling

Before introducing the detailed TS-DPP algorithm, we need to learn transition parameters and obtain belief states based on the Thompson sampling technique.

(1) *Learning Transition Parameters:* At the beginning of the l-th episode, the platform obtains a parameter estimation of the transition probability θ, denoted by θ^l, which is drawn from a posterior distribution Q^l. Specifically, θ_i^l and Q_i^l represent the i-th arm's parameter estimation and posterior distribution, respectively. As the episode progresses, the platform observes the *history* of actions and states, which is denoted by $\mathcal{H}_t=(A_1,X_{A_1}(1),\ldots,A_t,X_{A_t}(t))$, where $1\le t\le L$.

At the end of the episode, based on the observed history $\mathcal{H}_L$, the posterior distribution is updated accordingly. Since the arms are independent of each other, we slice the history $\mathcal{H}_L$ to get the historical data of each arm. Let $\Gamma_i=\{t:A_t=i\}$ represent the set of times when the i-th arm is selected. For a specific arm i, we denote the time of j-th arm selection by $t_j^i\in\Gamma_i$, and combine it with the corresponding state $s_j^i=X_i(t_j^i)$ to form a sequence of feedback $\mathcal{S}_i=\{(t_j^i,s_j^i)\}_{j=0}^{|\Gamma_i|}$, with initial value $t_0^i=0$ and $s_0^i=0$. Afterwards, we can learn the transition parameters by updating the posterior distribution according to the following theorem.

Theorem 1. ***(Posterior Update)*** *Given a prior distribution $Q_i^l(\theta_i)=Beta(\alpha_i^l,\beta_i^l)$ and a sequence of feedback $\mathcal{S}_i=\{(t_j^i,s_j^i)\}_{j=0}^{|\Gamma_i|}$. Then, the posterior distribution is given by $Q_i^{l+1}(\theta_i|\mathcal{S}_i)=Beta(\alpha_i^{l+1},\beta_i^{l+1})$, where*

$$\alpha_i^{l+1}=\alpha_i^l+\sum\nolimits_{j=1}^{|\Gamma_i|}I_j,\ \beta_i^{l+1}=\beta_i^l+\sum\nolimits_{j=1}^{|\Gamma_i|}s_j^i-(1-I_j)s_{j-1}^i. \quad (14)$$

Here, I_j is an indicator of whether new data is obtained in the j-th selection, formally, $I_j=\mathbb{I}(s_j^i<t_j^i-t_{j-1}^i)$.

Proof. The detailed proof is provided in Appendix. □

Algorithm 1: TS-DPP for Uncertain Event Capture

input : prior Q^0, episode length L;
1 Initialize: posterior $Q^1 = Q^0$;
2 **for** *episodes* $l = 1, \ldots, m$ **do**
3 Initialize: $\mathcal{H}_0 = \emptyset$;
4 **for** $i = 1, \ldots, N$ **do**
5 Draw a parameter $\theta_i^l \sim Q_i^l$;
6 **for** $t = 1, \ldots, L$ **do**
7 Observe $\omega^o(t); Z(t)$;
8 **for** $i = 1, \ldots, N$ **do**
9 Calculate belief state $X_i^b(t)$ according to Eq. (16);
10 Select arm A_t according to Eq. (13);
11 Observe feedback $(A_t, X_{A_t}(t))$;
12 Update virtual queue $Z(t+1)$ according to Eq. (12);
13 Update record M_{t+1};
14 Update history $\mathcal{H}_t = \mathcal{H}_{t-1} \cup (A_t, X_{t,A_t})$;
15 **for** $i = 1, \ldots, N$ **do**
16 Update posterior Q_i^{l+1} according to Eq. (14);

Remark 1. This theorem indicates that the beta distribution serves as a conjugate prior in our scenario. This property allows for straightforward closed-form solutions to the posterior distribution, thereby streamlining the update process.

(2) *Obtaining the Belief State:* Due to the Markov property, in order to estimate the current state at time t (i.e. $X_i(t)$), we only need to consider the state information obtained from the most recent play. The platform stored the information of each arm in the *record*, denoted by $M_t = \{\lambda_i, \tau_i, \delta_i\}_{i=1}^N$. This indicates that for arm i, its most recent play occurred at time τ_i, and the corresponding state information is $\lambda_i = X_i(\tau_i)$. Moreover, $\delta_i = t - \tau_i$ represents the time interval between the current time t and τ_i. Utilizing the information stored in the record M_t, we compute the distribution of arm i's state with respect to λ_i and δ_i, denoted as $F_i(\theta; \lambda_i, \delta_i)$. That is, we have

$$P_x^{\theta_i} = Pr(X_i(t) = x) = \begin{cases} \theta_i(1-\theta_i)^x & \text{if } x = 0, 1, ..., \delta_i - 1, \\ (1-\theta_i)^{\delta_i} & \text{if } x = \lambda_i + \delta_i. \end{cases} \tag{15}$$

Then, we calculate the belief state $X_i^b(t)$ as the mean of this distribution, which is exhibited as follows:

$$X_i^b(t) = \frac{1-\theta_i}{\theta_i} + (1-\theta_i)^{\delta_i}(\lambda_i - \frac{1-\theta_i}{\theta_i}). \tag{16}$$

D. The Detailed TS-DPP Algorithm

Building on the above designs, we propose the TS-DPP algorithm, which integrates the belief-DPP bandit policy with the Thompson sampling technique, as summarized in Alg. 1. It consists of two time-related loops, the outer loop is composed of m inner loops, and the inner loop is formed of L rounds. The algorithm takes the prior distribution Q^0 of all transition parameters and the episodic length L as input and outputs the decision of each time slot. First, we initialize the posterior as $Q^1 = Q^0$. At the beginning of episode l, we start by initializing $\mathcal{H}_0 = \emptyset$ and draw a parameter estimation θ_i^l from posterior Q_i^l for each arm i (Lines 4-5). Then the algorithm turns to the inner loop (Lines 4-14), where we fix the parameter estimation as θ_i^l. At the round t, we observe the AoI state of event copies $\omega^o(t)$ and the virtual queue $Z(t)$. After computing the belief state $X_i^b(t)$ for each arm i (Lines 8-9), we select the arm according to belief-DPP policy (Line 10). Based on the feedback, we update virtual queue $Z(t+1)$, record M_{t+1}, and history $\mathcal{H}_t$ (Lines 12-14). To end up this episode, the algorithm updates the posterior distribution of each arm based on the history we get (Line 16). Since the belief-DPP bandit policy operates each arm independently, the time complexity of TS-DPP is $\mathcal{O}(N)$.

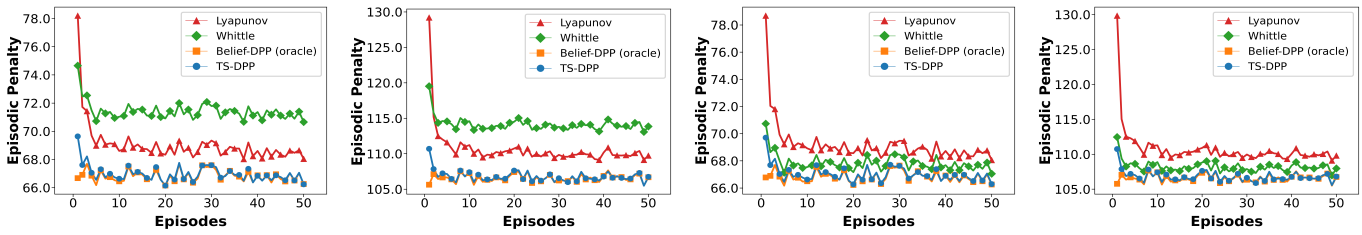
Discussion. While our main algorithm focuses on the single-PoI setting without delay, TS-DPP can be naturally extended to more general scenarios. (i) *Multiple PoIs*: We adopt a greedy selection strategy that iteratively selects PoIs one at a time. After each selection, the virtual queue is updated to reflect its impact on the budget and AoI dynamics, and the next PoI is chosen based on the updated system state. (ii) *Delayed feedback*: As discussed in Section II, our AoI model can accommodate transmission delays by setting $Y_i(t+\tau; \theta, \pi) = X_i(t; \pi) + \tau$ for a delay of τ time slots if $\pi_i(t) = 1$. However, during the intermediate delay period (i.e., $t+1$ to $t+\tau-1$), Y_i continues to increase, potentially causing the algorithm to re-select the same PoI before the previous event is delivered. To address this, we introduce a temporary freezing mechanism that marks PoI i as unavailable for selection before the event e_i is captured. This extension may lead to a slight degradation in performance due to the restricted action space, we will theoretically quantify the impact in future work.

IV. Theoretical Analysis

In this section, we analyze the theoretical performance of the TS-DPP algorithm. In restless bandit problems, if an oracle has full knowledge of both transition parameters and states, the problem becomes trivial. This is because the strategy can be directly derived from the known states [27]. Benchmarking a learning policy against such an oracle would result in regret that scales linearly with time T, since the oracle can always observe the states, while the learning policy cannot predict the transition based on the history, no matter how accurate the estimates of the transition probabilities are. Each transition introduces a non-vanishing regret, and the number of transitions grows linearly with T, leading to a linear regret.

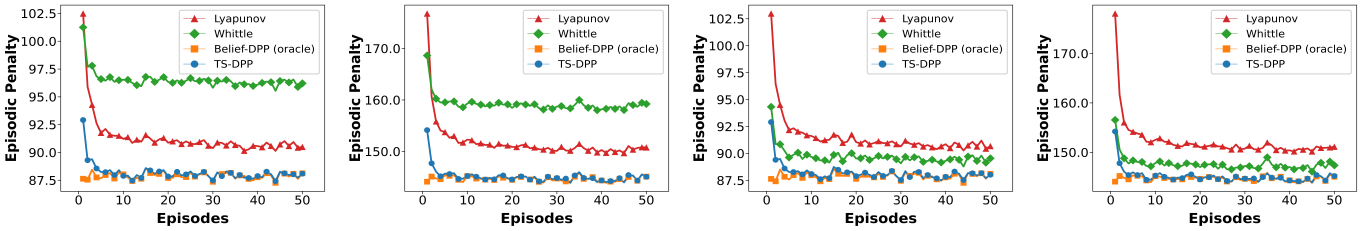
Since comparing to an oracle with state knowledge provides limited insight, we consider a weaker oracle that knows the transition parameters but not the states. This provides a more informative comparison by introducing uncertainty, while still providing useful guidance for the theoretical analysis. In this case, we use the belief-DPP bandit policy as the oracle, following the approach in previous studies [28].

Within the TS-DPP algorithm, we denote the policy adopted in the l-th episode as π^l, which incorporates the parameter estimation θ^l sampled from the posterior distribution at the episode's onset. Similarly, we denote π^* as our oracle,



(a) $L = 20, V = 0.2$ (b) $L = 30, V = 0.2$ (c) $L = 20, V = 0.5$ (d) $L = 30, V = 0.5$

Fig. 3. Episodic penalty trends for each algorithm under parameter settings: $N = 4$, $L = \{20, 30\}$, $V = \{0.2, 0.5\}$.



(a) $L = 20, V = 0.2$ (b) $L = 30, V = 0.2$ (c) $L = 20, V = 0.5$ (d) $L = 30, V = 0.5$

Fig. 4. Episodic penalty trends for each algorithm under parameter settings: $N = 8$, $L = \{20, 30\}$, $V = \{0.2, 0.5\}$.

equipped with full knowledge of the transition probability θ^*. Before defining the regret, we introduce a *value function*:

$$J^{\theta}_{\pi,t}(\mathcal{H}) = \mathbb{E}_{\theta,\pi}\big[\sum\nolimits_{j=t}^{L} r(j)|\mathcal{H}\big]. \tag{17}$$

In essence, the value function represents the expected penalty obtained by executing policy π from round t to L, given the transition probability θ and the initial history $\mathcal{H}$. Thus, we can define the Bayesian regret as

$$BR(T) = \mathbb{E}_{\theta^*\sim Q}\big[\sum\nolimits_{l=1}^{m} \mathbb{E}_{\theta^l\sim Q^l} J^{\theta^*}_{\pi^l,1}(\emptyset) - mJ^{\theta^*}_{\pi^*,1}(\emptyset)\big]. \tag{18}$$

The above expectation is with respect to the prior distribution about θ^*, and we assume the prior is known to the learner. The following theorem is our main result on regret bound.

Theorem 2. ***(Bayesian Regret Bound)*** *The Bayesian regret of TS-DPP algorithm satisfies the following bound:*

$$BR(T) = \mathcal{O}\big(\sqrt{L^3 T \log T}\big). \tag{19}$$

Proof. The detailed proof is provided in Appendix. □

Remark 2. As regret represents the cumulative gap between the value function of our algorithm and the oracle solution, the sublinear regret implies that our algorithm rapidly approaches the oracle solution. Recall that previous studies (e.g., [29]) have shown that the regret lower bound on the performance of Thompson sampling for the multi-armed bandit problem is $\Omega(\sqrt{T \log T})$, suggesting that the regret bound for TS-DPP in the online restless bandit problem is sufficiently tight.

Remark 3. Previous works (e.g. [18]) have shown that the gap between the solution obtained by the DPP algorithm and the optimal solution is $\mathcal{O}(\frac{L}{V})$, where V is the hyper-parameter in the DPP algorithm. In our episodic setting, if the hyperparameter V varies with the episode l, specifically $V_l = \sqrt{l}$, the gap becomes $\mathcal{O}(\sqrt{LT})$. Combining this with the result from Theorem 1, we conclude that the Bayesian regret of the TS-DPP algorithm relative to the optimal oracle is $\mathcal{O}(\sqrt{L^3 T \log T} + \sqrt{LT})$, which is competitive with previous algorithms in the online restless bandit problem, such as [30] (with $\tilde{\mathcal{O}}(T^{2/3})$ regret) and [28] (with $\mathcal{O}(\sqrt{T \log T})$ regret).

V. Simulation Results

In this section, we conduct extensive simulations on real-world and synthetic datasets to evaluate the performance of our proposed TS-DPP algorithm. First, we present our experimental settings. Then we display the evaluation results.

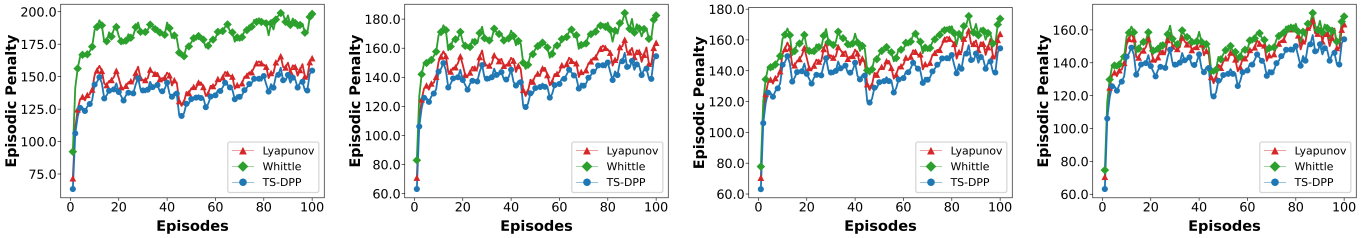
A. Experimental Setup

1) Simulation: We evaluate our approach using both real-world and synthetic datasets. For the real-world setting, we adopt an eCommerce event history dataset [31]. Due to varying event frequencies, not all events are suitable for evaluation. We exclude one type of infrequent event and retain the three event types, totaling 1,654,771 occurrences. Each event's timestamp is mapped to a uniform 1-second time slots, which serve as its actual occurrence time. We run $m = 100$ episodes for this setting. For the synthetic setting, we consider $N = 4, 8$ PoIs and run $m = 50$ episodes. For each arm i, event occurrences in each round follow a Bernoulli distribution with success probability θ_i, where θ_i is uniformly sampled from $[0, 1]$ (i.e., $\theta_i \sim \text{Beta}(1, 1)$). All simulations adopt the following shared configurations: the weights of AoI value γ_i for each arm are drawn from a Dirichlet distribution such that $\sum_{i=1}^{N} \gamma_i = 1$. The observed cost c_i is uniformly sampled from $[0.2, 0.5]$. The budget is fixed at $B = 0.35$, which equals the expected cost of uniformly random arm selection.

2) Baselines: In order to highlight the approximate optimality of the TS-DPP algorithm, we compare its performance against the offline oracle policy (i.e., the belief-DPP bandit policy). This offline oracle policy provides an upper bound on the performance that can be achieved under ideal conditions with perfect knowledge of the environment. In addition, we also evaluate the TS-DPP algorithm against several baselines: i) the Lyapunov index method as described in [20], and ii) the Whittle index approach detailed in [24].

B. Performance Results

1) Comparing to baselines: We evaluate TS-DPP on both synthetic and real-world datasets to comprehensively assess



(a) $L = 20, V = 0.04$ (b) $L = 20, V = 0.06$ (c) $L = 20, V = 0.08$ (d) $L = 20, V = 0.1$

Fig. 5. Episodic penalty trends for each algorithm under parameter settings: $N = 3$, $L = 20$, $V = \{0.04, 0.06, 0.08, 0.1\}$.

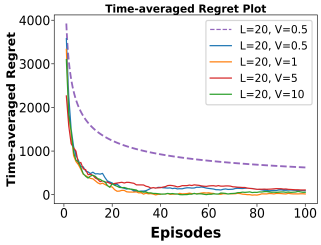
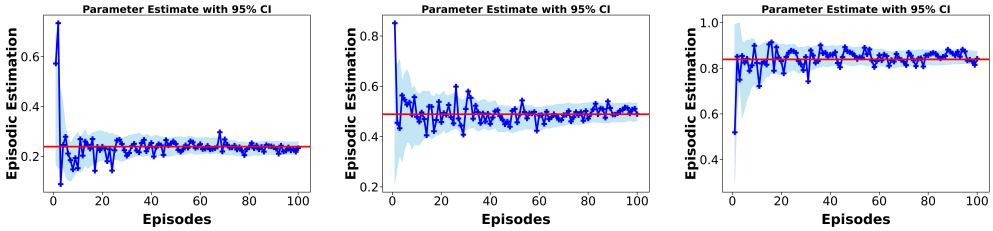


Fig. 6. Time-averaged regret.



(a) $\theta = 0.239$ (b) $\theta = 0.489$ (c) $\theta = 0.839$

Fig. 7. The parameter estimates trend plots of three categories of probabilities.

its effectiveness. In the synthetic experiments, we display the episodic penalty trends for each algorithm under various parameter settings, as illustrated in Fig. 3 and Fig. 4. Specifically, we choose the hyperparameters $V = 0.2, 0.5$, and the episode length $L = 20, 30$ for two experiments with $N = 4, 8$ PoIs. The results show that the TS-DPP algorithm consistently outperforms the Lyapunov index and Whittle index algorithms in all experiments. Moreover, TS-DPP shows a remarkable ability to closely approximate the performance of the oracle policy after a few episodes. This indicates that TS-DPP not only performs well compared to other baselines but also demonstrates a strong capability to approach the theoretical optimum provided by the oracle policy. To further demonstrate the practicality of TS-DPP, we evaluate its performance on a real-world dataset under varying values of the hyperparameter $V \in \{0.04, 0.06, 0.08, 0.10\}$. As illustrated in Fig. 5, TS-DPP consistently achieves the lowest episodic penalties among all algorithms. Moreover, TS-DPP exhibits strong robustness to different hyperparameters, maintaining a clear performance advantage over the baselines. In contrast, the performance of the Lyapunov and Whittle index methods is more sensitive to hyperparameters and shows greater variability.

2) Bayesian regret: We analyze the Bayesian regret of the TS-DPP algorithm with different settings. For this analysis, we fix the episode length at $L = 20$ and the number of episodes at $m = 100$. To approximate the expectation, we use a Monte Carlo simulation with a sample size of 1000. The time-averaged regret of the TS-DPP algorithm with different hyperparameters is shown in Fig. 6. The results indicate that the time-averaged regret of the TS-DPP algorithm converges to zero faster than the upper bound we established. This is an encouraging outcome, as it shows that the TS-DPP algorithm not only performs well in practice but also adheres closely to theoretical performance guarantees. The rapid convergence to zero regret signifies that the TS-DPP algorithm is effectively learning and adapting, minimizing the performance gap between itself and the optimal policy over time.

3) The efficiency of Thompson sampling: We validate the effectiveness of Thompson sampling by examining parameter estimation plots. For this analysis, we randomly select several PoIs from the previous simulations and evaluate their historical parameter estimations θ^l alongside the historical parameters of the beta distribution, i.e., α^l and β^l. These results are illustrated in Fig. 7. In the figures, the red lines represent the true parameter values, while the blue dots and lines depict the trends of the parameter estimations over time. The shaded regions indicate the 95% confidence intervals of the beta distribution at specific rounds. To provide a comprehensive analysis, we categorize the probabilities into three representative groups: large, medium, and small values. As shown in Fig. 7, during the initial episodes, the Thompson sampling framework is predominantly in the exploration phase. Consequently, the estimations of the transition probability are not yet close to the true values. However, as more episodes are completed, the parameter estimations begin to stabilize and approximate the true values. This phenomenon aligns with the observations in Fig. 3 and Fig. 4, where all algorithms initially incur high penalties. These penalties gradually decrease and stabilize after a few episodes, reflecting the transition from exploration to exploitation as the Thompson sampling framework refines its estimations and improves its performance over time.

VI. Related Work

We review the related works from the following aspects:

Mobile Crowdsensing. Recently, MCS has been widely investigated in many fields, like urban crowdsensing with for-hire vehicles and sparse crowdsensing [5], [7], [32]. Efforts to minimize AoI and improve data freshness in MCS have been relatively limited. Dai *et al.* in [16] considered an MCS system where mobile agents are scheduled to collect data, aiming to minimize the AoI of all sensor nodes while considering energy consumption. Xu *et al.* in [3] proposed a Stackelberg-game-based incentive mechanism for MCS accounting for AoI values of collected data and social benefits.

Age of Information. AoI has been widely studied across various applications, including wireless networks [33], mobile networks [34], and federated learning [19], [35]. The AoI models are typically categorized into push and pull paradigms [36]. The push model (e.g., queueing systems [10]–[12]) focuses on determining when and how to push (i.e., generate and transmit) updates, while the pull model (e.g., channel systems [37] and status update systems [13]) determines when and how to request updates. However, our model deviates from both paradigms, as it integrates event capture (a form of pull operation) into the decision-making process, while explicitly considering the timing of data generation. Prior works have applied the restless bandit framework for AoI minimization [22]–[25], [35] and developed Whittle index policies under the assumption of known transition dynamics. Additionally, Kadota *et al.* in [33] developed a DPP policy but their approach also assumes known transition dynamics. More recent studies have explored unknown state transitions in AoI modeling, yet they mainly concentrate on simplified settings and lack time-averaged cost constraints [38]. In contrast, our work develops a learning-based policy under unknown transition dynamics, providing a more general and practical approach for event-driven AoI minimization.

Restless Bandit. Restless bandit [26] is a variant of the multi-armed bandit [39] problem and addresses scenarios where arms have non-stationary reward distributions. A widely studied approach to solving restless bandit problems is the Whittle index [40], which provides an index-based heuristic for making provably asymptotically optimal decisions. The learning perspective of restless bandit problems, named online restless bandits, has also gained attention. Jung *et al.* in [28] proposed a TS-based method in episodic restless bandits under binary states, achieving a Bayesian regret bound of $\mathcal{O}(\sqrt{T\log T})$. Wang *et al.* in [30] designed Restless-UCB and achieved frequentist regret $\mathcal{O}(T^{2/3})$. Jiang *et al.* in [41] proposed TSEETC and established the Bayesian regret bound $\tilde{\mathcal{O}}(\sqrt{T})$. In a constrained restless bandit area, Wei *et al.* in [20] proposed a Lyapunov indexing approach to solve the restless bandit problem with time-averaged constraints, assuming known states. In contrast to the above works, our method tackles the online constrained restless bandit problem under unknown and unobservable states and transitions, and we propose a novel learning-based scheduling strategy.

Event Capture. Several studies focus on event capture across different domains, e.g., data stream [42], transportation [43], and healthcare [44], etc. Lamprier *et al.* [42] proposed a contextual bandit approach that utilizes users' current activities to predict future behavior. Noursalehi *et al.* [43] developed a deep learning-based framework for real-time prediction of transit passenger demand considering the spatial and temporal dependencies between transit stations and the influence of special events. Duan *et al.* [44] employed graph convolutional networks and pose estimation to facilitate real-time motion capture for athlete performance analysis. These studies neglect the freshness of the captured information. In contrast, our work is the first to address AoI-aware MCS in the context of uncertain event capture.

VII. Conclusion

In this paper, we explore the AoI-aware MCS system for uncertain event capture applications, where events may occur frequently but uncertainly, making the AoI update hard to be estimated. First, we model this uncertain event capture problem as a constrained episodic restless bandit problem with unknown transition probability. Next, we propose a belief-DPP bandit policy by extending the DPP policy. By combining belief-DPP and the Thompson sampling technique, we further propose the TS-DPP algorithm, so as to minimize the cumulative weighted AoI values of all events under a given budget constraint. We analyze the theoretical performance of the TS-DPP algorithm, and derive a sublinear Bayesian regret bound $\mathcal{O}(\sqrt{T\log T})$, where T is the size of time horizon. Additionally, we conduct extensive simulations to demonstrate the significant performance of the TS-DPP algorithm.

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grants 62436010, 62502483, and 62172386, in part by the Fundamental Research Funds for the Central Universities under Grant WK2150250040, and in part by the Natural Science Foundation of Jiangsu Province in China under Grant BK20231212. The corresponding authors are Yin Xu, Mingjun Xiao, and Jie Wu.

Appendix

Proof of Theorem 1: We define the time interval between the j-th and the $(j-1)$-th selection as $n_j = t_j^i - t_{j-1}^i$. Then, we compute the conditional distribution of s_j^i given s_{j-1}^i and n_j:

$$Pr(s_j^i|s_{j-1}^i, n_j; \theta_i) = \begin{cases} \theta_i(1-\theta_i)^{s_j^i} & \text{if } s_j^i = 0, 1, ..., n_j - 1, \\ (1-\theta_i)^{n_j} & \text{if } s_j^i = n_j + s_{j-1}^i. \end{cases} \tag{20}$$

Using the indicator I_j, we get a more general formula:

$$Pr(s_j^i|s_{j-1}^i, n_j; \theta_i) = \theta_i^{I_j}(1-\theta_i)^{s_j^i - (1-I_j)s_{j-1}^i}. \tag{21}$$

Based on Bayesian theory, we get:

$$\begin{aligned} &Q_i^{l+1}(\theta_i|\mathcal{S}_i) \propto Q_i^l(\theta_i) \cdot \Pi_{j=1}^{|\Gamma_i|} Pr(s_j^i|s_{j-1}^i, n_j; \theta_i) \\ &\propto \theta_i^{\alpha_i^l - 1} \cdot (1-\theta_i)^{\beta_i^l - 1} \cdot \theta_i^{\sum_{j=1}^{|\Gamma_i|} I_j} \cdot (1-\theta_i)^{\sum_{j=1}^{|\Gamma_i|} s_j^i - (1-I_j)s_{j-1}^i} \\ &\propto \theta_i^{\alpha_i^l - 1 + \sum_{j=1}^{|\Gamma_i|} I_j} \cdot (1-\theta_i)^{\beta_i^l - 1 + \sum_{j=1}^{|\Gamma_i|} s_j^i - (1-I_j)s_{j-1}^i} \\ &\propto Beta(\alpha_i^{l+1}, \beta_i^{l+1}). \end{aligned} \tag{22}$$

The proof of Theorem 1 is completed.

Proof of Theorem 2: A key idea in our proof centers around two aspects. First, we measure the distance between the true parameter θ^* and the parameter estimation θ^l. Due to the Thompson sampling technique, the distribution of θ^* and θ^l is identical given the same history [39]. As the historical data is used to update the posterior distribution, the variance decreases, indicating a concentration of θ^* and θ^l. We can utilize concentration inequalities to quantify their proximity.

Then, we evaluate the difference in penalties obtained by executing the same strategy under these two parameters. We decompose the Bayesian regret into individual rounds by

leveraging the conclusion from [28], which are presented in Lemma 3 and Lemma 4. It is worth noting that this conclusion is established in the case of *deterministic policy*.

Definition 4 (***deterministic policy [28]***). *A deterministic policy π takes time index and history $(t, \mathcal{H}_{t-1})$ as an input and outputs a fixed action $A_t = \pi(t, \mathcal{H}_{t-1})$. A deterministic policy mapping μ takes a transition probability θ as an input and outputs a deterministic policy $\pi = \mu(\theta)$.*

Lemma 1. *The belief-DPP bandit policy is a deterministic policy mapping.*

Lemma 2. *Given history $\mathcal{H}_{t-1}$, then record M_t is deterministic. Therefore, for any arm i, the distance between distribution $F_i(\theta; \lambda_i, \delta_i)$ and $F_i(\theta'; \lambda_i, \delta_i)$ can be represented as follows:*

$$\textstyle\sum_x |P_x^{\theta_i} - P_x^{\theta_i'}| \le 2\delta_i |\theta_i - \theta_i'|, \tag{23}$$

$$\textstyle\sum_x x|P_x^{\theta_i} - P_x^{\theta_i'}| \le (\frac{\delta_i(\delta_i-1)}{2} + \delta_i(\lambda_i + \delta_i))|\theta_i - \theta_i'|. \tag{24}$$

Proof. If $x \in \{0, 1, \ldots, n_i - 1\}$, let $f(\theta) = \theta(1-\theta)^x$. Then, by applying the mean value theorem, there exists ξ between θ and θ' such that the following equation holds:

$$\begin{aligned} &|P_x^{\theta_i} - P_x^{\theta_i'}| = |f(\theta_i) - f(\theta_i')| = |f'(\xi)| \cdot |\theta_i - \theta_i'| \\ &= |(1-\xi)^{x-1}(1-(x+1)\xi)| \cdot |\theta_i - \theta_i'| \le |\theta_i - \theta_i'|. \end{aligned} \tag{25}$$

If $x = s_i + n_i$, let $f(\theta) = \theta(1-\theta)^{n_i}$. Likewise, we obtain:

$$\begin{aligned} &|P_x^{\theta_i} - P_x^{\theta_i'}| = |f(\theta_i) - f(\theta_i')| = |f'(\xi)| \cdot |\theta_i - \theta_i'| \\ &= |\delta_i(1-\xi)^{\delta_i - 1}| \cdot |\theta_i - \theta_i'| \le \delta_i |\theta_i - \theta_i'|. \end{aligned} \tag{26}$$

When summing up for all x, we complete the proof. □

Lemma 3 (***Regret Decomposition [28]***). *The Bayesian regret of the TS-DPP algorithm can be decomposed as*

$$BR(T) = \mathbb{E}_{\theta^* \sim Q} \textstyle\sum_{l=1}^m \mathbb{E}_{\theta^l \sim Q^l}\big[J_{\pi^l,1}^{\theta^*}(\emptyset) - J_{\pi^*,1}^{\theta^*}(\emptyset)\big] \tag{27}$$

$$= \mathbb{E}_{\theta^* \sim Q} \textstyle\sum_{l=1}^m \mathbb{E}_{\theta^l \sim Q^l}\big[J_{\pi^l,1}^{\theta^*}(\emptyset) - J_{\pi^l,1}^{\theta^l}(\emptyset)\big]. \tag{28}$$

Eq. (27) describes the difference caused by adopting policies π^l and π^* under the same transition probability θ^*. However, Eq. (28) describes the difference resulting from applying the same policy π^l on different systems with parameters θ^* and θ^l. This transformation in the proof provides significant convenience. Subsequently, we aim to further decompose the regret of each episode into individual rounds. Before presenting the next lemma, we first define the Bellman operator:

$$\mathcal{T}_\pi^\theta J(\mathcal{H}_{t-1}) = \mathbb{E}_{\theta,\pi}\big[r(t) + J(\mathcal{H}_t)|\mathcal{H}_{t-1}\big]. \tag{29}$$

Lemma 4 (***Per-episode Regret Decomposition [28]***). *Fix θ^* and θ^l, and let $\mathcal{H}_0 = \emptyset$. Then we have*

$$J_{\pi^l,1}^{\theta^*}(\mathcal{H}_0) - J_{\pi^l,1}^{\theta^l}(\mathcal{H}_0) = \mathbb{E}_{\theta^*,\pi^l} \textstyle\sum_{t=1}^L (\mathcal{T}_{\pi^l}^{\theta^*} - \mathcal{T}_{\pi^l}^{\theta^l}) J_{\pi^l,t+1}^{\theta^l}(\mathcal{H}_{t-1}). \tag{30}$$

Lemma 5 (***Concentration Inequality [45]***). *If X has a beta distribution $Beta(\alpha, \beta)$ and Y is an independent copy of X, then, for all $0 \le r \le 1$, we have*

$$Pr(|X - Y| \ge r\mathbb{E}(X)) \le 2e^{-\alpha r^2/8}. \tag{31}$$

We fix transition probability θ^* and θ^l and try to analyze the regret in the episode l. It is worth noting that according to the key principle of the Thompson sampling technique, θ_i^* and θ_i^l are drawn from the same distribution $Q_i^l = Beta(\alpha_i^l, \beta_i^l)$ [39]. Define parameter space as

$$\Theta_i = \big\{(\theta_i^*, \theta_i^l) | \theta_i^* \sim Q_i^l, \theta_i^l \sim Q_i^l\big\}. \tag{32}$$

Then, we define an event ε_i for arm i as

$$\varepsilon_i = \big\{(\theta_i^*, \theta_i^l) \in \Theta_i | |\theta_i^* - \theta_i^l| < r_i^l \cdot \frac{\alpha_i^l}{\alpha_i^l + \beta_i^l}\big\}, \tag{33}$$

where $r_i^l = \max\big\{1, \sqrt{\frac{8\log T}{\alpha_i^l}}\big\}$. Using the concentration inequality in Lemma 5, we get $Pr(\varepsilon_i^c) \le 2e^{-\alpha_i^l {r_i^l}^2/8}$. Now, we pay attention to the Bellman operator:

$$\begin{aligned} &\mathcal{T}_{\pi^l}^\theta J_{\pi^l,t+1}^{\theta^l}(\mathcal{H}_{t-1}) = \mathbb{E}_{\theta,\pi^l}\big[r(t) + J_{\pi^l,t}^{\theta^l}(\mathcal{H}_t)|\mathcal{H}_{t-1}\big] \\ &= \mathbb{E}_{\theta,\pi^l}\big[r(t)|\mathcal{H}_{t-1}\big] + \textstyle\sum_x P_x^{\theta_{A_t}} J_{\pi^l,t}^{\theta^l}\big(\mathcal{H}_{t-1} \cup (A_t, x)\big). \end{aligned} \tag{34}$$

We then get the difference between different parameters.

$$\begin{aligned} &\mathbb{E}_{\theta^*,\pi^l}\big[r(t)|\mathcal{H}_{t-1}\big] - \mathbb{E}_{\theta^l,\pi^l}\big[r(t)|\mathcal{H}_{t-1}\big] \\ &= \gamma_{A_t}\big(\mathbb{E}_{\theta^*}[X_{A_t}(t) - Y_{A_t}(t-1) - 1] - \mathbb{E}_{\theta^l}[X_{A_t}(t) - Y_{A_t}(t-1) - 1]\big) \\ &= \gamma_{A_t}\big(\mathbb{E}_{\theta^*}[X_{A_t}(t)] - \mathbb{E}_{\theta^l}[X_{A_t}(t)]\big) \\ &= \gamma_{A_t}\big(\textstyle\sum_x x \cdot P_x^{\theta_{A_t}^*} - \sum_x x \cdot P_x^{\theta_{A_t}^l}\big). \end{aligned} \tag{35}$$

The above equations are based on the fact that the same decision A_t is generated, even though the transition probabilities may differ. This is because the policy π^l is a deterministic policy under the same history $\mathcal{H}_{t-1}$.

Lemma 6. *For any $\epsilon > 0$, with probability at least $1 - \epsilon$, it holds that $X_i(t) \le \lceil \frac{\log \epsilon}{\log(1-\theta_i)} \rceil$. Therefore, for analytical simplicity, we consider the state space of $X_i(t)$ to be finite, i.e., $X_i(t) \le D$, for any arm i. Furthermore, we assume that the time intervals for selecting the same arm are finite, i.e., $\delta_i \le C$. Then, we conclude the event is captured within $C + D$ rounds, which implies that $Y_i(t) \le C + D$ holds for any arm, and we get $r_t \le C + D$.*

Corollary 1. *In an episode, for any arm, its new data can be obtained at least $\eta = \lfloor \frac{L}{C+D} \rfloor$ times, moreover, $\alpha_i^l \ge \eta l$. In the subsequent analysis, we assume that L is sufficiently large, ensuring that $\eta > 0$.*

Therefore, using Lemma 2, we get:

$$\big|\mathbb{E}_{\theta^*,\pi^l}[r(t)|\mathcal{H}_{t-1}] - \mathbb{E}_{\theta^l,\pi^l}[r(t)|\mathcal{H}_{t-1}]\big| \le (\tfrac{1}{2}C^2 + LC)|\theta_{A_t}^* - \theta_{A_t}^l|, \tag{36}$$

$$\big|\textstyle\sum_x (P_x^{\theta_{A_t}^*} - P_x^{\theta_{A_t}^l}) J_{\pi^l,t}^{\theta^l}(\mathcal{H}_{t-1} \cup (A_t, x))\big| \le 2C(C+D)L|\theta_{A_t}^* - \theta_{A_t}^l|. \tag{37}$$

When the event ε_{A_t} happens, we have $|\theta_{A_t}^* - \theta_{A_t}^l| < \sqrt{\frac{8\log T}{\alpha_{A_t}^l}}$. So,

$$\begin{aligned} &\mathbb{E}_{\theta^*,\pi^l} \textstyle\sum_{t=1}^L \big|\theta_{A_t}^* - \theta_{A_t}^l\big| \le \sum_{t=1}^L \Big(\sqrt{\frac{8\log T}{\alpha_{A_t}^l}} Pr[\varepsilon_{A_t}] + 1 \cdot Pr[\varepsilon_{A_t}^c]\Big) \\ &\le \sum_{t=1}^L \Big(\sqrt{\frac{8\log T}{\alpha_{A_t}^l}} + 2e^{-\alpha_{A_t}^l {r_{A_t}^l}^2/8}\Big) \le L\sqrt{\frac{8\log T}{\eta l}} + \sum_{t=1}^L 2e^{-\alpha_{A_t}^l {r_{A_t}^l}^2/8}. \end{aligned}$$

Combined with Eq. (36) and Eq. (37), we use the regret decomposition in Lemma 3 and Lemma 4, and then get

$$\begin{aligned} &BR(T) \le \big(\tfrac{1}{2}C^2 + (2C^2 + 2CD + C)L\big) \\ &\Big(\textstyle\sum_{l=1}^m L\sqrt{\frac{8\log T}{\eta l}} + \sum_{l=1}^m \sum_{t=1}^L 2e^{-\alpha_{A_t}^l {r_{A_t}^l}^2/8}\Big). \end{aligned} \tag{38}$$

The first term is bounded by

$$\textstyle\sum_{l=1}^m L\sqrt{\frac{8\log T}{\eta l}} \le L\sqrt{\frac{8\log T}{\eta}} \cdot 2\sqrt{m}. \tag{39}$$

Regarding the second term, it is necessary to discuss the case when r_i^l equals 1. Since for any arm i, we have $\alpha_i^l \geqslant \eta l$, it can be deduced that r_i^l is less than 1 after $n = \lceil \frac{8 \log T}{\eta} \rceil$ episodes. So, we conclude that there exists

$$\begin{aligned} &\textstyle\sum_{l=1}^{m}\sum_{t=1}^{L} 2e^{-\alpha_{A_t}^l {r_{A_t}^l}^2/8} \leq \sum_{l=1}^{n}\sum_{t=1}^{L} 2e^{-\alpha_{A_t}^l/8} + \frac{2m}{T} \\ &\leq \textstyle\sum_{l=1}^{n} L \cdot 2e^{-\eta l/8} + \frac{2m}{T} \leq 1 - \frac{16}{\eta} e^{-\eta/8} \frac{1}{T} + \frac{2m}{T}. \end{aligned} \tag{40}$$

By summing Eq. (39) and Eq. (40), we get

$$\begin{aligned} &BR(T) \leq \left(\tfrac{1}{2}C^2 + (2C^2 + 2CD + C)L\right) \\ &\left(L\sqrt{\tfrac{32m\log T}{\eta}} + 1 - \tfrac{16}{\eta} e^{-\eta/8} \tfrac{1}{T} + \tfrac{2m}{T}\right) = \mathcal{O}\left(\sqrt{L^3 T \log T}\right). \end{aligned}$$

Now, we complete the proof of Theorem 2.

REFERENCES

[1] J. Wang, F. Wang, Y. Wang, D. Zhang, L. Wang, and Z. Qiu, "Social-network-assisted worker recruitment in mobile crowd sensing," *IEEE Transactions on Mobile Computing*, vol. 18, no. 7, pp. 1661–1673, 2019.

[2] M. H. Cheung, F. Hou, and J. Huang, "Make a difference: Diversity-driven social mobile crowdsensing," in *INFOCOM*, 2017.

[3] Y. Xu, M. Xiao, Y. Zhu, J. Wu, S. Zhang, and J. Zhou, "Aoi-guaranteed incentive mechanism for mobile crowdsensing with freshness concerns," *IEEE Transactions on Mobile Computing*, vol. 23, no. 5, pp. 4107–4125, 2024.

[4] C. H. Liu, Z. Dai, H. Yang, and J. Tang, "Multi-task-oriented vehicular crowdsensing: A deep learning approach," in *INFOCOM*, 2020.

[5] A. Scalingi, D. Giustiniano, R. Calvo-Palomino, N. Apostolakis, G. Bovet, *et al.*, "A framework for wireless technology classification using crowdsensing platforms," in *INFOCOM*, 2023.

[6] X. Li, K. Xie, J. Wen, G. Zhang, W. Liang, G. Xie, and K. Li, "Joint neural matrix completion for multi-attribute mobile crowd sensing," in *INFOCOM*, 2025.

[7] Y. Xu, J. Liu, E. Wang, B. Yang, D. Luan, Y. Yang, and J. Deng, "Rethinking the effect of sparse data completion on sparse mobile crowdsensing tasks," *IEEE Transactions on Mobile Computing*, vol. 24, no. 6, pp. 5094–5105, 2025.

[8] S. Kaul, R. Yates, and M. Gruteser, "Real-time status: How often should one update?," in *INFOCOM*, 2012.

[9] R. D. Yates, Y. Sun, D. R. Brown, S. K. Kaul, E. Modiano, and S. Ulukus, "Age of information: An introduction and survey," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 5, pp. 1183–1210, 2021.

[10] A. M. Bedewy, Y. Sun, and N. B. Shroff, "Minimizing the age of information through queues," *IEEE Transactions on Information Theory*, vol. 65, no. 8, pp. 5215–5232, 2019.

[11] A. M. Bedewy, Y. Sun, S. Kompella, and N. B. Shroff, "Optimal sampling and scheduling for timely status updates in multi-source networks," *IEEE Transactions on Information Theory*, vol. 67, no. 6, pp. 4019–4034, 2021.

[12] E. Erbayat, A. Maatouk, P. Zou, and S. Subramaniam, "Age of information optimization and state error analysis for correlated multi-process multi-sensor systems," in *Mobihoc*, 2024.

[13] Z. Liu, K. Zhang, B. Li, Y. Sun, Y. T. Hou, and B. Ji, "Learning-augmented online minimization of age of information and transmission costs," in *INFOCOM WKSHPS*, 2024.

[14] Z. Zhao and I. Kadota, "Optimizing age of information without knowing the age of information," in *INFOCOM*, 2025.

[15] S. Wang and Y. Cheng, "Deep learning-augmented shs model for accurate aoi analysis in heterogeneous unsaturated csma networks," in *INFOCOM*, 2025.

[16] Z. Dai, H. Wang, C. H. Liu, R. Han, J. Tang, and G. Wang, "Mobile crowdsensing for data freshness: A deep reinforcement learning approach," in *INFOCOM*, 2021.

[17] Z. Dai, C. H. Liu, Y. Ye, R. Han, Y. Yuan, G. Wang, and J. Tang, "Aoi-minimal uav crowdsensing by model-based graph convolutional reinforcement learning," in *INFOCOM*, 2022.

[18] M. Neely, *Stochastic network optimization with application to communication and queueing systems*. Springer Nature, 2022.

[19] Y. Cui, N. Ding, and M. H. Cheung, "Aoi-aware federated unlearning for streaming data with online client selection and pricing," in *INFOCOM*, 2025.

[20] X. Wei and M. J. Neely, "Power-aware wireless file downloading: A lyapunov indexing approach to a constrained restless bandit problem," *IEEE/ACM Transactions on Networking*, vol. 24, no. 4, pp. 2264–2277, 2015.

[21] F. Li, J. Liu, and B. Ji, "Combinatorial sleeping bandits with fairness constraints," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 3, pp. 1799–1813, 2019.

[22] J. Sun, Z. Jiang, B. Krishnamachari, S. Zhou, and Z. Niu, "Closed-form whittle's index-enabled random access for timely status update," *IEEE Transactions on Communications*, vol. 68, no. 3, pp. 1538–1551, 2019.

[23] A. Maatouk, S. Kriouile, M. Assaad, and A. Ephremides, "Asymptotically optimal scheduling policy for minimizing the age of information," in *ISIT*, 2020.

[24] Y.-P. Hsu, "Age of information: Whittle index for scheduling stochastic arrivals," in *ISIT*, 2018.

[25] V. Tripathi and E. Modiano, "A whittle index approach to minimizing functions of age of information," in *Allerton*, 2019.

[26] P. Whittle, "Restless bandits: Activity allocation in a changing world," *Journal of applied probability*, vol. 25, no. A, pp. 287–298, 1988.

[27] X. Zhou, Y. Xiong, N. Chen, and X. Gao, "Regime switching bandits," in *NeurISP*, 2021.

[28] Y. H. Jung and A. Tewari, "Regret bounds for thompson sampling in episodic restless bandit problems," in *NeurIPS*, 2019.

[29] S. Agrawal and N. Goyal, "Further optimal regret bounds for thompson sampling," in *AISTATS*, 2013.

[30] S. Wang, L. Huang, and J. Lui, "Restless-ucb, an efficient and low-complexity algorithm for online restless bandits," in *NeurIPS*, 2020.

[31] "ecommerce events history," 2020. [Online]. Available: https://www.kaggle.com/datasets/mkechinov/ecommerce-events-history-in-cosmetics-shop.

[32] E. Wang, M. Zhang, B. Yang, Y. Yang, and J. Wu, "Large-scale spatiotemporal fracture data completion in sparse crowdsensing," *IEEE Transactions on Mobile Computing*, vol. 23, no. 7, pp. 7585–7601, 2024.

[33] I. Kadota, A. Sinha, and E. Modiano, "Scheduling algorithms for optimizing age of information in wireless networks with throughput constraints," *IEEE/ACM Transactions on Networking*, vol. 27, no. 4, pp. 1359–1372, 2019.

[34] M. Zhang, H. H. Yang, A. Arafa, and H. V. Poor, "Age of information in mobile networks: Fundamental limits and tradeoffs," in *Mobihoc*, 2024.

[35] C. Wu, M. Xiao, J. Wu, Y. Xu, J. Zhou, and H. Sun, "Towards federated learning on fresh datasets," in *MASS*, 2023.

[36] F. Li, Y. Sang, Z. Liu, B. Li, H. Wu, and B. Ji, "Waiting but not aging: Optimizing information freshness under the pull model," *IEEE/ACM Transactions on Networking*, vol. 29, no. 1, pp. 465–478, 2021.

[37] J. Pan, A. M. Bedewy, Y. Sun, and N. B. Shroff, "Minimizing age of information via scheduling over heterogeneous channels," in *Mobihoc*, 2021.

[38] Y.-P. Hsu, E. Modiano, and L. Duan, "Scheduling algorithms for minimizing age of information in wireless broadcast networks with random arrivals," *IEEE Transactions on Mobile Computing*, vol. 19, no. 12, pp. 2903–2915, 2019.

[39] T. Lattimore and C. Szepesvári, *Bandit algorithms*. Cambridge University Press, 2020.

[40] G. Xiong and J. Li, "Finite-time analysis of whittle index based q-learning for restless multi-armed bandits with neural network function approximation," in *NeurIPS*, 2023.

[41] B. Jiang, B. Jiang, J. Li, T. Lin, X. Wang, and C. Zhou, "Online restless bandits with unobserved states," in *ICML*, 2023.

[42] S. Lamprier, T. Gisselbrecht, and P. Gallinari, "Contextual bandits with hidden contexts: A focused data capture from social media streams," *Data Mining and Knowledge Discovery*, vol. 33, no. 6, pp. 1853–1893, 2019.

[43] P. Noursalehi, H. N. Koutsopoulos, and J. Zhao, "Real time transit demand prediction capturing station interactions and impact of special events," *Transportation Research Part C: Emerging Technologies*, vol. 97, pp. 277–300, 2018.

[44] C. Duan, B. Hu, W. Liu, and J. Song, "Motion capture for sporting events based on graph convolutional neural networks and single target pose estimation algorithms," *Applied Sciences*, vol. 13, no. 13, p. 7611, 2023.

[45] S. Bobkov and M. Ledoux, *One-dimensional empirical measures, order statistics, and Kantorovich transport distances*, vol. 261. American Mathematical Society, 2019.