Variational Thompson Sampling for Relational Recurrent Bandits

Sylvain Lamprier[1], Thibault Gisselbrecht[1 2 3], and Patrick Gallinari[1]

[1] Sorbonne Universités, UPMC Univ Paris 06, CNRS, LIP6 UMR 7606, 4 place Jussieu, 75005 Paris, France
firstname.lastname@lip6.fr
[2] IRT SystemX, 8 Avenue de la Vauve, 91120 Palaiseau, France
[3] SNIPS, 18 Rue Saint-Marc, 75002 Paris

Abstract. In this paper, we introduce a novel non-stationary bandit setting, called relational recurrent bandit, where rewards of arms at successive time steps are interdependent. The aim is to discover temporal and structural dependencies between arms in order to maximize the cumulative collected reward. Two algorithms are proposed: the first one directly models temporal dependencies between arms, as the second one assumes the existence of hidden states of the system behind the observed rewards. For both approaches, we develop a Variational Thompson Sampling method, which approximates distributions via variational inference, and uses the estimated distributions to sample reward expectations at each iteration of the process. Experiments conducted on both synthetic and real data demonstrate the effectiveness of our approaches.

1 Introduction

A multi-armed bandit problem is a real time sequential decision process in which, at each iteration, a learner is asked to select an action - called arm - among a set of K available ones. The aim of the forecaster is to maximize the cumulative reward over iterations by balancing exploitation (arms with higher observed rewards should be selected often) and exploration (all arms should be explored to improve the knowledge of their utility). The stochastic multi-armed bandit, originally introduced in [24], has been widely studied in the literature. In this instance, the agent assumes stationary distributions of rewards. As an alternative to the optimistic UCB algorithm [7], Thompson Sampling (TS) [32] is another well-known algorithm, based on posterior distributions for reward expectations, to deal with this kind of problem. Another area of research concerns the problem of time varying distributions, in which the expected value of each arm is allowed to change from iteration to iteration. Different scenarios regarding non-stationarity have been studied in the literature: expected rewards may vary either abruptly as in [19], stochastically as in [30] or [27], or with a budgeted total variation of the expected reward as in [9].

In this paper we introduce a new setup, called relational recurrent bandit, where rewards of actions are interdependent over time. This corresponds to problems where rewards of arms have both temporal and structural dependencies that can be exploited to improve the efficiency of the reward collection. This is typically the case for tasks of sensors selection, where the aim is to collect useful data from streams under some budget constraints: for some reason (cost of sensor activation, restriction constraints, technical limits, etc.), data from every stream cannot be recorded simultaneously, the process must focus on the best current streams according to the task. In this context, dependencies can occur between streams, due to behavior correlations (similar reactions of sources to some external stimuli for instance, possibly with different delays), or to some kind of reward propagation (social influences between data sources for instance). To the best of our knowledge, such an instance of the multi-armed bandit problem has not been studied in the literature, although it corresponds to a realistic setting that can be met in several tasks, such as for instances dynamic sensor selection for climate modeling, useful source detection from social data streams, online information diffusion prediction and tracking, or even for advertising campaigns when the targeted communities have different reaction delays and/or influence relationships. This has to be distinguished from problems with structural dependencies between bandits, where reward distributions for connected situations are inter-dependent, for which there has been an increasing interest in the last few years [14, 17].

We investigate relational recurrent bandits for the multiple-plays scenario. While relational recurrent bandit could be defined for classical single-play process, considering temporal dependencies indeed becomes really attractive when more than one arm reveal their reward at each step. In this multiple-play problem, the agent selects $k > 1$ actions simultaneously, which leads to the collection of k rewards at each step. These observed rewards can be used to estimate reward expectations of every arm at the next time step, with the goal to select the most useful ones. Obviously, the temporal dependencies between arms' rewards are unknown a priori. They need to be learned online. The major difficulty in this problem is that the agent only knows the rewards of played arms, which leads to a problem of online learning with missing data.

Assuming linear dependencies between rewards of successive steps, we define a Bayesian model for the derivation of posterior distributions for unknown correlation parameters and the $K - k$ unobserved rewards at each time step. However, the exact computation of such posterior distributions is not analytically tractable. To overcome this difficulty, in order to build a TS procedure for relational recurrent bandits, a Variational Inference approach is developed to approximate true posterior distributions. In this context, two probabilistic models are considered: while a first one directly models temporal dependencies between arms, a second one is based on an underlying hidden Markov process that generates the rewards at each iteration. Although the first model better captures explicit relationships when some strong direct influences exist in the data, the second one allows one to greatly reduce the complexity and permits to encode more complex relationships.

To summarize, the contribution of this paper is three-fold:

- We propose a new instance of the bandit problem where distributions of rewards are defined recursively;
- We design a corresponding Thompson Sampling algorithm for two different formulations of the problem;
- We conduct experiments that assess the effectiveness of our approach on both synthetic and real data.

The paper is organized as follows. In section 2, we review related state-of-the-art literature. In section 3, we propose a first formulation of the relational recurrent bandit problem and an associated algorithm for this new bandit setting. Then, a second formulation for more complex dependencies is proposed in section 4. Finally, section 5 reports experimental results on both artificial and real data.

2 Related Works

Bandit problems have been extensively studied since the seminal paper of Lai & Robbins in 1958 [24]. For the case of stationary distributions of rewards, a huge variety of methods have been proposed to design efficient selection policies. The famous Upper Confidence Bound (UCB) algorithm proposed in [7] and many other UCB-based algorithms [5, 4, 20] have already proven to be efficient both empirically and theoretically. In these optimistic approaches, confidence intervals on the reward expectations are determined given uncertainty on the distribution estimates. The selected arm at each step is the one with the highest upper confidence bound, which allows one to guarantee logarithmic bounds of the regret[4]. On the other hand, the TS algorithm [32] introduces randomness on the exploration by sampling distribution parameters from their posterior at each time-step, before selecting the arm with the best reward expectation following the sampled distribution. It recently attracted increasing attention [15], due to its good performances and simplicity of use. It has also been proven to have a logarithmic regret bound in [2]. When more than one arm (say k) are selected at each iteration, the problem is called multiple-play multi-armed bandit, for which UCB-based [16] and TS policies [23] have been derived.

When K is very large, the problem becomes more challenging, since inducing sometimes intractable exploration spaces. Fortunately, in such scenarios, the rewards often exhibit some structural properties whose the agent can leverage. For instances, contextual versions of UCB are given in [25, 1], for the case where rewards depend on some available contextual knowledge, according to unknown parameters to be learned. Arms with similar contexts at a given time t are supposed to lead to similar rewards, which helps the exploration process. Different cases have been investigated in the literature, for example linear [18, 3, 1], unimodal [29] or Lipschitz [10]. In every case, the structure reflects some relation between arms rewards.

[4] Where the regret corresponds to the expectation of what we missed with a given policy compared with an optimal strategy that knows exact distribution parameters.

Closer to our work, still quite far, are approaches that focus on bandits on graphs, first introduced in [26] for the adversarial bandit and then in [12] for the stochastic bandit. Later, in [11], the UCB-LP was proposed for an equivalent setting and improved the results of [12]. Note that in those cases, the graph structure is supposed to be known beforehand and is explicitly used by the algorithm. Basically, the main assumption is that when an arm is played, not only its reward is revealed, but the reward of all its neighbors in the graph is also shown to the learner. Another area of research concerns contextual bandit on graphs, for which each node is a contextual bandit and the edges of the graph are used to define the similarity between the weights of the different contextual bandits [14, 17]. In [14] the edges of the graph are known, whereas in [17] the learner has to find the different clusters online. Our work differs from all of these approaches, by leveraging both temporal and structural dependencies between arms (and not between situations such as in [14, 17]). Observing some good actions at some time step can provide information on the utility of the others ones for the following time step. In our setting, rewards at step $t-1$ modify the distribution of all rewards at the next time step t. Capturing these dependencies may allow one to better handle non stationarity of rewards.

Non stationary bandits have been studied in some recent works, such as [19] which considers that the reward distribution can change during the process, but with a limited number of changes, and proposes two algorithms for this case: `Discounted UCB` which uses a discount factor to give more importance to recent observations than to older ones and `Sliding Window UCB` which uses a sliding window of size τ that restricts estimations to be performed only from observations in the last τ time steps. Rather than assuming radical distribution changes as in [19], *restless bandits* [33] consider that the state of each action evolves according to some random process[5]. Following this principle, [30] assume independent Brownian motions between consecutive steps, while [27], [6] or [31] consider that some hidden Markov process drives evolutions of reward expectations on each arm. These works differ from ours by the fact that reward expectations of different arms are independent. Moreover, considered states are usually discrete in existing works.

Finally, the recent work of [13] tackles the local influence maximization problem, in which authors do not assume any knowledge of the graph, but consider a setting where it can be gradually discovered. Indeed, the only information the learner has is a set of nodes each arm is currently influencing. Even if the temporal dependencies we aim at discovering in our approach can be seen as an influence graph, our proposal greatly differs from [13], which only focus on identifying the most influencial nodes of an unknown network without explicitly modeling the underlying dynamics of the successive rewards. Recurrent bandits stand as a novel instance of bandit problems, where temporal dependencies have to be extracted from incomplete data in an online fashion, in order to deal with the non-stationarity of the reward expectations.

[5] When only the state of the selected action changes at each iteration, the problem is called *rested bandit*.

3 Relational Recurrent Bandit

The bandit problem with multiple plays processes as follows: at each time step $t \in \{1, ..., T\}$, a subset $\mathcal{K}_t \subset \mathcal{K} = \{1, ..., K\}$ of k arms is selected and for every arm $i \in \mathcal{K}_t$ the agent receives the corresponding rewards $r_{i,t} \in \mathbb{R}$. The choice of $\mathcal{K}_t$ is done based on the historical decisions $\mathcal{H}_{t-1} = \{(i, r_{i,s}), i \in \mathcal{K}_s, s = 1..t-1\}$. The function that selects at each time-step t the subset $\mathcal{K}_t$ is called a bandit policy or algorithm.

In this section we propose to design a first TS algorithm for the relational recurrent bandit setting. The principle of TS is to produce a sample of the reward expectation according to its posterior distribution, and then to choose the arm with the best sampled expectation.

3.1 Probabilistic Model

In this section, we propose to consider direct recurrent relationships between arm's rewards from one time step to the following one. More specifically, we assume that the expected reward of an arm i at time step t is a linear combination of every rewards at time step $t-1$ plus a bias term, which we formalize as follows:

$$\forall i \in \{1, ..., K\}, \exists \theta_i \in \mathbb{R}^{K+1} \text{ such that } : \forall t \in \{2, ..., T\} : \mathbb{E}[r_{i,t}|R_{t-1}] = \theta_i^\top R_{t-1}^+ \tag{1}$$

where $R_t = (r_{1,t}, ..., r_{K,t})^\top \in \mathbb{R}^K$ is the vector of rewards at time step t and $R_t^+ = (R_t, r_{K+1,t})$ appends an additional constant bias term $r_{K+1,t}$ always equal to 1 at the end of R_t. We consider the following assumptions to derive our relational recurrent model:

- **Likelihood of data:** $\forall i \in \{1, ..., K\}, \exists \theta_i \in \mathbb{R}^{K+1}$ such that $\forall t \in \{2, ..., T\}$: $r_{i,t} = \theta_i^\top R_{t-1}^+ + \epsilon_{i,t}$, where $\epsilon_{i,t} \sim \mathcal{N}(0, \sigma^2)$ (Gaussian noise with mean 0 and variance σ^2).
- **Prior on parameters:** $\forall i \in \{1, ..., K\}$: $\theta_i \sim \mathcal{N}(0, \alpha^2 I)$ (($K+1$)-dimensional Gaussian vector with mean 0 and covariance matrix $\alpha^2 I$).
- **Prior at time 1:** $\forall i \in \{1, ..., K\}$: $r_{i,1} \sim \mathcal{N}(0, \sigma^2)$.

For clarity, we consider simple covariance matrices for the different priors, however any prior can be used, depending on the knowledge of the agent. In the following, we propose a probabilistic analysis of the above model, which is then used to derive our bandit policy.

3.2 Algorithm

To perform TS, at each time step $t \geq 2$, we thus need to be able to sample a value $\tilde{r}_{i,t}$ from the expected reward posterior distribution $\theta_i^\top R_{t-1}^+$ for each action i. If the rewards of unplayed arms were revealed at each time step, i.e if the full vector R_{t-1} was available, this would come down to a traditional contextual bandit problem. However, for every time step $s \in \{1, ..., t-1\}$ and $i \notin \mathcal{K}_s$, the reward $r_{i,s}$ is not available and must be treated as a random variable. Thus, TS must be performed from the following joint distribution:

$$P((r_{i,s})_{s=1..t-1, i \notin \mathcal{K}_s}, (\theta_i)_{i=1..K} | (r_{i,s})_{s=1..t-1, i \in \mathcal{K}_s}) \tag{2}$$

However, due to the recurrent aspect of the problem, this distribution cannot be directly obtained. To overcome this difficulty, we propose to adopt a Variational Inference approach, which stands as an alternative to MCMC (Monte Carlo Markov Chain) methods, such as Gibbs Sampling, to approximate complex distributions. While MCMC methods provide numerical approximations via successive samplings following the true joint distribution, variational inference outputs an analytical locally optimal approximation of this distribution. In practice the use of MCMC approaches for relational recurrent bandits would induce too important computation costs at each step, and worst, too many steps to converge toward tight distributions.

The idea of Variational Inference is to approximate the true targeted distribution by a simpler distribution. We propose here to consider the following mean field approximation:

$$Q((r_{i,s})_{s=1..t-1, i \notin \mathcal{K}_s}, (\theta_i)_{i=1..K}) = \prod_{i=1}^{K} q_{\theta_i}(\theta_i) \prod_{s=1}^{t-1} \prod_{i \notin \mathcal{K}_s} q_{r_{i,s}}(r_{i,s}) \tag{3}$$

with all q_{θ_i} and $q_{r_{i,s}}$ standing as independent variational distributions set for every factor of Q. The aim is then to find variational distributions for each factor, that minimize the Kullback-Leibler divergence (KL) from the true joint probability distribution P (as defined in (2)) to the variational joint distribution Q. This leads to an approximated distribution Q that is included in P: it can ignore some modes of P, but does not give posterior mass to regions where P has vanishing density. Following this, we obtain the variational distributions given in the two following distributions (all proofs are given in the supplementary material[6]).

Proposition 1 *Let $D_{t-1} = (R_s^{+\top})_{s=1..t-1}$ be the $(t-1) \times (K+1)$ matrix where row s corresponds to the rewards vector at time s concatenated with an additional component set to 1. We also note $D_{1..t-2}$ the $(t-2) \times (K+1)$ matrix of the $t-2$ first rows of D_{t-1} and $D_{i:2..t-1}$ the vector containing the $t-2$ last components of the i-th column of D_{t-1} (i.e., rewards obtained by i from 2 to $t-1$). Then, for all $t \geq 2$, and for $i \in \{1, ..., K\}$, the best variational distribution of $q^*_{\theta_i}$ corresponds to a Gaussian $\mathcal{N}(A_{i,t-1}^{-1} b_{i,t-1}, A_{i,t-1}^{-1})$, with:*

- $A_{i,1} = \frac{I}{\alpha^2}$ *and* $b_{i,1} = 0$
- $A_{i,t-1} = \frac{\mathbb{E}[D_{1..t-2}^\top D_{1..t-2}]}{\sigma^2} + \frac{I}{\alpha^2}$ *and* $b_{i,t-1} = \frac{\mathbb{E}[D_{1..t-2}^\top]}{\sigma^2} \mathbb{E}[D_{i:2..t-1}]$ *for* $t > 2$

where $\mathbb{E}[D_{1..t-2}^\top D_{1..t-2}] = \sum_{s=1}^{t-2} \mathbb{E}[R_s^+]\mathbb{E}[R_s^+]^\top + Var(R_s^+)$, with values of $\mathbb{E}[R_s^+]$ and $Var(R_s^+)$ determined according to proposition 2.

[6] Available at *http://www-connex.lip6.fr/~lampriers/ECML2017-supMat.pdf*

Proposition 2 *We note Θ the $K \times (K+1)$ matrix where row i equals to $\theta_i^\top$ and β_j the j-th column of Θ. For $t \geq 2$ and $1 \leq s \leq t-1$, the best variational distribution $q^*_{r_{i,s}}$ is a Gaussian $\mathcal{N}(\mu_{i,s}, \sigma^2_{i,s})$, with:*

- *If $s = 1$:* $\mu_{i,s} = \dfrac{\mathbb{E}[\beta_i]^\top \mathbb{E}[R_{s+1}] - \sum\limits_{j=1, j\neq i}^{K+1} \mathbb{E}[\beta_i^\top \beta_j]\mathbb{E}[r_{j,s}]}{1 + \mathbb{E}[\beta_i^\top \beta_i]}$, $\sigma^2_{i,s} = \dfrac{\sigma^2}{1 + \mathbb{E}[\beta_i^\top \beta_i]}$
- *If $s = t-1$:* $\mu_{i,s} = \mathbb{E}[\theta_i]^\top \mathbb{E}[R^+_{s-1}]$,$\sigma^2_{i,s} = \sigma^2$
- *Else:* $\mu_{i,s} = \dfrac{\mathbb{E}[\beta_i]^\top \mathbb{E}[R_{s+1}] - \sum\limits_{j=1, j\neq i}^{K+1} \mathbb{E}[\beta_i^\top \beta_j]\mathbb{E}[r_{j,s}] + \mathbb{E}[\theta_i]^\top \mathbb{E}[R^+_{s-1}]}{1 + \mathbb{E}[\beta_i^\top \beta_i]}$, $\sigma^2_{i,s} = \dfrac{\sigma^2}{1 + \mathbb{E}[\beta_i^\top \beta_i]}$

where $\mathbb{E}[\beta_i^\top \beta_j] = \sum\limits_{l=1}^{K} Var(\theta_l)_{i,j} + \mathbb{E}[\theta_l]_i \mathbb{E}[\theta_l]_j$.

Variational distributions given in propositions 1 and 2 are inter-dependent. Their estimation must therefore be performed via an iterative procedure, which is described in algorithm 1. This algorithm uses a parameter *nbIt* which corresponds to the number of iterations to achieve. Note that, when a reward is observed, one obviously uses the associated value rather than its expectation. That is, component i of $\mathbb{E}[R^+_s]$ (denoted $\mathbb{E}[R^+_s]_i$) equals $r_{i,s}$ in propositions 1 and 2 if $i \in \mathcal{K}_s$, $\mathbb{E}[r_{i,s}]$ otherwise. Moreover, $Var(R_s)$ corresponds to a diagonal matrix where element (i,i) equals 0 if $i \in \mathcal{K}_s$, $\sigma^2_{i,s}$ otherwise. Note at last that $r_{K+1,s}$ is always known: $\mathbb{E}[r_{K+1,s}] = 1$ and $Var(r_{K+1,s}) = 0$ for any iteration s.

Algorithm 1: `Variational Inference`

```
Input: $nbIt$
for $It = 1..nbIt$ do
    for $i = 1..K$ do
        Compute $A_{i,t-1}$, $b_{i,t-1}$ w.r.t. proposition 1;
        for $s = 1..t-1$ do
            if $i \notin \mathcal{K}_s$ then
                Compute $\mu_{i,s}$, $\sigma^2_{i,s}$ w.r.t. proposition 2 ;
        end
    end
end
```

Algorithm 2: `Recurrent Thompson Sampling`

```
for $t = 1..T$ do
    Perform Variational Inference;
    for $i = 1..K$ do
        Sample $\tilde{\theta}_i \sim q^*_{\theta_i}$;
        if $i \notin \mathcal{K}_{t-1}$ then
            Sample $\tilde{r}_{i,t-1} \sim q^*_{r_{i,t-1}}$;
        $\tilde{r}_{i,t} = \tilde{\theta}_i^\top \tilde{R}^+_{t-1}$;
    end
    $\mathcal{K}_t \leftarrow \underset{\hat{\mathcal{K}} \subseteq \mathcal{K}, |\hat{\mathcal{K}}| = k}{\arg\max} \sum_{i \in \hat{\mathcal{K}}} \tilde{r}_{i,t}$ ;
    for $i \in \mathcal{K}_t$ do Collect $r_{i,t}$ ;
end
```

Algorithm 2 describes our recurrent relational TS procedure, which uses algorithm 1 to estimate variational distributions for hidden variables at each iteration of the process. At each iteration t, the algorithm samples every hidden reward at time $t-1$ from $q^*_{r_{i,t-1}}$ and θ parameters from $q^*_{\theta_i}$ for every action i. It allows one to compute an expectation score $\tilde{r}_{i,t} = \tilde{\theta}_i^\top \tilde{R}^+_{t-1}$ for every action. The k actions with best $\tilde{r}_{i,t}$ scores are performed and the associated rewards are collected.

The complexity of the proposed algorithm increases linearly with the number of time steps from the beginning. At time t, we have $K(K+1)+(t-1)(K-k)$ random variables whose inter-dependent distributions have to be re-estimated at each iteration to include new observations (there are $(t-1)(K-k)$ missing rewards at time t). This is not compatible with the online nature of the bandit problem, since it might lead to memory overhead and complexity problems. To cope with this issue, we propose to use an approximated algorithm that restrains to a limited amount of time-steps in the past by introducing a sliding window of size S: instead of considering every missing reward from time 1 to time $t-1$ at each iteration t, the algorithm restricts distribution re-estimations to missing rewards from time $t-S-1$ to $t-1$, making its complexity constant with time. On the other hand both the $K(K+1)$ and the $(K-k)$ factors cannot be reduced easily with the proposed method. This comes from the fact that the algorithm tries to learn every weight of the model. When the number of arms becomes large, the model turns out to be very hard to learn. To cope with this issue, and also to deal with longer term dependencies, we propose a second model which considers transitions between hidden states of the system rather than explicit relationships.

4 State-Based Recurrent Bandit

In this section, we assume the existence of an underlying hidden state $h_t \in \mathbb{R}^d$ responsible for reward values at each iteration t. Moreover, the size d of this state is assumed to be smaller than K. On the other hand, unlike in the previous model, the recurrence is assumed to take place in the hidden state, such that there exists a linear transformation between h_{t-1} and h_t. Formally:

$$\exists \Theta \in \mathbb{R}^{d \times d}, \forall t \in \{2, ..., T\} : \mathbb{E}[h_t | h_{t-1}] = \Theta h_{t-1} \tag{4}$$

$$\forall i \in \{1, ..., K\}, \exists W_i \in \mathbb{R}^d, \exists b_i \in \mathbb{R}, \forall t \in \{1, ..., T\} : \mathbb{E}[r_{i,t} | h_t] = W_i^\top h_t + b_i \tag{5}$$

with $h_t \in \mathbb{R}^d$ the hidden state of the system at iteration t, Θ the transition matrix $d \times d$, $W_i \in \mathbb{R}^d$ the mapping vector from any state to the expected reward for i and b_i a bias term for every action i. The model is illustrated on 3 iterations in figure 1, where one observes temporal dependencies between states. The model is able to deal with long term dependencies thanks to recurrent relationships between continuous states.

We consider the following assumptions:

- **Likelihood 1:** $\exists \Theta \in \mathbb{R}^{d \times d}, \forall t \in \{2, ..., T\} : h_t = \Theta h_{t-1} + \epsilon_t$, where $\epsilon_t \sim \mathcal{N}(0, \delta^2 I)$.
- **Likelihood 2:** $\forall i \in \{1, ..., K\}, \exists W_i \in \mathbb{R}^d, \exists b_i \in \mathbb{R}$ such that : $\forall t \in \{1, ..., T\} : r_{i,t} = W_i^\top h_t + b_i \; \epsilon_{i,t} \sim \mathcal{N}(0, \sigma^2)$.
- **Prior on h_1:** $h_1 \sim \mathcal{N}(0, \delta^2 I)$, where I is the identity matrix of size d.

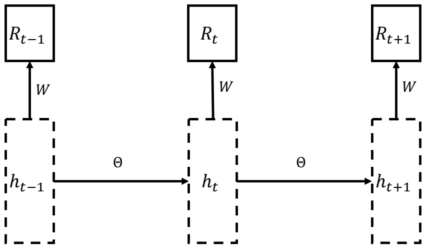

Fig. 1. Generative State-Based Recurrent Model

- **Prior on Θ:** $\forall i \in \{1, ..., d\}$: $\theta_i \sim \mathcal{N}(0, \alpha^2 I)$, where θ_i is the i-th row of Θ.
- **Prior on W:** $\forall i \in \{1, ..., K\}$: $(W_i, b_i) \sim \mathcal{N}(0, \gamma^2 I)$, with (W_i, b_i) the $d+1$ sized vector resulting from appending b_i to W_i, and I the identity matrix.

This defines a Linear Dynamic System for which parameters have to be estimated. With known transition Θ and emission $(W_i)_{i=1..K}$ matrices, this model would be similar to a Kalman Filter [22], usually designed to get the state of a physical system from noisy observations but with a known dynamical model. Our problem falls in Bayesian Linear Dynamical Systems, for which a variational approach has been proposed in [8]. However, the direct application of the very generic method proposed in this paper is complex. We now describe the resulting distributions for our specific case (proofs are given in the supplementary material).

Following the same approach as in the previous section we consider the following mean-field approximation:

$$Q(h_1, ..., h_{t-1}, \Theta, W, b) = \prod_{s=1}^{t-1} q_{h_s}(h_s) \prod_{i=1}^{K} q_{W_i, b_i}(W_i, b_i) \prod_{j=1}^{d} q_{\theta_j}(\theta_j) \tag{6}$$

Propositions 3, 4 and 5 give variational distributions based on this factorization.

Proposition 3 *Let W be a $K \times d$ matrix whose row i equals $W_i^\top$ and b a bias vector of size K. At step $t \geq 2$, for all s such that $1 \leq s \leq t-1$, the best variational distribution $q^*_{h_s}$ is a Gaussian $\mathcal{N}(F_s^{-1} g_s, F_s^{-1})$ with:*

$$\begin{cases} If \ 1 < s < t-1 : F_s = \mathcal{A}_s + \mathcal{B}_s + \mathcal{C}_s \ and \ g_s = \mathcal{D}_s + \mathcal{E}_s + \mathcal{F}_s \\ If \ s = 1 : F_s = \mathcal{A}_s + \mathcal{B}_s + \mathcal{C}_s \ and \ g_s = \mathcal{E}_s + \mathcal{F}_s \\ If \ s = t-1 : F_s = \mathcal{A}_s + \mathcal{C}_s \ and \ g_s = \mathcal{D}_s + \mathcal{E}_s \end{cases}$$

$$\mathcal{A}_s = \frac{I}{\delta^2}, \ \mathcal{B}_s = \frac{\mathbb{E}[\Theta^\top \Theta]}{\delta^2}, \ \mathcal{C}_s = \frac{\sum\limits_{i \in \mathcal{K}_s} \mathbb{E}[W_i W_i^\top]}{\sigma^2}, \ \mathcal{D}_s = \frac{\mathbb{E}[\Theta]\mathbb{E}[h_{s-1}]}{\delta^2},$$
$$\mathcal{E}_s = \frac{\sum\limits_{i \in \mathcal{K}_s} \mathbb{E}[W_i] r_{i,s} - \mathbb{E}[W_i b_i]}{\sigma^2}, \ \mathcal{F}_s = \frac{\mathbb{E}[\Theta]^\top \mathbb{E}[h_{s+1}]}{\delta^2};$$

$\mathbb{E}[\Theta^\top \Theta] = \sum_{i=1}^{d} \mathbb{E}[\theta_i \theta_i^\top] = \sum_{i=1}^{d} (\mathbb{E}[\theta_i]\mathbb{E}[\theta_i]^\top + Var(\theta_i));$

$\mathbb{E}[W_i W_i^\top] = \mathbb{E}[W_i]\mathbb{E}[W_i]^\top + Var(W_i);$

$\mathbb{E}[W_i b_i] = \mathbb{E}[W_i]\mathbb{E}[b_i] + Cov(W_i, b_i)$*, with* $Cov(W_i, b_i)$ *the* d *first components of the last row of* $Var((W_i, b_i))$*.*

Proposition 4 *Let* $D_{t-1} = (h_s^\top)_{s=1..t-1}$ *be the* $(t-1) \times d$ *matrix of states until* $t-1$*,* $D_{1..t-2}$ *the matrix of the* $t-2$ *first rows of* D_{t-1} *and* $D_{i:2..t-1}$ *the vector of the* $t-2$ *last components of the i-th column of* D_{t-1}*. Then, for* $t \geq 2$ *the best variational distribution* $q^*_{\theta_i}$ *is a Gaussian* $\mathcal{N}(A_{i,t-1}^{-1} b_{i,t-1}, A_{i,t-1}^{-1})$ *with:*

$$\begin{cases} A_{i,1} = \dfrac{I}{\alpha^2};\ b_{i,1} = 0 \\ A_{i,t-1} = \dfrac{I}{\alpha^2} + \dfrac{\mathbb{E}[D_{1..t-2}^\top D_{1..t-2}]}{\delta^2};\ b_{i,t-1} = \dfrac{\mathbb{E}[D_{1..t-2}]^\top}{\delta^2}\mathbb{E}[D_{i:2..t-1}] \ \textit{for}\ t > 2 \end{cases}$$

where $\qquad \mathbb{E}[D_{1..t-2}^\top D_{1..t-2}] = \sum_{s=1}^{t-2} (\mathbb{E}[h_s]\mathbb{E}[h_s]^\top + Var(h_s))$

Proposition 5 *For each action* i*, we note* $\mathcal{T}_{i,t-1}$ *the set of iterations where* i *has been played before iteration* t*, i.e.,* $\mathcal{T}_{i,t-1} = \{s, i \in \mathcal{K}_s \textit{ for } 1 \leq s \leq t-1\}$*. We also note* $M_{i,t-1} = ((h_s, 1)^\top)_{s \in \mathcal{T}_{i,t-1}}$ *and* $c_{i,t-1} = (r_{i,s})_{s \in \mathcal{T}_{i,t-1}}$*. For every* i *and* $t \geq 1$*, the best distribution* $q^*_{(W_i, b_i)}$ *is a Gaussian* $\mathcal{N}(V_{i,t-1}^{-1} v_{i,t-1}, V_{i,t-1}^{-1})$ *with:*

$$\begin{cases} V_{i,1} = \dfrac{I}{\gamma^2};\ v_{i,1} = 0 \\ V_{i,t-1} = \dfrac{I}{\gamma^2} + \dfrac{\mathbb{E}[M_{i,t-1}^\top M_{i,t-1}]}{\sigma^2}; v_{i,t-1} = \dfrac{\mathbb{E}[M_{i,t-1}]^\top c_{i,t-1}}{\sigma^2} \end{cases}$$

where $\mathbb{E}[M_{i,t-1}^\top M_{i,t-1}] = \sum_{s \in \mathcal{T}_{i,t-1}} (\mathbb{E}[(h_s, 1)]\mathbb{E}[(h_s, 1)]^\top + Var((h_s, 1)))$ *and* $Var((h_s, 1))$ *is* $Var(h_s)$ *with an additional final row and column of* 0*.*

Based on propositions 3,4 and 5, a TS algorithm can be easily derived following a similar process as described in algorithms 1 and 2: at each iteration t, a variational inference step allows one to estimate accurate distributions for the different factors. Then, distributions on h_{t-1}, Θ and each (W_i, b_i) allow one to sample a reward expectation score for each action $\tilde{r}_{i,t} = \tilde{W_i}^\top \tilde{\Theta} \tilde{h_{t-1}} + \tilde{b_i}$. A major benefit compared to the previous approach is that the complexity is much lower in the number of arms. Indeed here, at time t we only need to consider $d^2 + d(t-1) + K(d+1)$ random variables, which does not increase quadratically with the number of arms but with the dimension of the hidden space. Moreover, the same trick than before, which consists in restraining to a time window for learning, allows us to end up with a constant complexity. However, such a method might lead to forget important knowledge. To cope with this, we propose to introduce a memory, which consists in using values computed at time $t-1-S$ as new *priors*. For instance, for θ_i, rather than considering $A_{i,t-1} = I/\alpha^2 + \mathbb{E}[D_{t-2-S..t-2}^\top D_{t-2-S..t-2}]/\delta^2$, it comes down to set $A_{i,t-1} = A_{i,t-1-S} + \mathbb{E}[D_{t-2-S..t-2}^\top D_{t-2-S..t-2}]/\delta^2$. Similar operations can be performed on (W_i, b_i). In this setting, since complexity mostly arises from matrix inversions (which is in $\mathcal{O}(n^3)$ for a $n \times n$ matrix via Gauss-Jordan elimination), required updates at each step for our state-based recurrent bandit is in $\mathcal{O}(K * (d+1)^3 + S * d^3)$, with S the number of historical steps to consider. This appears reasonable compared with bandits with structural dependencies, such as [14] or [17] , which are in $\mathcal{O}(K^2)$ in the first iterations ($d << K$ in most applications).

5 Experiments

In this section, we compare our algorithm with direct relationships defined in section 3, hereafter called `RelationalTS`, and the state-based version defined in section 4, hereafter denoted `StateTS_`d (with d the number of dimensions of the states). For `RelationalTS`, we set standard deviations on rewards to $\sigma = 0.1$ and standard deviations on parameters to $\alpha = 0.1$. Note that, to give more freedom to the bias parameter, we finally set the item $(K+1, K+1)$ to 1 in the prior covariance matrix of each θ_i). For each version of `StateTS`, $\gamma = \alpha = 1$ and $\sigma = \delta = 0.1$. These settings allowed us to observe the best average results. We consider a time window $S = 200$ and a number of variational inference iterations set to $nbIt = 10$. We also consider a memory for every version of our approach (see end of section 4), as it allowed us to observe a slight gain of performances in every tested setting, while preventing from dramatic forgetting (e.g., if no activity is recorded during S iterations).

To the best of our knowledge, no algorithm already treats the recurrent setting we introduced. As baselines, we consider the following state-of-the-art policies, in addition to a random policy that chooses arms uniformly at each step:

- Algorithms for the stationary case: two combinatorial UCB policies `CUCB` [7] and `CUCBV` [21] and a Thompson Sampling algorithm `TS` designed for Gaussian rewards [2]. Those algorithms, while not designed to deal with time-varying rewards, could at least discover the bias part of the reward distributions.

- Algorithms for non-stationary rewards: non-stationary UCB-based policies `D-UCB` and `SW-UCB` [19] which can be adapted to the multiple-plays formulation by selecting the top-k arms with highest UCB scores. Note that `D-UCB` incorporates a discount factor while `SW-UCB` uses a sliding window in order to deal with the changes of expected rewards values through time. We tuned their parameter according to remarks 3 and 9 in [19], which allowed us to observe the best average results in our experiments.

Note that, while graph based approaches such as [17] could appear close to our work, they are not applicable in our setting. They are indeed usually designed for a finite set of decision situations (mostly users for which items have to be recommended), with relationships between situations (i.e., users with similar behaviors) and some observed features for the available actions (knowledge about the items to be recommended), which is not the case here.

5.1 Artificial Data

Two different sets of simulated data are considered:

- **XP1:** Rewards are generated following the relational model described in section 3 (by taking $\sigma = \alpha = 1$ and $\mu_i = 0$ for every arm i. A matrix Θ for each action i and an initial rewards vector R_1 are randomly generated according to their prior distributions [7]. Then, rewards are iteratively generated following the relational model of section 3.1.
- **XP2:** Rewards are generated following cycles defined for each action: the horizon of T iterations is split in periods of 25 iterations. For each action i, we uniformly sample 4 different means $\mu_{i,j} \in]0;1[$ (with $j \in \{0..3\}$). Then, at each iteration $t \in \{1, ..., T\}$ corresponds a period $per_t = (t \div 25)$. For each action i, a reward is sampled for iteration t from a Gaussian $\mathcal{N}(\mu_{i,x}, 1)$, with $x = per_t \mod 4$.

For both sets of experiments, a small version, with $K = 30$ arms and $T = 1000$ iterations, and a large version, with $K = 200$ and $T = 10000$, are considered. 100 datasets of each kind are generated, results are averages on these different datasets. While **XP1** aims at observing performances when direct temporal relationships exist between arms utility, **XP2** aims at assessing the approaches for settings with implicit structural dependencies, when some cycles exist in the data.

We present the results in term of cumulative reward at the last iteration as a function of k, in figure 2 for **XP1** (up) and **XP2** (bottom). In both pairs of plots, the curves on the left concern experiments on the small dataset ($K = 30$), while those on the right correspond to experiments on the larger one ($K = 200$). Note that, due to its complexity, `RelationalTS` has only been tested on the small versions of the datasets. Note also that the bell-shaped aspect of the curves of **XP1** is due to the fact that rewards can be negative. Hence, the final cumulative reward can decrease even with large k values. This is not the case for **XP2** where rewards are only positive.

For **XP1**, `RelationalTS` performs better than any other tested algorithm. While all baselines are only able to capture differences in average utilities (due to the bias component), our relational recurrent bandit efficiently discovers dependencies between arms to capture non-stationarity of rewards. On both datasets, there is no clear improvement with `D-UCB` and `SW-UCB` compared to traditional policies such as `CUCB`, except on the large dataset for `D-UCB` where this approach succeeds to capture some tendencies, but whose results remain significantly lower than the recurrent approaches proposed in this paper. On the other hand, despite

[7] Note however that, to insure a non divergent model, Θ must be chosen such that $\lambda_{max}(\Theta^\top \Theta) \leq 1$, with $\lambda_{max}(A)$ the maximal eigenvalue of a matrix A (see the supplementary material for more details).

it does not explicitly try to catch the model used for the data generation, our `StateTS` approach obtains very good results, while not as good as `RelationalTS` which exactly follows the underlying assumptions, but for a much lower complexity. This allows impressive results on the large dataset, where `RelationalTS` cannot be employed by encoding relational dependencies in a low-dimensional hidden space. We observe that the performance of our `StateTS` algorithm increases with the number of dimensions of the hidden space. However, on the large dataset, even if the rate of improvement between 5 and 10 or 10 and 20 dimensions is high, we notice that it then decreases and reaches a limit.

On the other hand, on **XP2**, `RelationalTS` is not able to outperform classical approaches. It has no mechanism to handle the non-stationarity of the dataset, where cycles of reward distributions are observed. However, our `StateTS` approach, which models utility distributions by hidden states of the system, is able to capture implicit dependencies and obtains interesting results on both small and large datasets. This highlights an additional capability of our state-based approach, which is able to adapt to various configurations of non-stationarity.

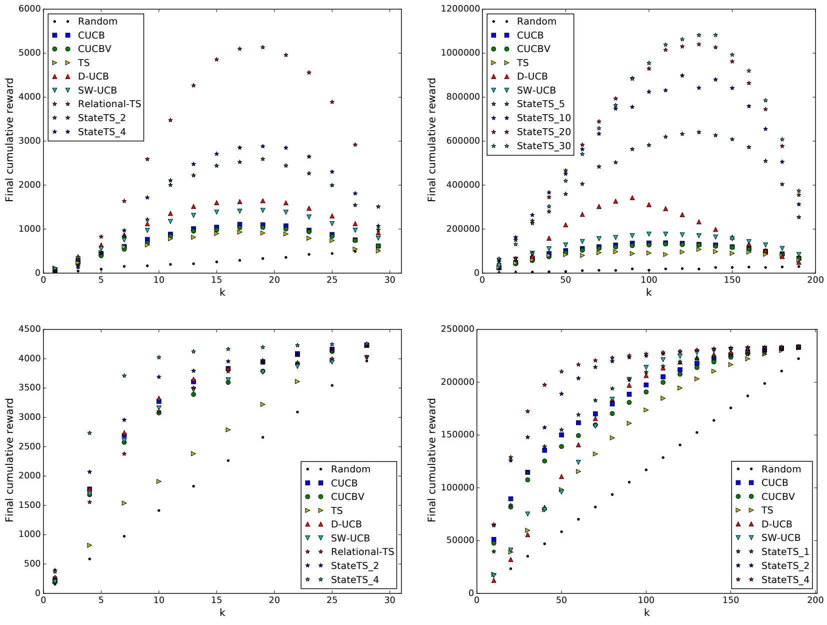

Fig. 2. Final cumulative reward w.r.t. k, for the small (on the left) and the large (on the right) version on the dataset for $XP1$ (up) and $XP2$ (bottom)

5.2 Real Data

Finally, two experiment sets on real data have been conducted:

- Car traffic measurements: we are given a total of $K = 30$ sensors to measure the traffic at different locations in the city of Paris[8] each hour in July 2015 ($T = 744$ time steps). The task is to efficiently select $k = 5$ sensors to monitor at each step in order to maximize the accumulated measured scores. This corresponds to a prediction of where traffic jams are the most likely to happen at each step, knowing some partial observation of the past. While rather artificial, this task simulates real-world settings where one has to secure a given area, but resources for observations and actions are limited and must be focused on areas requiring emergency responses.
- Social data capture: this task introduced in [21] aims at selecting users to follow on a social media such as Twitter to maximize the amount of useful collected data w.r.t. a need. At each iteration, the process has to choose $k = 50$ users to follow during a certain period of time (30 minutes in our experiments). All messages posted from these users during the period are collected, and their usefulness is determined by a reward function. We used a dataset collected from $K = 500$ users on Twitter during the Olympic games of 2016. The targeted users were those that were the first to use words "#Rio2016", "#Olympics", "#Olympics2016" or "#Olympicgames" in a preliminary capture from the random stream API of Twitter. The dataset contains 15 010 322 messages. We used a reward function that returns probability scores for messages to address politics, according to a logistic regression model learned on *20 Newsgroups*[9] for this thematic (messages are represented as TF-IDF bags of words). If a user i posts multiple messages or if its messages are retweeted during a period t, rewards for this user are added to form $r_{i,t}$.

We present results in figure 3 in term of cumulative reward over time, for Car traffic measurements on the left and for Social data capture on the right. For the experiment on social data capture, given the number of users to deal with, `RelationalTS` is only evaluated for the task on Car traffic measurements (on which it is only slightly better than `StateTS`), the number of arms of the task on social data capture being too high for such a complex approach. For both experiments, there is a high stationary component that allows classical approaches such as `CUCB` to perform quite well. However, considering the past, as it is done with our approaches, allows one to collect more rewards. Indeed, in both cases, there exist locations or users that respectively record more frequent vehicles or post more useful contents than others, but considering assumptions about the state of the system allows one to improve the predictions. For instances, crowded places or active users can vary according to the time of the day. Past rewards allow one to identify these different situations.

6 Conclusion

In this paper, we have proposed a new multi-armed bandit problem that considers relations between arms' rewards, based on a linear recurrent model, in a

[8] http://opendata.paris.fr/ [9] http://qwone.com/jason/20Newsgroups/

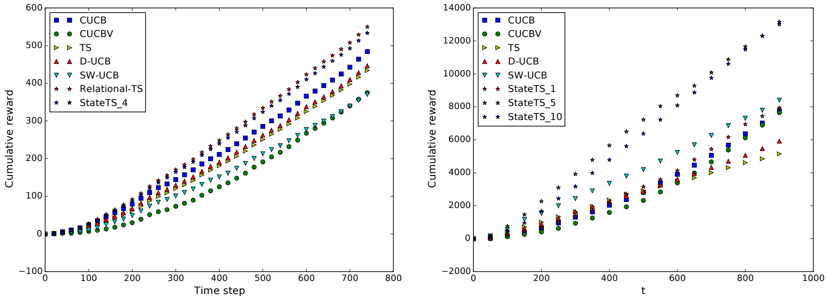

Fig. 3. Results on real data (car traffic on the left, social data capture on the right)

multiple-plays setting. In this case, not only the weights of the linear model are unknown to the learner, but a majority of the rewards component - which act as features - are hidden, since the agent is only allowed to play a restricted set of arms at each iteration. We proposed two new Thompson sampling algorithms that are able to leverage past observations via variational inference. While approximations performed avoid theoretical guarantees for the regret, they allowed us to obtain very interesting results on both synthetic and real-world datasets.

Future work concerns the introduction of non-linearity in our state-based model, notably by the use of Bayesian Recurrent Neural Networks (see [28] for instance), as transition functions between successive states and encoding/decoding functions of the rewards. Since they also use variational inference for obtaining posterior distributions of hidden variables, their application for relational recurrent exploration/exploitation problems appears really promising.

Acknowledgments The work was supported by the IRT SystemX and the ANR project LOCUST (2015-2019, ANR-15-CE23-0027).

References

1. Abbasi-yadkori, Y., Pál, D., Szepesvári, C.: Improved algorithms for linear stochastic bandits. In: NIPS (2011)
2. Agrawal, S., Goyal, N.: Analysis of thompson sampling for the multi-armed bandit problem. In: COLT (2012)
3. Agrawal, S., Goyal, N.: Thompson sampling for contextual bandits with linear payoffs. ICML (2013)
4. Audibert, J.Y., Bubeck, S.: Minimax policies for adversarial and stochastic bandits. In: COLT (2009)
5. Audibert, J.Y., Munos, R., Szepesvari, C.: Tuning bandit algorithms in stochastic environments. In: ALT (2007)
6. Audiffren, J., Ralaivola, L.: Cornering stationary and restless mixing bandits with remix-ucb. In: NIPS. pp. 3339–3347 (2015)
7. Auer, P., Cesa-Bianchi, N., Fischer, P.: Finite-time analysis of the multiarmed bandit problem. Mach. Learn. (2002)

8. Beal, M.J.: Variational Algorithms for Approximate Bayesian Inference. Ph.D. thesis, Gatsby Computational Neuroscience Unit, University College London (2003)
9. Besbes, O., Gur, Y., Zeevi, A.: Stochastic multi-armed-bandit problem with non-stationary rewards. In: NIPS (2014)
10. Bubeck, S., Stoltz, G., Szepesvári, C., Munos, R.: Online optimization in x-armed bandits. In: NIPS (2009)
11. Buccapatnam, S., Eryilmaz, A., Shroff, N.B.: Stochastic bandits with side observations on networks. In: SIGMETRICS (2014)
12. Caron, S., Kveton, B., Lelarge, M., Bhagat, S.: Leveraging side observations in stochastic bandits. In: UAI (2012)
13. Carpentier, A., Valko, M.: Revealing graph bandits for maximizing local influence. In: AISTATS. Seville, Spain (2016)
14. Cesa-Bianchi, N., Gentile, C., Zappella, G.: A gang of bandits. In: NIPS (2013)
15. Chapelle, O., Li, L.: An empirical evaluation of thompson sampling. In: NIPS. Curran Associates, Inc. (2011)
16. Chen, W., Wang, Y., Yuan, Y.: Combinatorial multi-armed bandit: General framework and applications. In: ICML (2013)
17. Claudio, G., Shuai, L., Giovanni, Z.: Online clustering of bandits. In: ICML (2014)
18. Dani, V., Hayes, T.P., Kakade, S.M.: Stochastic linear optimization under bandit feedback. In: COLT (2008)
19. Garivier, A., Moulines, E.: On upper-confidence bound policies for switching bandit problems. In: ALT (2011)
20. Garivier, A.: The kl-ucb algorithm for bounded stochastic bandits and beyond. In: COLT (2011)
21. Gisselbrecht, T., Denoyer, L., Gallinari, P., Lamprier, S.: Whichstreams: A dynamic approach for focused data capture from large social media. In: ICWSM (2015)
22. Kalman, R.E.: A new approach to linear filtering and prediction problems. Transactions of the ASME–Journal of Basic Engineering 82(Series D), 35–45 (1960)
23. Komiyama, J., Honda, J., Nakagawa, H.: Optimal regret analysis of thompson sampling in stochastic multi-armed bandit problem with multiple plays. In: ICML (2015)
24. Lai, T., Robbins, H.: Asymptotically efficient adaptive allocation rules. Advances in Applied Mathematics 6(1), 4 – 22 (1985)
25. Li, L., Chu, W., Langford, J., Schapire, R.E.: A contextual-bandit approach to personalized news article recommendation. In: WWW (2010)
26. Mannor, S., Shamir, O.: From bandits to experts: On the value of side-observations. In: NIPS (2011)
27. Ortner, R., Ryabko, D., Auer, P., Munos, R.: Regret bounds for restless markov bandits. Theor. Comput. Sci. 558, 62–76 (2014)
28. Pczos, B., Lrincz, A., Ghahramani, Z.: Identification of recurrent neural networks by bayesian interrogation techniques. JMLR (2009)
29. Richard, C., Alexandre, P.: Unimodal bandits: Regret lower bounds and optimal algorithms. In: ICML (2014)
30. Slivkins, A., Upfal, E.: Adapting to a changing environment: the brownian restless bandits. In: COLT (2008)
31. Tekin, C., Liu, M.: Online learning of rested and restless bandits. IEEE Trans. Information Theory 58(8), 5588–5611 (2012)
32. Thompson, W.: On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. American Math. Soc. 25, 285–294 (1933)
33. Whittle, P.: Restless bandits: Activity allocation in a changing world. Journal of Applied Probability 25, 287–298 (1988)