Actively Tracking the Optimal Arm in Non-Stationary Environments with Mandatory Probing

Gourab Ghatak, *Member, IEEE*

***Abstract*—We study a novel multi-armed bandit (MAB) setting which mandates the agent to probe all the arms periodically in a non-stationary environment. In particular, we develop `TS-GE` that balances the regret guarantees of classical Thompson sampling (TS) with the broadcast probing (BP) of all the arms simultaneously in order to actively detect a change in the reward distributions. Once a system-level change is detected, the changed arm is identified by an optional subroutine called group exploration (GE) which scales as $\log_2(K)$ for a $K-$armed bandit setting. We characterize the probability of missed detection and the probability of false-alarm in terms of the environment parameters. The latency of change-detection is upper bounded by $\sqrt{T}$ while within a period of $\sqrt{T}$, all the arms are probed at least once. We highlight the conditions in which the regret guarantee of `TS-GE` outperforms that of the state-of-the-art algorithms, in particular, `ADSWITCH` and `M-UCB`. Furthermore, unlike the existing bandit algorithms, `TS-GE` can be deployed for applications such as timely status updates, critical control, and wireless energy transfer, which are essential features of next-generation wireless communication networks. We demonstrate the efficacy of `TS-GE` by employing it in a n industrial internet-of-things (IIoT) network designed for simultaneous wireless information and power transfer (SWIPT).**

***Impact Statement*—Most practical environments that reinforcement learning algorithms target are non-stationary in nature. In case the number of choices are large, detecting non-stationarity, especially in the sub-optimal arms is non-trivial, and hence the state-of-the-art algorithms incur a regret that scales as a product of $\sqrt{K}$ and $\sqrt{T \log T}$ for a K-armed bandit setting. This article develops a novel bandit algorithm that not only detects the changed-arm in $\mathcal{O}(\log K)$, but also it ensures periodic probing of all the arms in the environment. In particular, unlike existing algorithms, our proposal ensures an upper bound of $\sqrt{T}$ for the age of probing each arm. This feature makes it an attractive solution for applications where not only the current best-arm needs to be identified but also all the arms need to be interacted with periodically. The proposed algorithm will find applications in wireless communications (e.g., simultaneous wireless information and power transfer), in portfolio optimization (e.g., hedging across multiple instruments), and computational advertisement (e.g., building user profiles while maximizing revenue).**

***Index Terms*—Multi-armed bandits, Thompson sampling, Non-stationarity, Online learning.**

I. Introduction

Sequential decision making problems in reinforcement-learning (RL) are popularly formulated using the multi-armed bandit (MAB) framework, wherein, an agent (or player) selects one or multiple options (or arms) out of a set of arms at each time slot [1]–[5]. The player performs such an action-selection based on the current estimate or belief of the expected reward of the arms. Each time the player selects an arm or a group of arms, it receives a reward characterized by the reward distribution of the played arm/arms. The player updates its belief of the played arms based on the reward received. In case the reward distribution of the arms is stationary, several algorithms have been shown to perform optimally [6]. On the contrary, most real-world applications such as internet of things (IoT) networks [7], wireless communications [8], computational advertisement [9], and portfolio optimization [10] are better characterized by non-stationary rewards. However, non-stationarity in reward distributions are notoriously difficult to handle analytically.

To address this, researchers either i) construct passively-adaptive algorithms that are change-point agnostic and work by discounting the impact of past rewards gradually or ii) derive frameworks to actively detect the changes in the environment. Among the actively-adaptive algorithms, the state-of-the-art solutions, e.g., `ADSWITCH` by Auer *et al.* [11] provide a regret guarantee of $\mathcal{O}\left(\sqrt{KN_CT\log T}\right)$ for a K-armed bandit setting experiencing N_C changes in a time-horizon T. Recently, researchers have also explored predictive sampling frameworks for tackling non-stationarity [12].

In contrast to this we propose an algorithm based on grouped probing of the arms that identifies the arm that has undergone a change in its mean. We investigate the conditions under which the proposed algorithm achieves superior regret guarantees than `ADSWITCH`.

A. Related Work

Actively-adaptive algorithms have been experimentally shown to perform better than the passively-adaptive ones [13]. In particular, `ADAPT-EVE`, detects abrupt changes via the Page-Hinkley statistical test (PHT) [14]. However, their evaluation is empirical without any regret guarantees. Similarly, another work [10] employs the Kolmogorov-Smirnov statistical test to detect a change in distribution of the arms. Interestingly, tests such as the PHT has been applied in different contexts in bandit frameworks, e.g., to adapt the window length of `SW-UCL` [15]. The results by [16], [17], and by Cao *et al.* [18] detect a change in the empirical means of the rewards of the arms by assuming a constant number of changes within an interval. While the algorithm in [18], called `M-UCB` achieves a regret bound of $\mathcal{O}(\sqrt{KN_CT\log T})$, the work by Yu *et al.* leverages side-information to achieve a regret of $\mathcal{O}(K\log T)$. However, the proposed algorithms in both these works assume a prior knowledge of either the number of changes or the change frequency. On these lines, recently, Auer *et al.* [11] have proposed `ADSWITCH` based on the mean-estimation based checks for all the arms. Remarkably, the authors show regret guarantees of the order of $\mathcal{O}(\sqrt{KN_CT\log T})$ for `ADSWITCH` without any pre-condition on the number of changes N_C for the K-arm bandit problem. If the number of changes N_C is known, a safeguard against a

The author is with the Department of Electrical Engineering at IIT Delhi, India 110016. Email: gghatak@ee.iitd.ac.in.

change in an inferior arm a_i can be achieved by sampling it in inverse proportion of its sub-optimality gap Δ_i. This achieves a regret of $\mathcal{O}(\sqrt{KLT})$. However, if the number of changes is not known, setting the sampling rate is challenging and if the number of changes is greater than $\sqrt{T}$, several algorithms experience linear regret. In order to avoid this, the main idea in ADSWITCH is to draw consecutive samples from arms, thereby incurring a regret that scales as $\mathcal{O}\left(\sqrt{K}\right)$ in the worst-case. Nevertheless, since both M-UCB and ADSWITCH provide the same regret guarantees, we choose them as the competitor algorithms for our proposal.

B. Motivation and Contribution

Unlike M-UCB but similar to ADSWITCH, we consider a framework where the number of changes N_C is not known a-priori but to be fewer than $\sqrt{T}$. Furthermore, we target an additional requirement - the agent algorithm should guarantee that the age between two consecutive plays of each arm is bounded. Although the issue of age has previously been addressed in some works (e.g., see [19]), the solutions cater to stationary environments. On the contrary, in this work, under an assumption of the hard-core distance between two consecutive changes, we propose TS-GE which outperforms ADSWITCH and M-UCB under several regimes of K and the time-horizon, T, while simultaneously satisfying the mandatory probing requirement. The major innovation in this paper is two-fold - i) by allowing simultaneous probing of multiple arms in a coded manner, we reduce the scaling of changed-arm identification form $\mathcal{O}(K)$ to $\mathcal{O}(\log_2(K))$, and ii) by design, TS-GE guarantees that the last sample of each arm is not older than $\sqrt{T}$. Overall, the main contributions of this paper are:

- We develop and characterize TS-GE, tuned for non-stationary environments with unknown number of change-points. The additional design guarantee of TS-GE is the periodic mandatory probing of all the arms. Although this is relevant for several applications, to the best of our knowledge, this requirement has not been treated previously in literature. By balancing the regret guarantees of stationary Thompson sampling with grouped probing of all the arms, TS-GE ensures an upper bound of $\sqrt{T}$ in the sampling age of each arm.
- We propose a coded grouping of the arms based on the arm indices and consequently, derive the probability of missed detection of change and the probability of false alarm and highlight the conditions to limit these probabilities. Based on this, we show that TS-GE achieves sub-linear regret, $\mathcal{O}\left(K\log T + \sqrt{T}\left[\max\{N_C\left(1+\log K\right), T^{\frac{2}{5}}\}\right]\right)$. We compare this bound with the best known bound of $\mathcal{O}(\sqrt{KN_C T\log T})$ and discuss the conditions under which the bound of TS-GE outperforms the latter.
- Finally, as a case-study, we consider an industrial internet of things (IIoT) network where a central controller is required to sustain simultaneous wireless information and power transfer (SWIPT) services to the IoT devices. The different phases of TS-GE are mapped to the data-transfer and energy-transfer operations of the network. We demonstrate the performance of TS-GE with respect to the statistical upper-bound derived using stochastic geometry tools. Contrary to our proposal, in other algorithms such as ADSWITCH and M-UCB the exists non-zero probability that the IoT devices do not receive any energy transfer and hence our proposal will find applications that constrain sample age.

The rest of the paper is organized as follows. In Section II we describe the system model and the proposed TS-GE algorithm. The regret analysis is presented in Section III. We present a discussion on the regret bound in Section IV and demonstrate the efficacy of TS-GE in a case-study in Section V. Finally the paper concludes in Section VI.

II. TS-GE: Algorithm Description and Features

Consider a K-arm bandit setting with arms $\{a_i\} \in \mathcal{K}$, where $i = 1, 2, \ldots, K$. For the discussion in this paper, let us assume that $K = 2^d$. It may be noted that in case the number of arms is not a power of 2, the same can be transformed into one by adding *dummy* arms which are sub-optimal with a probability of 1 (e.g., arms with a constant reward of $-\infty$). The reward $R_{a_i}(t)$ of an arm a_i at time t is assumed to be a Gaussian distributed random variable with mean $\mu_i(t)$ and variance σ^2. Thus, the variance of all the arms is constant and is the same for all arms[1], however, the mean is a function of time.

Assumption 1. *We assume that at all times:* $R_{a_i}(t) \leq R_{max}, \quad \forall i = 1, 2, \ldots, K.$

In other words, the reward of all the arms is bounded above. Such an assumption is valid for most practical wireless applications, e.g., upper bound on received power or data-rate. Additionally, we assume a success or failure event associated to each $R_{a_i}(t)$ which are generated according to Bernoulli i.i.d. observations. Corresponding to each reward, the agent observes a success event with a probability:

$$R_\pi(t) = \frac{R_{a_i}(t)}{r_{max}}.$$

For example, in wireless communication applications, this corresponds to a success or failure in transmission associated with a data-rate of $R_{a_i}(t)$. The agent has access to both $R_{a_i}(t)$ and $R_\pi(t)$. Accordingly, the Thompson Sampling (TS) phase of our algorithm works with Beta priors with the sequence of $R_\pi(t)$. Furthermore, we assume that at any time-slot, multiple arms can be played by the agent. However, in that case, the agent observes a weighted average of the rewards of the pulled arms. This is formally mentioned below.

Assumption 2. *In case at any time-slot the agent pulls multiple (say n) arms, $\mathcal{S} = \{a_k\} \subset \mathcal{K}$, then the reward observed by the player is:*

$$R(t) = \frac{1}{n}\sum_{a_k \in \mathcal{S}} R_{a_k}$$

[1] This assumption is simply due to the ease of notations and the derivation of the regret bounds. The algorithm can be executed with this assumption does not hold. However, the regret analysis for such a case is much more involved and will be investigated in a future work.

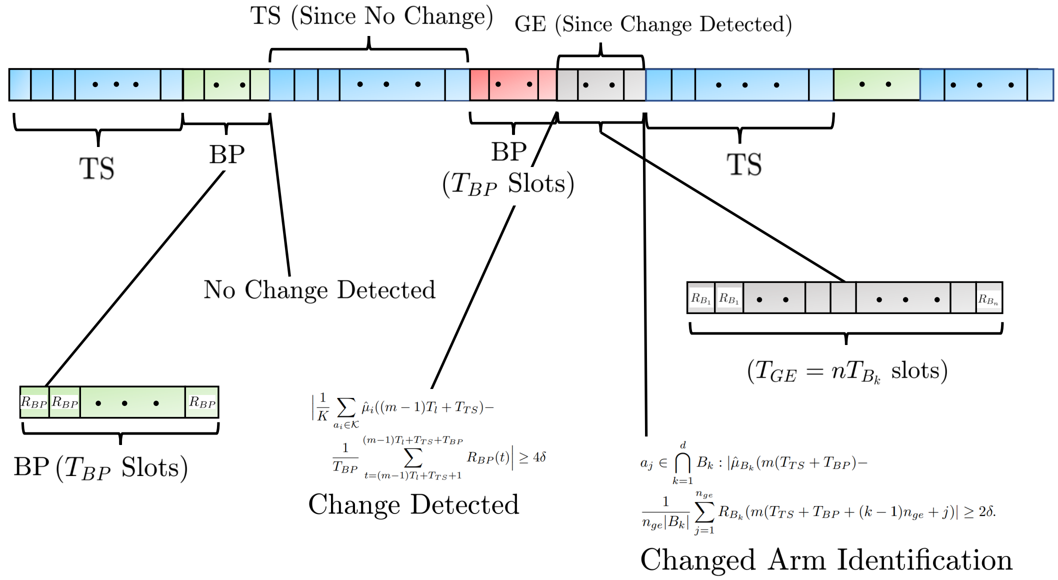


Fig. 1. Illustration of the different phases of TS−GE after the initialization with ETC for TS−GE.

Note that the player does not have access to the individual arm rewards of the set of arms it has played. At this stage, we note that such an assumption is not present in [11]. However, in several applications this is feasible - e.g., in wireless communications this may correspond to distributing the total transmit power among the available channels, in portfolio optimization this may correspond to distributing the total capital among different options.

A. Non-stationarity model

The player interacts with the bandit framework in a sequence of N_l *episodes*, denoted by E_i, $i = 1, 2, \ldots, N_l$, each of length T_l. Consequently, the total time-horizon T can be expressed as $T = N_l T_l$.

Condition 1. *The framework mandates that each arm be probed (either individually or in a group) at least once in each episode.*

We assume a piece-wise stationary environment in which changes in the reward distribution occur at time slots called *change points*, denoted by T_{C_j}, $j = 1, 2, \ldots$, where each $T_{C_j} \in [T]$. At each change point, exactly one of the arms a_i, uniformly selected from $\mathcal{K}$ experiences a change (increase or decrease) in its mean by an unknown amount Δ_i^C.

Assumption 3. *We assume Δ_C^i to be bounded as: $\Delta_i^C \geq 2\sigma$.*

Furthermore, we assume that during each episode, at most one change point occurs with a probability p_C and the total number of changes is N_C within T, which is unknown to the player. In particular, during each time slot of an episode, the environment samples a Bernoulli random variable C with success probability p_b. In case of a success, the change occurs in that slot, while in case of a failure, the bandit framework does not change. Once a change occurs in an episode, the change framework is paused until the next episode. Thus,

$$\mathbb{P}\left(\text{Episode } E_i \text{ experiences } c_i \text{ changes}\right) = \begin{cases} p_C; & c_i = 1, \\ 1 - p_C; & c_i = 0, \\ 0; & c_i > 1. \end{cases}$$

where, $p_C = \sum_{k=1}^{T_l} (1 - p_b)^{k-1} p_b$.

Assumption 4. *For a time horizon of T, we assume that the probability of change in each slot p_b is lower bounded as:*

$$p_b \geq 1 - \left(\frac{1}{T}\right)^{\frac{1}{\sqrt{T} - T^{\frac{2}{5}}}}. \tag{1}$$

Since the right hand side of the above is a decreasing function, for large values of T, the above is a fairly mild assumption.

The challenge for the player is to *quickly* identify changes that any arm has undergone and adapt its corresponding parameters. For benchmarking the performance of a candidate player policy, at a given time slot, a policy π competes against a policy class which selects the arm with the maximum expected reward at that time slot. Thus, any policy π that intends to balance between the exploration-exploitation trade-off of the bandit framework experiences a regret given by:

$$\mathcal{R}(T) = \sum_{t=1}^{T} \max_i \mu_i(t) - \mathbb{E}\left[\mu_\pi(t)\right], \tag{2}$$

where $\mu_\pi(t)$ is the mean of the arm $a_\pi(t)$ picked by the policy π at time t. It can be noted here that unlike stationary environments, the identity of the best arm a_j, where, $j = \arg\max_i \mu_i(t)$ is not fixed and may change with at each change point.

The key features of our proposed algorithm TS−GE are i) actively detecting the change in the bandit framework,

ii) identifying the arm which has undergone a change, and iii) modify its probability of getting selected in the further rounds based on the amount of change. The TS−GE algorithm consists of an initialization phase called explore-then-commit, ETC, followed by two alternating phases: classical TS phase followed by a broadcast probing (BP) phase to determine a change in the system. In case a change is detected in the BP phase, the arm which has undergone a change is identified using an optional sub-routine called group exploration (GE). Thus, the GE phase is only triggered if a change is detected in the BP phase. The overall algorithm is illustrated in Fig. 1 and presented in Algorithm 1. Each E_i consists of one TS phase, one BP phase, and (optionally) one GE phase. Next let us elaborate on the constituent phases of TE−GE.

B. Initialization: ETC for TS−GE

For initialization, the player performs a ETC for TS−GE, wherein each arm is played an n_{ETC} number of times and consequently, their mean μ_i is estimated to be $\hat{\mu}_i$.

Definition 1. *An arm a_i is defined to be well-localized if the empirical estimate $\hat{\mu}_i$ of its mean μ_i is bounded as:*

$$|\hat{\mu}_i(t) - \mu_i(t)| \leq \delta. \tag{3}$$

Lemma 1. *In the stationary regime, in order for the arm a_i to be well-localized with a probability $1 - p_L$, the arm needs to have been played at least n_{ETC} times, where:*

$$n_{ETC} = \frac{1}{2\delta^2} \ln \frac{1}{p_L}$$

Proof: The proof follows from Hoeffding's inequality. ■

Thus, the ETC phase lasts for at least $T_{ETC} = Kn_{ETC} = \frac{K}{2\delta^2} \ln \frac{1}{p_L}$ rounds. Naturally, in order to restrict p_L to $\mathcal{O}(\frac{1}{T})$, n_{ETC} needs to be $\mathcal{O}(\ln T)$.

C. Alternating TS and BP phases

Each episode consists of a TS phase, a BP phase, and an optional GE phase. Let us set $T_l = \sqrt{T}$. In the TS phase, the player performs the action selection of the choices according to the TS algorithm for T_{TS} slots as given in [20]. We set $T_{TS} = \sqrt{T} - T^{\frac{2}{5}}$. Each arm a_i is characterized by its TS parameters α_i and β_i, all of which are initially set to unity[2]. After each play of an arm a_i, its estimated mean $\hat{\mu}_i$ is updated. Let a_i be played at time slots $\{t_i\} \in [T]$ and the number of times it is played is $n_i(t)$ until (and including) the time-slot t, then:

$$\hat{\mu}_i(t) = \frac{1}{n_i(t)} \sum_{t \in t_i} R_{a_i}(t) \tag{4}$$

Additionally, the TS parameters for the played arm a_i, i.e., α_i and β_i are updated as per a Bernoulli trial with a success probability $R_\pi(t), \forall t \in \{t_i\}$ each time a_i is played. Here $R_\pi(t)$ is the normalized version of the reward obtained by playing the arm a_i as shown in step 8 of Algorithm 1.

[2] Note that the TS parameters can be initialized according to the observed rewards in the ETC phase. However, in order to safeguard against changes in the ETC phase, we begin the TS phase with fresh parameters.

Each TS phase is followed by the BP phase for $T_{BP} = T^{\frac{2}{5}}$ time-slots, where the player samples all the arms simultaneously for T_{BP} rounds. During this phase, the reward observed by the player is the average of the rewards from all the arms as mentioned in Assumption 2. The reward in the BP phase is then compared with the average of the estimates of all the arms to detect whether an arm of the framework has changed its mean. Recall that as per Assumption 2, during the BP phase of the m-th episode, the player receives the following reward for each play:

$$R_{BP}(t) = \frac{1}{K} \sum_{a_i \in \mathcal{K}} R_i(t) \sim \mathcal{N}\left(\frac{1}{K} \sum_{a_i \in \mathcal{K}} \mu_i(t), \frac{\sigma^2}{K}\right)$$
$$\forall (m-1)T_l + T_{TS} < t \leq (m-1)T_l + T_{TS} + T_{BP}$$

At the end of the m-th BP phase, a change is detected if:

$$\Bigg| \frac{1}{K} \sum_{a_i \in \mathcal{K}} \hat{\mu}_i((m-1)T_l + T_{TS}) - \frac{1}{T_{BP}} \sum_{t=(m-1)T_l+T_{TS}+1}^{(m-1)T_l+T_{TS}+T_{BP}} R_{BP}(t) \Bigg| \geq 4\delta \tag{5}$$

Here the first term is the average of the estimated means of all the arms at the end of the m-th TS phase, while the second term represents the same evaluated during the m-th BP phase. In case the change does not occur or goes undetected, the algorithm continues with the next TS phase. However, in case a change is detected or a false-alarm is generated, the algorithm moves on to the GE sub-routine as described below.

D. Policy after change detection

If a change is detected in the BP phase, the GE phase begins for the identification of the changed arm. The key step in this phase is the creation of d sets $B_k \subset \mathcal{K}, k = 1, 2, \ldots, d$, called *super arms* as shown in Algorithm 2. Recall that d is a number such that $d = \log_2(K)$. It may be noted that an optimal grouping of arms may be derived that considers the fact that the arms that have been played a fewer number of times have a larger error variance of its mean estimate. However, such a study is out of scope for the current text and will be treated in a future work. The i-th arm, where $i = 1, 2, \ldots, K$ is added to a super arm B_k if and only if the binary representation of i has a "1" in the k-th place. In other words, a_i is added to B_k if:

$$\texttt{bin2dec}(\texttt{dec2bin}(i) \ \texttt{AND} \ \texttt{onehot}(k)) \neq 0$$

where bin2dec() and dec2bin() are respectively operators that convert binary numbers to decimals and decimal numbers to binary. Additionally, onehot(k) is a binary number with all zeros except 1 at the k-th binary position. AND is the bit-wise AND operator. In the GE phase, each super arm is played n_{ge} times and the player obtains a reward which is the average of rewards of all the arms that belong to B_k as per Assumption 2, i.e., each time the super arm B_k is played, the player gets a reward that is sampled from the distribution:

$$R_{B_k} \sim \mathcal{N}\left(\frac{1}{|B_k|}\sum_{i\in B_k}\mu_i, \frac{\sigma^2}{|B_k|}\right)$$

Let the mean reward of the super arm B_k be denoted by μ_{B_k}. Before the beginning of each GE phase, μ_{B_k} is estimated using the individual mean estimates. As an example, let a change be detected after the m-th BP phase. Then, the estimate of the mean reward of the super arm B_k is:

$$\hat{\mu}_{B_k}(m(T_{TS}+T_{BP})) = \frac{1}{|B_k|}\sum_{a_i\in B_k}\hat{\mu}_i(m(T_{TS}+T_{BP})) \tag{6}$$

Then, the arm with the changed mean is the one that belongs to the all super arms in which a change of mean is detected. In other words, a change in arm a_j is detected if:

$$a_j \in \bigcap_{k=1}^{d} B_k : |\hat{\mu}_{B_k}(m(T_{TS}+T_{BP})- \\ \frac{1}{n_{ge}|B_k|}\sum_{j=1}^{n_{ge}} R_{B_k}(m(T_{TS}+T_{BP}+(k-1)n_{ge}+j)| \geq 2\delta. \tag{7}$$

Once the change is detected and the arm is identified, the corresponding mean of a_j is updated as:

$$\hat{\mu}_j = \sum_{k:a_j\in B_k}\hat{\mu}_{B_k} - \sum_{k:a_j\in B_k}\sum_{a_i\in B_k:i\neq j}\hat{\mu}_i \tag{8}$$

Then the TS parameters of the arm is updated. In particular, we set the parameters of a_j to be same as the arm that has an estimated mean closest ot a_j as:

$$\alpha_j = \alpha_k, \quad \beta_j = \beta_k$$
$$\text{where } k = \arg\min_{i\neq j}|\hat{\mu}_i - \hat{\mu}_j|$$

In the next section we characterize the probability with which TS-GE misses detection a change or raises a false alarm in case of no change. This eventually leads to the regret.

III. Analysis of TS-GE

Let us recall that the GE phase is triggered only if a change is detected in the BP phase. Consequently, the algorithm can miss detecting the chase in case the change occurs either in the TS phase or the BP phase, which we analyze below.

A. Probability of Missed Detection

Let the change occur in the arm a_i at t_c time slots within the m-th TS phase, i.e., $T_{ETC}+(m-1)(T_{TS}+T_{BP}) < t_c \leq T_{ETC}+mT_{TS}+(m-1)T_{BP}$. The mean is assumed to change from μ_i^- to μ_i^+. In other words, the distribution of the reward of a_i is given as:

$$R_i(t) \sim \\ \begin{cases} X_i^- \sim \mathcal{N}\left(\mu_i^-, \sigma^2\right); \; t \leq T_{ETC}+(m-1)(T_N+T_{BP})+t_c \\ X_i^+ \sim \mathcal{N}\left(\mu_i^+, \sigma^2\right); \; t > T_{ETC}+(m-1)(T_N+T_{BP})+t_c \end{cases}$$

Algorithm 1 TS-GE

Parameters:
Initialization: $\alpha_k = \beta_k = 1, \forall k = 1, \dots K$.
Thompson Sampling Phase:
for $e_i = E_1, \dots, E_{N_l}$ **do**
  for $t = 1, \dots, T_N$ **do**
    $\theta_i \sim$ Beta (α_i, β_i). \\ Sample the Beta prior.
    $a_j \leftarrow a_i|\theta_j = \max(\theta_i)$ \\ Select the best arm.
    $R_{\texttt{TS-GE}}(t) \leftarrow R_{a_j}(t)$ \\ Reward at time t.
    $R_\pi(t) \leftarrow \frac{R_{a_j}(t)}{R_{max}}$ \\ Normalize for Beta update.
    R^* = Bern $(R_\pi(t))$ \\ Bernoulli trial for Beta update.
    $\alpha_j \leftarrow \alpha_j + 1 - R^*$ \\ Update priors.
    $\beta_j \leftarrow \beta_j + R^*$ \\ Update priors.
    $n_j \leftarrow n_j + 1$ \\ Count of arm a_j.
    $\hat{\mu}_j(t) = \frac{\sum_{s=1}^{t} R_{a_j}(s)\mathcal{I}(a_j(s))}{n_j}$ \\ Update estimated mean of a_j.
  end for
  $p \leftarrow p + 1$. \\End of the p-th TS phase.
Broadcast Probing Phase:
  Play all the arms simultaneously for T_{BP} rounds and build the estimate:

$$\hat{\mu}_{BP} = \frac{1}{T_{BP}}\sum_{t=(e_i-1)T_l+T_{TS}+1}^{(e_i-1)T_l+T_{TS}+T_{BP}} R_{BP}(t)$$

  if Equation (5) holds **then**
    Change is detected.
Group Exploration Phase:
    Construct super-arms $\{B_k\} = \texttt{CSA}(\mathbf{a})$.
    for T_{GE} slots **do**
      Play B_k for T_{B_k} rounds.
      \\ Update μ_{B_k}:

$$\hat{\mu}_{B_k}(e_i(T_{TS}+T_{BP})) = \frac{1}{|B_k|}\sum_{a_i\in B_k}\hat{\mu}_i(e_i(T_{TS}+T_{BP}))$$

    end for
    $a_j \in \bigcap_{k=1}^{n} B_k : |\hat{\mu}_{B_k} - \frac{1}{n}\sum_{i\in B_k}\hat{\mu}_i(pT_N)| \geq \Delta, k = 1, 2, \dots, n.,$ \\ Identify changed arm.
    $\hat{\mu}_j(e_iT_l + 1) = \sum_{k:a_j\in B_k}\hat{\mu}_{B_k} - \sum_{k:a_j\in B_k}\sum_{a_i\in B_k:i\neq j}\hat{\mu}_i$ \\ Update the changed arm.
    Update the Beta parameters of the changed arm:

$$\alpha_j = \alpha_k, \quad \beta_j = \beta_k$$
$$\text{where } k = \arg\min_{i}|\hat{\mu}_i - \hat{\mu}_j|$$

  else
    Continue. \\ When no change is detected
  end if
end for

Algorithm 2 Construct Super-Arms `CSA`

Input: $\mathbf{a}$ and $\mathbf{n}$.
Initialize: $B_k = \{\}, \forall k = 1, 2, \ldots, K$.
for $k = 1$ to n **do**
  for $i = 1$ to K **do**
    if $\texttt{dec2bin}(i)$ AND $\texttt{onehot}(k) \neq \texttt{zeros}(1, n)$ **then**
      $B_k = B_k \cup a_i$
    end if
  end for
end for
Return B_k

The following lemma characterizes the probability of missed detection when the change occurs in the TS phase.

Lemma 2. *Let the arm a_i change its mean from μ_i^- to μ_i^+, where Δ_i^C at a time slot t_c in the m-th TS phase. Then the probability of missed detection after the m-th BP phase following this change is upper bounded by:*

$$\mathcal{P}_M^{TS} \leq \frac{1}{T} \tag{9}$$

Proof: See Appendix A. ∎

Lemma 3. *Let the arm a_i change its mean from μ_i^- to μ_i^+, where Δ_i^C at a time slot t_c in the m-th BP phase. Then the probability of missed detection after the m-th BP phase following this change has the following characteristic:*

$$\mathcal{P}_M^{BP} = \begin{cases} \mathcal{P}_{M,Case1}^{BP}; & \textit{with probability } > 1 - \frac{1}{T} \\ \mathcal{P}_{M,Case2}^{BP}; & \textit{with probability } \leq \frac{1}{T}, \end{cases} \tag{10}$$

where, $\mathcal{P}_{M,Case1}^{BP} \leq \frac{1}{T}$ and $\mathcal{P}_{M,Case2}^{BP} > 1 - \frac{1}{T}$.

Proof: See Appendix B. ∎

Thus, a change in the BP phase results in a different probability of missed detection based on the exact point of change.

B. Probability of False Alarm

The BP phase can raise a false alarm when a change has not occurred in an episode while, the condition (5) holds true simultaneously. However, in case of no change, the test statistic is simply:

$$Z_{NC} \sim \mathcal{N}\left(0, \sigma_{NC}\right) \tag{11}$$

where $\sigma_{NC}^2 = \frac{\sigma^2}{K}\left(\frac{1}{n_{ETC}} + \frac{1}{mT_{BP}} + \sum_{a_j \in \mathcal{K}} \frac{1}{n_j(m(T_{TS}+T_{BP}))}\right)$. Here $n_j(mT_l)$ is the number of times the arm a_j has been played in all the TS phases. Thus,

$$\mathcal{P}_{FA} = \mathbb{P}\left(|Z_{NC}| \geq 4\delta\right) \leq \mathcal{Q}\left(\frac{4\delta}{\sigma_{NC}}\right) \leq \frac{1}{T} \tag{12}$$

C. On the Regret of `TS-GE`

Now we have all the necessary results to derive the regret bound for `TS-GE`. Each episode either experiences a change or doesn't. Accordingly, the regret cna be dissected into the following components.

1) Regret in case of no change.: The number of such episodes is $N_l - N_C$. Each such episode experiences a mandatory regret bounded by:

$$\mathcal{R}^1_{\text{no change}}(T_l) \leq \underbrace{\mathcal{O}\left[\log\left(\sqrt{T} - T^{\frac{2}{5}}\right)\right]}_{A} + \underbrace{\mathcal{O}\left(T^{\frac{2}{5}}\right)}_{B},$$

where the term A is due to the TS phase and the term B is due to the BP phase. In case of a false alarm, the algorithm subsequently experiences worst-case regret in all the subsequent phases. This occurs with a probability of $\mathcal{P}_{FA}$, and hence its contribution to the overall regret is:

$$\mathcal{R}^2_{\text{no change}}(T_l) \leq \mathcal{P}_{FA}\Delta_{max}T \overset{(a)}{\leq} K_1$$

The step (a) follows from (12). Thus, overall, for the case of no change, the regret is:

$$\mathcal{R}_{\text{no change}}(T_l) \leq \mathcal{O}\left[\log\left(\sqrt{T} - T^{\frac{2}{5}}\right)\right] + \mathcal{O}\left(T^{\frac{2}{5}}\right) + K_1. \tag{13}$$

2) Regret in case of change.: The number of such episodes is N_c. Each such episode experiences a mandatory regret bounded by:

$$\mathcal{R}^1_{\text{change}} \leq \mathcal{O}\left(\Delta_{max}\sqrt{T}\right).$$

However, in case of missed detection, the algorithm subsequently experiences worst-case regret in all the subsequent phases. This occurs with a probability $\mathcal{P}_M = p_C^{TS}\mathcal{P}_M^{TS} + p_C^{BP}\mathcal{P}_M^{BP}$ and hence its contribution to the overall regret is:

$$\mathcal{R}^2_{\text{change}} \leq \mathcal{P}_M\Delta_{max}T \overset{(a)}{=} K_2$$

Thus, overall, for the case of no change, the regret is:

$$\mathcal{R}_{change} \leq \mathcal{O}\left(\Delta_{max}\sqrt{T}\right) + K_2 \tag{14}$$

Using the above development, we can bound the regret of `TS-GE` as follows:

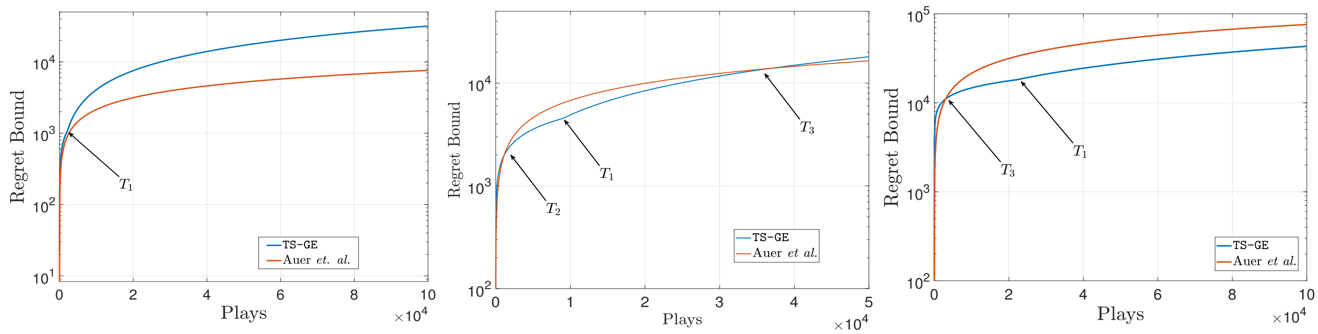
$$\begin{aligned} \mathcal{R}(T) &= \mathcal{R}_{ETC}\left(T_{ETC}\right) + \sum_{i=1}^{N_l}\mathcal{R}_i(T_l) \\ &\leq \mathcal{O}\left(K\log T\right) + (N_l - N_C)\mathcal{R}_{\text{no change}} + N_C\mathcal{R}_{\text{change}} \\ &\leq \mathcal{O}\left(K\log T + \sqrt{T}\left[\max\{N_C\left(1 + \log K\right), T^{\frac{2}{5}}\}\right]\right) \end{aligned} \tag{15}$$

Thus, not only the regret of `TS-GE` is sub-linear but also as discussed in the next section, it outperforms the known bounds under several regimes.

IV. Discussion

In this section, we discuss the derived regret bound of `TS-GE` with respect to the best known bound of `ADSWITCH` [11] and `M-UCB` [18], Let us define three time slots T_1, T_2. and T_3 as follows:

- $T_1 \doteq t : N_C(1 + \log_2 K) = t^{\frac{2}{5}}$.
- $T_2, T_3 \doteq t : \mathcal{R}_{\texttt{TS-GE}}(t) = \sqrt{N_C K t \log t}$, $T_2 \leq T_3$.



(a) $K = 100$.

(b) $K = 500$.

(c) $K = 1000$.

Fig. 2. Comparison of the regret bound of `TS-GE` with the best known regret bounds of `ADSWITCH` and `M-UCB` for different values of K

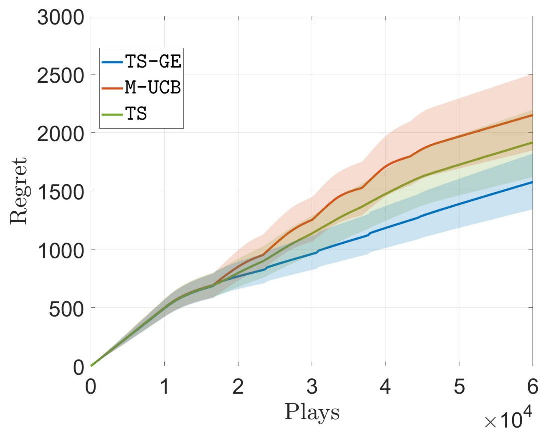


Fig. 3. Comparison of the regret of `TS-GE` with `M-UCB` and `TS`.

In other words, T_2 and T_3 are the time instants where the regret bound of `TS-GE` matches the bound of `ADSWITCH` or `M-UCB`. We compare the regret bounds for three different number of arms relevant for a massive IoT setup: small - $K = 100$ arms, medium - $K = 500$ arms, and large - $K = 1000$ arms. We consider a time-horizon of $T = 1e5$ number of plays.

For $K = 500$, Fig. 2b shows that there are specific regions where `TS-GE` outperforms `M-UCB`. Beyond T_2, `M-UCB` outperforms `TS-GE`. The point of interest for our discussion is the exact location of T_2 for different values of K. For $K = 100$, the value of T_2 is low (see Fig. 2a) and the regret bound of `M-UCB` is lower than `TS-GE` for most part of the time-horizon. However, in case of $K = 1000$, the value of T_2 is beyond the time-horizon, and accordingly, beyond time step 5000, throughout the time frame of interest, `TS-GE` outperforms `M-UCB`. This highlights the fact that the time-period of interest in a specific application would dictate the choice of a particular algorithm.

In Fig. 3, we compare the classical TS algorithm with `TS-GE` and `M-UCB`. In order to highlight the change detection framework, we consider fixed change points across different realizations of the algorithms. Specifically, changes occur at $E_i = 30, 60, 90, 120, 150$. We see that `TS-GE` outperforms the others by detecting the exact arm that has undergone a change. Interestingly, `M-UCB` performs worse than TS, mainly due to the fact that `M-UCB` flushes all the past (potentially relevant) rewards and restarts the exploration procedure once a system-level change is detected.

V. Case Study: SWIPT in an IIoT Network

In this section, we employ the `TS-GE` algorithm to evaluate an IIoT network where a central controller transmits data-packets to the device with the best channel condition and simultaneously performs wireless power transfer to all the devices.

A. Network Model

Let us consider an IIoT network consisting of a central wireless access point (AP) and K IoT devices. The set of the devices is denoted by $\mathcal{K}$. Typically an industrial environment deals with a large K that represent multiple sensors and cyber-physical systems. The AP provisions two wireless services in the network: i) periodic wireless power transfer (wpt) to all the IoT devices and ii) a unicast broadband data transmission to one selected IoT device. [3]

1) Network Parameters: Let the scenario of interest be modeled as a two-dimensional disk $\mathcal{B}(0, R)$ of radius R centered around the origin similar to [21]. The transmit power of the AP is P_t. The location of the devices is assumed to be uniform in $\mathcal{B}(0, R)$. Each AP-device link may be blocked by roaming blockages in the environment. The probability that a link of length r is in line of sight (LOS) is assumed to be $p_L(r) = exp(-\omega r)$ [22]. Furthermore, note that due to the presence of a large number of metallic objects, an industrial scenario presents a dense scattering environment. Consequently, we assume that each transmission link experiences a fast-fading h modeled as a Rayleigh distributed random variable with variance 1. Thus, The received power at an IoT device at a distance of r from the AP is given by $P_r(r) = KP_t hr^{-\gamma}$ with a probability $p_L(r)$. Here K and γ respectively are the path-loss coefficient and the path-loss exponent. The total transmission bandwidth is assumed to be B which is orthogonally allotted to the users scheduled in one time-slot.

[3] The unicast service is relevant in cases when the AP intends to select the best storage-enabled IoT device to transfer large files for caching at the edge. This may then be accessed by the other IoT devices, e.g., using device-to-device link. However, in this paper, we do not delve deeper into such an analysis.

At each episode, the controller selects the device with the best channel conditions and executes information transfer in a sequence of time-slots using the TS phase of the algorithm. The device-specific transmission can be facilitated by employing techniques such as beamforming. However we do not consider the details of such procedures. The TS phase information transfer is followed by joint power transfer to all the devices. This is mapped to the BP phase of the algorithm. At the end of the BP phase, the total energy harvested at all the devices at the end of the BP phase is reported back to the controller. Using this total energy transfer report, the controller detects whether a change in the large-scale channel conditions has taken place. If so, then the controller probes multiple devices grouped together as per `TS-GE` to detect the device with the current best channel conditions.

B. Performance Bounds using Stochastic Geometry

Before proceeding with the evaluation of `TS-GE` in this network, let us first derive the upper bound on the statistical performance of data-rate. This will enable a comparison with not only an existing algorithm but also the performance limit. Since the location of the devices is assumed to be uniform across the factory floor, they form a realization of a binomial point process (BPP). Additionally, due to the assumption that the blockage in each link is independent of each other, the IoT devices are either in LOS or NLOS state. The probability that at least one of the IoT devices is in LOS state is given by:

$$\begin{aligned} B_L &= \left[\int_0^R \exp(-\omega t)\frac{2t}{R}dt\right]^K \\ &= \left(2\frac{1-\exp\left(-\omega R^2\right)\left(\omega R^2+1\right)}{\omega^2 R^2}\right)^K \end{aligned}$$

The above expression follows similarly to [23]. On the same lines, the probability that at least one of the IoT devices is in NLOS state is given by

$$B_N = \left(R - 2\frac{1-\exp\left(-\omega R^2\right)\left(\omega R^2+1\right)}{\omega^2 R^2}\right)^K$$

1) Best-link transmission for information transfer: Out of the possible IoT devices, the central controller selects the device with the best channel condition for information transfer. For that first, let us derive the distance distributions of the nearest LOS and NLOS devices.

Lemma 4. *The distribution of the distance to the nearest LOS device, r_{L1} and the nearest NLOS device, r_{N1} are respectively:*

$$\begin{aligned} \mathbb{P}\left(r_{L1} \geq x\right) &= \frac{\left(x^2 U_L(x)\frac{R^2}{R^2-x^2}\right)^{K+1} - 1}{x^2 U_L(x)\frac{R^2}{R^2-x^2} - 1}\left(\frac{R^2-x^2}{R^2}\right)^K \\ \mathbb{P}\left(r_{N1} \geq x\right) &= \frac{\left(x^2 U_N(x)\frac{R^2}{R^2-x^2}\right)^{K+1} - 1}{x^2 U_N(x)\frac{R^2}{R^2-x^2} - 1}\left(\frac{R^2-x^2}{R^2}\right)^K \end{aligned} \tag{16}$$

where,

$$\begin{aligned} U_L(x) &= \frac{2\left(1-\exp\left(-\omega x\left(\omega x+1\right)\right)\right)}{\omega^2 x} \\ U_N(x) &= x - \frac{2\left(1-\exp\left(-\omega x\left(\omega x+1\right)\right)\right)}{\omega^2 x} \end{aligned}$$

Proof: Using the void probabilities (see [24]) we have:

$$\begin{aligned} &\mathbb{P}\left(r_{L1} \geq x\right) \\ &= \sum_{k=0}^{K}\left[\int_0^x \exp\left(-\omega t\right)\frac{2t}{x^2}dt\right]^k\left(\frac{x^2}{R^2}\right)^k\left(\frac{R^2-x^2}{R^2}\right)^{K-k} \\ &= \sum_{k=0}^{K}\left[x^2 U_L(x)\left(\frac{R^2}{R^2-x^2}\right)\right]^k\left(\frac{R^2-x^2}{R^2}\right)^K \end{aligned}$$

Evaluating the above series derives the result. The case for the NLOS device also follows in a similar manner. ■

Corollary 1. *Thus, the probability that the best device is in LOS state is given as:*

$$\begin{aligned} \mathcal{P}_L &= \mathbb{P}\left(KPr_{L1}^{-\gamma_L} \geq KPr_{N1}^{-\gamma_N}\right) \\ &= \mathbb{P}\left(r_{L1} \leq r_{N1}^{\frac{\gamma_N}{\gamma_L}}\right) \\ &= \mathbb{E}_{r_{N1}}\left[1 - \frac{\left(r_{N1}^{\frac{2\gamma_N}{\gamma_L}} U_L(r_{N1}^{\frac{\gamma_N}{\gamma_L}})\frac{R^2}{R^2-r_{N1}^{\frac{2\gamma_N}{\gamma_L}}}\right)^{K+1} - 1}{r_{N1}^{\frac{2\gamma_N}{\gamma_L}} U_L(r_{N1}^{\frac{\gamma_N}{\gamma_L}})\frac{R^2}{R^2-r_{N1}^{\frac{2\gamma_N}{\gamma_L}}} - 1}\right. \\ &\qquad \left.\left(\frac{R^2-r_{N1}^{\frac{2\gamma_N}{\gamma_L}}}{R^2}\right)^K\right]. \end{aligned}$$

Across different realizations of the network, the best link experiences a fading-averaged downlink received power

$$P_r = \begin{cases} KP_t r_{L1}^{-\gamma_L}; & \text{with probability } \mathcal{P}_L \\ KP_t r_{N1}^{-\gamma_N}; & \text{with probability } 1-\mathcal{P}_L \end{cases} \tag{17}$$

Over a time-horizon of T slots the throughput experienced by the system can then be evaluated as:

$$\mathcal{T} = \frac{N_l T_{TS}}{N_l T_{BP} + T_{ETC} + N_C T_{GE}}\mathbb{E}\left[B\log_2\left(1+\frac{P_r}{N_0}\right)\right] \tag{18}$$

where the expectation taken over P_r as per (17).

2) Multicast/Broadcast transmission for power transfer: Let us assume that in the multicast transmission phase, the AP transmits data to a subset $\mathcal{J} \subset \mathcal{K}$ of the IoT devices, where $|\mathcal{J}| = N_J$. In this case, the available bandwidth B is shared among the N_J devices. The harvested power experienced by an IoT device of $\mathcal{J}$ is:

$$\mathcal{T}_j = \begin{cases} \theta_e \frac{N_j}{K} KPr_j^{-\gamma_L} & \text{with probability } \exp\left(-\omega r_j\right) \\ \theta_e \frac{N_j}{K}\frac{BN_0}{N_j} & \text{with probability } 1-\exp\left(-\omega r_j\right) \end{cases} \tag{19}$$

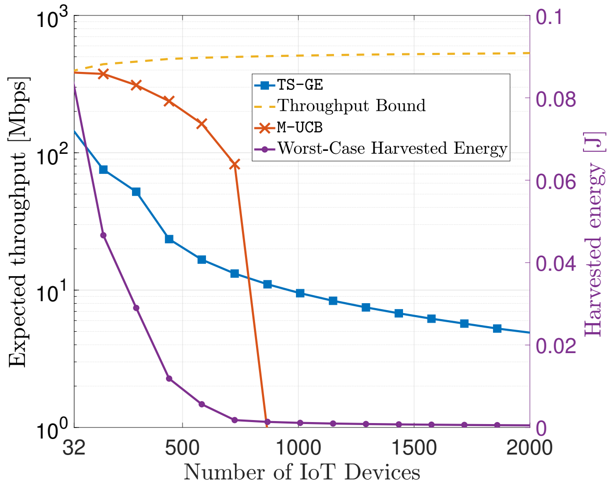


Fig. 4. Expected bes-device throughput and worst-case harversted energy with `TS-GE`.

Accordingly, the network sum-energy is given by:

$$\mathcal{T}_T = \sum_{j \in \mathcal{J}} T_j, \tag{20}$$

in one slot. In case of the BP phase, naturally we have $\mathcal{J} = \mathcal{K}$.

C. Numerical Example

We run the `TS-GE` algorithm in our IIoT network for a total of 1000s with time-slots of 10 ms [25]. Additionally we assume slow moving blockages in which each 30 seconds the visibility state of exactly one IoT device changes from LOS to NLOS or vice-versa. In Fig. 4 we plot the average throughput of the information transfer phase (i.e., to the best device) as well as the minimum harvested energy in a device in the IIoT network. We observe that in case of a fewer IoT devices, the `M-UCB` algorithm performs better than `TS-GE`. Indeed resetting all arms does not incur a large exploration loss in `M-UCB` in case K is small. Additionally, `M-UCB` is not constrained by mandatory exploration. Accordingly, it enjoys a higher throughput as compared to `TS-GE` which needs to transfer energy all the devices.

Interestingly, as the number of devices in the network increases beyond a threshold, `TS-GE` outperforms `M-UCB` especially due to rapid changed state identification in the device. Naturally, as K increases, the amount of time dedicated for energy transfer decreases. This is reflected in the reduced energy harvested in the worst device.

Several open problems are apparent. For example, the condition on that each episode can experience only one change can be too stringent to be applied meaningfully in contexts. Furthermore, change detection in the distributions rather than mean and extension to multiple players are indeed interesting directions of research which will be treated in a future work.

VI. Conclusion

Existing multi-armed bandit algorithms tuned for non-stationary environments sample sub-optimal arms in a probabilistic manner in proportion to their sub-optimality gap. However, several applications require periodic mandatory probing of all the arms. To address this, we develop a novel algorithm called `TS-GE` which balances the regret guarantees of classical Thompson sampling with a periodic group-exploration phase which not only ensures the mandatory probing of all the arms but also acts as a mechanism to detect changes in the framework. We show that the regret guarantees provided by `TS-GE` outperforms the state-of-the-art algorithms like `M-UCB` and `ADSWITCH` for several time-horizons, especially for a high number of arms. We demonstrated the efficacy of `TS-GE` in an industrial IoT network designed for simultaneous wireless information and power transfer.

References

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

[2] W. R. Thompson, "On the likelihood that one unknown probability exceeds another in view of the evidence of two samples," *Biometrika*, vol. 25, no. 3/4, pp. 285–294, 1933.

[3] J. Gittins, K. Glazebrook, and R. Weber, *Multi-armed bandit allocation indices*. John Wiley & Sons, 2011.

[4] S. Pilarski, S. Pilarski, and D. Varró, "Delayed reward bernoulli bandits: Optimal policy and predictive meta-algorithm pardi," *IEEE Transactions on Artificial Intelligence*, vol. 3, no. 2, pp. 152–163, 2022.

[5] S. Pilarski, S. Pilarski, and D. Varró, "Optimal policy for bernoulli bandits: Computation and algorithm gauge," *IEEE Transactions on Artificial Intelligence*, vol. 2, no. 1, pp. 2–17, 2021.

[6] E. Contal, D. Buffoni, A. Robicquet, and N. Vayatis, "Parallel gaussian process optimization with upper confidence bound and pure exploration," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 225–240, Springer, 2013.

[7] A. Uprety and D. B. Rawat, "Reinforcement learning for iot security: A comprehensive survey," *IEEE Internet of Things Journal*, vol. 8, no. 11, pp. 8693–8706, 2020.

[8] S. K. Singh, V. S. Borkar, and G. Kasbekar, "User association in dense mmwave networks as restless bandits," *IEEE Transactions on Vehicular Technology*, pp. 1–1, 2022.

[9] J. Huh and E. C. Malthouse, "Advancing computational advertising: Conceptualization of the field and future directions," *Journal of Advertising*, vol. 49, no. 4, pp. 367–376, 2020.

[10] G. Ghatak, H. Mohanty, and A. U. Rahman, "Kolmogorovsmirnov test-based actively-adaptive thompson sampling for non-stationary bandits," *IEEE Transactions on Artificial Intelligence*, 2021.

[11] P. Auer, P. Gajane, and R. Ortner, "Adaptively tracking the best bandit arm with an unknown number of distribution changes," in *Conference on Learning Theory*, pp. 138–158, 2019.

[12] Y. Liu, B. Van Roy, and K. Xu, "Nonstationary bandit learning via predictive sampling," *arXiv preprint arXiv:2205.01970*, 2022.

[13] J. Mellor and J. Shapiro, "Thompson sampling in switching environments with bayesian online change detection," in *Artificial Intelligence and Statistics*, pp. 442–450, 2013.

[14] C. Hartland, S. Gelly, N. Baskiotis, O. Teytaud, and M. Sebag, "Multi-armed bandit, dynamic environments and meta-bandits," 2006.

[15] V. Srivastava, P. Reverdy, and N. E. Leonard, "Surveillance in an abruptly changing world via multiarmed bandits," in *53rd IEEE Conference on Decision and Control*, pp. 692–697, IEEE, 2014.

[16] J. Y. Yu and S. Mannor, "Piecewise-stationary bandit problems with side observations," in *Proceedings of the 26th annual international conference on machine learning*, pp. 1177–1184, 2009.

[17] G. Ghatak, "A change-detection-based thompson sampling framework for non-stationary bandits," *IEEE Transactions on Computers*, vol. 70, no. 10, pp. 1670–1676, 2020.

[18] Y. Cao, Z. Wen, B. Kveton, and Y. Xie, "Nearly optimal adaptive procedure with change detection for piecewise-stationary bandit," in *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 418–427, 2019.

[19] S. Fatale, K. Bhandari, U. Narula, S. Moharir, and M. K. Hanawal, "Regret of age-of-information bandits," *IEEE Transactions on Communications*, vol. 70, no. 1, pp. 87–100, 2022.

[20] S. Agrawal and N. Goyal, "Analysis of thompson sampling for the multi-armed bandit problem," in *Conference on learning theory*, pp. 39–1, 2012.

[21] G. Ghatak, S. R. Khosravirad, and A. De Domenico, "Stochastic geometry framework for ultra-reliable cooperative communications with random blockages," *IEEE Internet of Things Journal*, 2021.

[22] T. Bai, R. Vaze, and R. W. Heath, "Analysis of Blockage Effects on Urban Cellular Networks," *IEEE Trans. Wireless Commun.*, vol. 13, no. 9, pp. 5070–5083, 2014.

[23] T. Bai and R. W. Heath, "Coverage and rate analysis for millimeter-wave cellular networks," *IEEE Transactions on Wireless Communications*, vol. 14, no. 2, pp. 1100–1114, 2014.

[24] D. Stoyan, W. S. Kendall, S. N. Chiu, and J. Mecke, *Stochastic geometry and its applications*. John Wiley & Sons, 2013.

[25] J.-C. Lin, "Synchronization requirements for 5g: An overview of standards and specifications for cellular networks," *IEEE Vehicular Technology Magazine*, vol. 13, no. 3, pp. 91–99, 2018.

Appendix A
Proof of Lemma 2

Let the number of times the arm a_i is played is t_i^- times before t_c and t_i^+ times after t_c. The test statistic (i.e., the parameter to be compared to 2δ) is simply a random variable Z_{TS} given by:

$$Z_{TS} = \frac{1}{Kn_{ETC}} \sum_{q=(i-1)n_{ETC}+1}^{in_{ETC}} X_i^-(q) + \frac{1}{K} \sum_{a_j \neq a_i} \frac{1}{n_{ETC}} \sum_{q=(j-1)n_{ETC}+1}^{jn_{ETC}} X_j(q) + \frac{1}{(K)n_j(T')} \sum_{a_j \neq a_i} \sum_{q_j=T_{ETC}+1}^{T'} \cdot X_j(q)\mathbb{I}(a_{\texttt{TS-GE}}(q) = a_j) + \frac{1}{t_i^- + t_i^+} \left[\sum_q X_i^-(q) + \sum X_i^+(q) \right] - \frac{1}{KT_{BP}} \sum_{q=T'}^{T''} X_i^+(q) - \frac{1}{KT_{BP}} \sum_{q=T'}^{T''} \sum_{a_j \neq a_i} X_j(q) \tag{21}$$

Here $T' = T_{ETC} + (m-1)(T_{TS} + T_{BP}) + T_{TS}$, $T'' = T_{ETC} + m(T_{TS} + T_{BP})$, and $n_j(t)$ is the number to times the arm a_j has been played until time t. Since all the arms except a_i remain stationary, we have:

$$\mathbb{P}\left(\Big| \frac{1}{K} \sum_{a_j \neq a_i} \frac{1}{n_{ETC}} \sum_{q=(j-1)n_{ETC}+1}^{jn_{ETC}} X_j(q) + \frac{1}{(K)n_j(T')} \sum_{a_j \neq a_i} \sum_{q_j=T_{ETC}+1}^{T'} X_j(q)\mathbb{I}(a_{\texttt{TS-GE}}(q) = a_j) + - \frac{1}{KT_{BP}} \sum_{q=T'}^{T''} \sum_{a_j \neq a_i} X_j(q) \Big| \geq 2\delta \right) \leq \mathcal{O}\left(\frac{1}{T^2}\right) \tag{22}$$

Consequently, for the decision of change detection it is of interest to consider the following random variable instead:

$$Z'_{TS} = \frac{1}{Kn_{ETC}} \sum_{q=(i-1)n_{ETC}+1}^{in_{ETC}} X_i^-(q) + \frac{1}{K(t_i^- + t_i^+)} \left[\sum_q X_i^-(q) + \sum X_i^+(q) \right] - \frac{1}{KT_{BP}} \sum_{q=T'}^{T''} X_i^+(q), \tag{23}$$

and compare it to a threshold of $2delta$. Note that Z'_{TS} is Gaussian distributed with mean $\mu_{Z'_{TS}} = \Delta_i^C + \frac{1}{t_i^- + t_i^+}\left(t_i^- \mu_i^- + t_i^+ \mu_I^+\right)$ and variance given by $\sigma^2_{Z'_{TS}} = \frac{\sigma^2}{K^2}\left(\frac{1}{n_{ETC}} + \frac{1}{t_i^- + t_i^+} + \frac{1}{T_{BP}}\right)$. Consequently, there are two cases of interest:

Case 1 - $\Delta_i^C > 0$: This is the case where the mean of the arm a_i increases from μ_i^- to μ_i^+. Accordingly the missed detection probability can be written as:

$$\begin{aligned}
\mathcal{P}_M^{TS} &= \mathbb{P}\left(|Z'_{TS}| \leq 2\delta\right) \leq \mathcal{Q}\left[\frac{\mu_{Z'_{TS}} - 2\delta}{\sigma_{Z'_{TS}}}\right] \\
&\overset{(a)}{\leq} \mathcal{Q}\left[\frac{\Delta_i^C - 2\delta}{\frac{\sigma}{K}\sqrt{\frac{1}{n_{ETC}} + \frac{1}{t_i^- + t_i^+} + \frac{1}{KT_{BP}}}}\right] \\
&\overset{(b)}{\leq} \mathcal{Q}\left[\frac{K\sqrt{n_{ETC} + t_i^- + t_i^+ + T_{BP}}}{3} \frac{\Delta_i^C - 2\delta}{\sigma}\right] \overset{(c)}{\leq} \frac{1}{T}.
\end{aligned}$$

In the above $\mathcal{Q}(\cdot)$ is the Gaussian-Q function. The inequality (a) follows from the facts that $\frac{1}{t_i^- + t_i^+}\left(t_i^- \mu_i^- + t_i^+ \mu_I^+\right) \geq 0$ and the $\mathcal{Q}(\cdot)$ is a decreasing function. The step (b) follows from the AM $>$ HM inequality, while the step (c) follows from the Assumption 3 and the inequality $Q(K.\sqrt{x^(2/5)}) \leq \frac{1}{x}$ for $K \geq 1$.

Case 2 - $\Delta_i^C \leq 0$: This refers to the case where the mean of arm a_i decreases from μ_i^- to μ_i^+, i.e., $\mu_i^+ \leq \mu_i^-$. Accordingly, the missed detection probability follows similarly to the above:

$$\mathcal{P}_M^{TS} = \mathbb{P}\left(|Z'_{TS}| \leq 2\delta\right) \leq \frac{1}{T}$$

Appendix B
Proof of Lemma 3

Let in the BP phase, the number of times all the arms are played simultaneously be t_i^- times before t_c and t_i^+ times after t_c. Given that other arms a_j where $j \neq i$ have not changed, the test statistic (i.e., the parameter to be compared to 2δ) is simply a random variable Z_{BP} similar to Z_{TS}. Since all the arms except a_i remain stationary, we have:

$$\mathbb{P}\left(\Big| \frac{1}{K} \sum_{a_j \neq a_i} \frac{1}{n_{ETC}} \sum_{q=(j-1)n_{ETC}+1}^{jn_{ETC}} X_j(q) + \frac{1}{Kn_j(T')} \sum_{a_j \neq a_i} \sum_{q_j=T_{ETC}+1}^{T'} X_j(q)\mathbb{I}(a_{\texttt{TS-GE}}(q) = a_j) + - \frac{1}{KT_{BP}} \sum_{q=T'}^{T''} \sum_{a_j \neq a_i} X_j(q) \Big| \geq 2\delta \right) \leq \mathcal{O}\left(\frac{1}{T^2}\right) \tag{24}$$

Consequently, for the decision of change detection it is of interest to consider the following random variable instead:

$$Z'_{BP} = \frac{1}{Kn_{ETC}} \sum_{q=(i-1)n_{ETC}+1}^{in_{ETC}} X_i^-(q) + \frac{1}{Kn_i(T')} \sum_q X_i^-(q) - \frac{1}{K(t_i^- + t_i^+)} \left[\sum_q X_i^-(q) + \sum X_i^+(q) \right] \tag{25}$$

Note that Z'_{BP} is Gaussian distributed with mean $\mu_{Z'_{BP}} = \mu_i^- - \frac{1}{t_i^- + t_i^+}\left(t_i^- \mu_i^- + t_i^+ \mu_i^+\right)$ and variance given by $\sigma^2_{Z'_{TS}} = \frac{\sigma^2}{K^2}\left(\frac{1}{n_{ETC}} + \frac{1}{t_i^- + t_i^+} + \frac{1}{T_{BP}}\right)$.

Case 1 - $\Delta_i^C > 4\delta$ and $t_i^- \leq \frac{T_{BP}(\Delta_i^C - 4\delta)}{\Delta_i^C}$:

This case occurs with a *high* probability. Due to the fact that for this case, we have $t_i^- \leq \frac{T_{BP}(\Delta_i^C - 4\delta)}{\Delta_i^C}$, i.e., $\mu_{Z'_{BP}} \geq 4\delta$, thus, similar to the Lemma 2,

$$\mathcal{P}^{BP}_{M|\text{Case 1}} \leq \frac{1}{T} \tag{26}$$

Thus, we have:

$$\mathcal{P}^{BP}_{M,\text{Case 1}} = \mathcal{P}^{BP}_{M|\text{Case 1}} \cdot \mathcal{P}_{\text{Case 1}} \leq \frac{1}{T}, \tag{27}$$

Case 2 - $\Delta_i^C > 4\delta$ and $t_i^- > \frac{T_{BP}(\Delta_i^C - 4\delta)}{\Delta_i^C}$: Here we have $\mu_{Z'_{BP}} < 4\delta$, and accordingly, the probability of missed detection is high. However, let us first observe the probability that the change occurs such that $t_i^- > \frac{T_{BP}(\Delta_i^C - 4\delta)}{\Delta_i^C}$. We have:

$$\begin{aligned}\mathbb{P}\left(\text{Case 2}\right) &= \mathbb{P}\left(t_i^- > \frac{T_{BP}(\Delta_i^C - 4\delta)}{\Delta_i^C}\right)\\ &= \mathbb{P}\left(\frac{t_i^+}{t_i^-} \geq \frac{4\delta}{\Delta_i^C - 4\delta}\right) \overset{(a)}{\leq} \frac{1}{T}\end{aligned}$$

where the step (a) is due to Assumption 4.

Case 3 - $\Delta_i^C < 4\delta$ and $t_i^- \leq \frac{T_{BP}(-\Delta_i^C - 4\delta)}{\Delta_i^C}$: This case is similar to Case 1 and hence we skip the detailed proof for brevity. In summary, similar to Case 1, for Case 3 the probability that the change occurs at a time step such that $t_i^- \leq \frac{T_{BP}(-\Delta_i^C - 4\delta)}{\Delta_i^C}$ holds is high. However, the probability of missed detection is bounded by $\frac{1}{T}$.

Case 4 - $\Delta_i^C < 4\delta$ and $t_i^- > \frac{T_{BP}(-\Delta_i^C - 4\delta)}{\Delta_i^C}$: This is similar to Case 2, wherein the probability of missed detection is high, while due to the Assumption 4, the occurrence of the change such that the condition $t_i^- > \frac{T_{BP}(-\Delta_i^C - 4\delta)}{\Delta_i^C}$ holds is bounded by $\frac{1}{T}$.