Improving Online Marketing Experiments with Drifting Multi-armed Bandits

Giuseppe Burtini, Jason Loeppky and Ramon Lawrence
University of British Columbia, Kelowna, Canada

Keywords: Non-stationary and Restless Bandits, Multi-armed Bandits, Online Marketing, Web Optimization.

Abstract: Restless bandits model the exploration vs. exploitation trade-off in a changing (non-stationary) world. Restless bandits have been studied in both the context of continuously-changing (drifting) and change-point (sudden) restlessness. In this work, we study specific classes of drifting restless bandits selected for their relevance to modelling an online website optimization process. The contribution in this work is a simple, feasible weighted least squares technique capable of utilizing contextual arm parameters while considering the parameter space drifting non-stationary within reasonable bounds. We produce a reference implementation, then evaluate and compare its performance in several different true world states, finding experimentally that performance is robust to time drifting factors similar to those seen in many real world cases.

1 INTRODUCTION

Lai and Robbins (1985) introduced the standard stochastic, finite-armed, multi-armed bandit problem and produced an efficient[1] solution where the rewards of a given arm are stationary and *independent and identically distributed* (i.i.d.) with no *contextual information*. A large number of algorithms have been proposed since Lai and Robbins (1985) including upper-confidence bound techniques (Auer et al., 2002), ε-exploration approaches (Watkins, 1989; Vermorel and Mohri, 2005), probability matching techniques (Agrawal and Goyal, 2012) and others. Many variants of the initial problem have also been investigated in the literature including the many- or infinite-armed, combinatorial, adversarial, contextual, and non-stationary cases.

In this work, we explore the variant of the problem where the reward distributions may be changing in time. Specifically, we explore the case where the reward distributions may be *drifting* in time and contextual information is available. This replicates the online marketing scenario, where an experiment to modify a webpage may be set up at a given time and run indefinitely with the aim of maximizing revenue. The web environment has context: user factors (web browser, operating system, geolocation), world factors (day of the week), and arm factors (grouping of modifications). Utilization of contextual variables allows learning behavior for classes of users and observable world effects in order to improve the results.

[1] In the same work, Lai and Robbins demonstrate an asymptotic lower bound of *regret* of $O(\log N)$ for any algorithm.

2 BACKGROUND

A (finite-armed) multi-armed bandit problem is a process where an agent must choose repeatedly between K independent and unknown *reward distributions* (called *arms*) over a (known or unknown) time horizon T in order to maximize his total reward (or equivalently, minimize the total *regret*, compared to an oracle strategy). At each time step, t, the *strategy* or *policy* selects (*plays*) a single arm i_t and receives a reward of r_{i_t} drawn from the ith arm distribution which the policy uses to inform further decisions. In our application, individual arms represent webpage modifications with the goal of maximizing desired user behavior (sales, time engaged, etc.).

2.1 Regret

We define expectation-expectation regret ($\bar{R}^E$) as our objective variable of interest, computed as the difference between the expected value of the best arm (per play) minus the expected value of the arm that is se-

Burtini G., Loeppky J. and Lawrence R..
Improving Online Marketing Experiments with Drifting Multi-armed Bandits.
DOI: 10.5220/0005458706300636
In *Proceedings of the 17th International Conference on Enterprise Information Systems* (ICEIS-2015), pages 630-636
ISBN: 978-989-758-096-3

lected by the algorithm, conditional on all contextual variables, with expectation taken over repeated plays of an arm at a fixed point in time. This distinguishes from the stochastic measures of regret computed using empirical estimates of the mean or observed reward values and from the *adversarial* measures of regret where the best (oracle) arm is taken only in expectation over all plays. Formally, our objective value is

$$\mathrm{E}[\bar{R}^E] = \mathrm{E}_{\mathbb{P}}\left[\left(\max_{i=1,2,\ldots,K}\sum_{t=1}^{T}\mathrm{E}_{\mathbb{A}}[r_{i,t}]\right) - \sum_{t=1}^{T}\mathrm{E}_{\mathbb{A}}[r_{\underline{i}_t,t}]\right] \quad (1)$$

Where $\underline{i}_t$ is the arm selected (*played*) at time t, $\mathrm{E}_{\mathbb{P}}$ is expectation taken over repeated plays and $\mathrm{E}_{\mathbb{A}}$ expectation taken over the arm distribution at a given time t.

2.2 UCB Algorithms

A well-studied family of algorithms for the stationary bandit problem are called *Upper Confidence Bound* (UCB) strategies. In a UCB strategy, at each time step t, the arm is played that maximizes the average empirical payoff ($A(x)$) plus some padding function ($V(x)$). The best basic UCB strategy we explore is UCB-Tuned (Auer et al., 2002) where $A(x)$ is defined as the total observed payoff divided by its number of plays and $V(x)$ uses the empirical estimate of the arm's variance plus a factor to achieve a upper bound of the true value in high probability. This algorithm is intended for stationary multi-armed bandits, however, we test it in the non-stationary case.

2.3 Restless Bandits

Change-point analysis, also known as change detection or *structural breakpoints* modelling, is a well-studied problem in the applied stochastic process literature. Intuitively, a change-point is a sudden, discrete or "drastic" (non-continuous) change in the shape of the underlying distribution. In an offline fashion, change-points may be detected efficiently and with an adequate set of tuneable parameters with clustering algorithms. For bandits, the problem is necessarily an online problem and offline algorithms for change point detection are not feasible. The basic idea of online change-point bandits is to use a mechanism to detect change-points, generally parameterized for some acceptable *false alarm rate*, and then utilizing some mechanism to "forget" learned information after each change-point as necessary.

Hartland et al. (2006) propose an algorithm called Adapt-EvE based on the UCB-Tuned algorithm (Auer et al., 2002). Adapt-EvE uses the frequentist Page-Hinckley test to identify change-points. Upon detection of a change-point, Adapt-EvE treats the problem as a meta-bandit problem. That is, a second layer of bandit optimization is instituted with two arms: (1) continues using the learned data and (2) restarts the UCB-Tuned algorithm from scratch. This meta-bandit forms a hierarchical strategy that can be expected to efficiently evaluate the *cost* in regret of each detected change. This technique was the winning technique in the PASCAL Exploration vs. Exploitation challenge in 2006 (Hussain et al., 2006) demonstrating its ability to handle both drifting and change-point type bandits.

Kocsis and Szepesvári (2006) present a variant of UCB-Tuned called DiscountedUCB which applies a continuous discount factor to the estimates in time. Garivier and Moulines (2008) introduce Sliding Window UCB (SW-UCB) parameterized by a window length and show it performs similarly to DiscountedUCB contingent on appropriately selected parameterizations.

Mellor and Shapiro (2013) present an online Bayesian change-point detection process for *switching* (discrete change) bandits with constant *switching rate* – the frequency with which the distributions change – in the contexts where switching occurs globally or per-arm and when switching rates are known or must be inferred. Their algorithm is probability matching based, but, as presented does not support contextual variables. Further, their technique addresses a bandit with switching behavior, rather than drifting behavior as explored in this work.

2.3.1 Stochastic Drift

In time-series analysis, *stochastic drift* is used to refer to two broad classes of non-stationarity in the population parameter being estimated: (1) cyclical or modelable drift that arise because of model misspecification and (2) the random component. Often it is possible to *detrend* non-stationary data by fitting a model that includes time as a parameter. Where the function of time is well-formed and appropriate for statistical modelling, a *trend stationary* model can be found with this detrending process. For models where detrending is not sufficient to make a process stationary, *difference stationary* models may fit, where the differences between values in time Y_t and Y_{t-n} can be represented as a well-formed function appropriate for statistical modelling.

Difference stationary models are represented with autoregressive models. The generalized representation of the simple autoregressive model is referred to as $AR(n)$ where n is the number of time steps back the current value maintains a dependency upon.

$$AR(n):\ Y_t = \alpha_0 + \alpha_1 Y_{t-1} + \alpha_2 Y_{t-2} + \cdots + \alpha_n Y_{t-n} + \varepsilon_t \quad (2)$$

Where ε_t is the error term with the normal characteristics of zero mean ($E[\varepsilon_t] = 0$), variance σ^2 and independence across times ($E[\varepsilon_t \varepsilon_s] = 0, \forall t \in \{t \neq s\}$) after fitting the autoregressive correlations. If these two detrending strategies are not sufficient to make a given process stationary, more complex filters such as a band-pass or Hodrick-Prescott filter may be applied.

2.4 Generalized Linear Bandits

Filippi et al. (2010) use generalized linear models (GLMs) for bandit analysis, extending the work of (Dani et al., 2008; Rusmevichientong and Tsitsiklis, 2010) to utilize the UCB strategy of (Auer et al., 2002) and proving (high-probability) pseudo-regret bounds under certain assumptions about the link function and reward distributions. In some sense, our work extends the Filippi et al. result to an experimental analysis within the non-stationary case, as well as introducing a Thompson sampling based strategy for integrating GLMs, rather than the UCB technique.

2.5 Probability Matching

Probability matching, especially randomized probability matching known as *Thompson sampling*, has been explored in the reinforcement learning (Wyatt, 1998; Strens, 2000), and multi-armed bandits literature (Mellor and Shapiro, 2013; Scott, 2010; Kaufmann et al., 2012; May et al., 2012; Chapelle and Li, 2011; Granmo and Glimsdal, 2013; Durand and Gagné, 2014). The basic technique is to express a model that matches the probability of playing a particular arm with the probability of that arm being the best, conditional on all the information observed thus far. That is, select arm $i \sim P[r_i \text{ is max}]$. In general, this technique benefits from the same uncertainty "boosting" that the UCB policies achieve; for the purpose of exploration, it is beneficial to "boost" the predictions of uncertain actions (Chapelle and Li, 2011). This technique has become very popular of recent as various experimental and specific model theoretical analyses (Kaufmann et al., 2012) have demonstrated regret comparable or better than the popular upper confidence bound (Auer, 2003) and Exp4 (Auer et al., 2002) derived techniques. Recently, scalability has been studied by introducing a bootstrap-based variant of Thompson sampling (Eckles and Kaptein, 2014). Importantly, practical implementation of the probability matching technique is simple in a modern statistical computing environment.

A number of results have shown improvements by performing *optimistic Thompson sampling* (Chapelle and Li, 2011; May et al., 2012) where one only considers the positive uncertainty surrounding an arm estimate. Unlike UCB-based policies, traditional Thompson sampling both increases (if the draw is above the point estimate of the mean) *and* decreases (if the draw is below the point estimate of the mean) a prediction, depending on the sample draw; for the purpose of maximizing reward (minimizing regret), the decrease appears to have no benefit. For this reason, optimistic Thompson sampling, which only increases predictions proportional to their uncertainty, outperforms the traditional technique.

3 OVERVIEW OF THE APPROACH

The general technique we experiment with is to fit a regression model of varying form to the data and then to utilize the technique of optimistic Thompson sampling to predict arm payoffs in the next iteration of the algorithm. We explore and compare two primary models, the autoregressive, time-detrended approach and the weighted least squares approach for handling non-stationarities with a regression framework.

3.1 Autoregression and Detrending

Formally, we fit a model

$$Y_{t,i} = \alpha_t + \mathrm{AR}_i(p) + \mathrm{Trend}_i(t) + A_{t,i} + \varepsilon_{t,i} \quad (3)$$

Where $\mathrm{Trend}(t)$ is a function representing the expected time trend, $\mathrm{AR}(p)$ is the autoregressive term of order p and $Y_{t,i}$ is the expected reward for arm i at time t. In practice, this model is generally fit as a model of Y_t with binary ("dummy") variables $A_{t,i}$ and relevant interaction terms indicating which arm is detected. In our experimental results, we explore how variations (especially *overspecification* of the functional form) in the "correctness" of the selection of $Trend(t)$ affect the overall results. This model, fit with the ordinary least squares technique, the ridge regression technique (Tikhonov, 1963) or the Bayesian conjugate prior technique, returns an estimated set of time-detrended, plausibly stationary[2] coefficients $\hat{\beta}$ and estimates of their standard errors $\widehat{SE}(\hat{\beta})$. This model can be readily extended to contain any contextual variables, such as demographic information about the user (in the web optimization context) or grouping criteria on the arms to improve the learning rate.

[2] As long as the detrending process successfully removed the non-stationarity.

Combined, we follow in standard experiment design terminology and call the terms in our model α, $AR(p)$, $Trend(t)$, and $A_{t,i}$ the *design matrix* and refer to it as X.

3.2 Penalized Weighted Least Squares

The *weighted least squares* (WLS) process introduces a multiplicative weighting of "reliability" for each observation, resulting in a technique which minimizes the reliability-adjusted squared errors. In the multi-armed bandit context with drifting arms (without any *a priori* knowledge of the functional form of the drift), the weights are set to the inverse of their recency, indicating that at each time step t, older data provides a less reliable estimate of the current state.

Intuitively, weighted least squares provides a simple, well-explored, highly tractable technique to discount the confidence of old data, increasing predictive uncertainty as time progresses. This is a desirable quality within the context of restless bandits as it appropriately accounts for the growing predictive uncertainty of old observations.

Formally, the weighted least squares procedure picks $\hat{\beta}$, coefficients on a set of variables, X, called the independent variables (or regressors), according to the equation $\hat{\beta} = (X^T \Omega X)^{-1}(X^T \Omega y)$ where Ω is the matrix of weights and y is the rewards as observed (or, in general, the regressand). Standard errors of the coefficients are also computed, producing an estimate of the standard deviation of our estimators.

To apply the weighted least squares procedure, we follow in the work of Pavlidis et al. (2008) which uses a standard linear regression to compute the estimates of each arm and the work of the LinUCB algorithm (Li et al., 2010) which applies a non-weighted penalized linear regression to compute estimates of the payoff for each arm. As we are *a priori* uncertain about the functional form of the non-stationarity in our bandit arms, we experiment with a variety of time weighting techniques – logarithmic, with varying scale and base; linear, with varying polynomials; exponential, with varying coefficients; and sinusoidal – demonstrating the generality of this technique. In all cases we strictly decrease the weight of a sample as it becomes further in time from our current prediction time. When additional information about the form of non-stationarity is available, weights can be specified appropriately to reduce the predictive uncertainty.

3.3 Optimistic Thompson Sampling

Extending the LinUCB algorithm, we propose a technique that exploits the assumptions of the linear model and the probability matching technique of Thompson sampling. Based on the assumption of normality, the regression coefficients, $\hat{\beta}$, are normal and hence the predictions $\hat{y}_t$ are normal. We then optimistically sample (drawing only values above the mean) from a normal distribution with mean $\sum_i (\hat{\beta}_i \cdot x_{i,t})$ and variance $\sum_i (\widehat{\mathrm{Var}}(\hat{\beta}_i) \cdot x_{i,t}^2)$ to approximate $\hat{y}_t$. A more general form of this fundamentally Bayesian algorithm can be constructed utilizing the techniques of Bayesian regression (Minka, 2001) at the cost of higher computational complexity.

4 SIMULATION ENVIRONMENT

To test our combined strategies and produce objective comparisons, we produce a synthetic simulator with a wide variety of "true worlds" (unobserved to the agent) including arm distribution type and parameters, arm count, and drift type from a set of func-

```
Input: λ the penalty factor
       w(t) the weighting strategy

function penalizedWLS(X, y, Ω, λ)
 β̂ = (X^T ΩX + λI)^-1 (X^T Ωy)
 s^2 = (y − β̂X)^T (y − β̂X)/(n − p)
 Var̂(β̂) = diag[s^2 (X^T ΩX + λI)^-1]
end

function optimisticSampler(β̂, Var̂(β̂))
  samples = []
  for each arm
     estimates[arm] = sample estimated reward
          payoff ŷ_t
  end
  arg max_arm estimates
end

function generateWeightMatrix(t)
  Ω = []
  foreach i < t
     append w(i) to Ω
  end
  I · Ω
end

X = y = Ω = []
t = 0
while playing
  r_i,t = play arm optimisticSampler(
        penalizedWLS(X, y, Ω, λ)
  )
  extend X, the design matrix
  append r_i,t to rewards history y
  Ω = generateWeightMatrix(t++)
end
```

Figure 1: Pseudocode of combined algorithm.

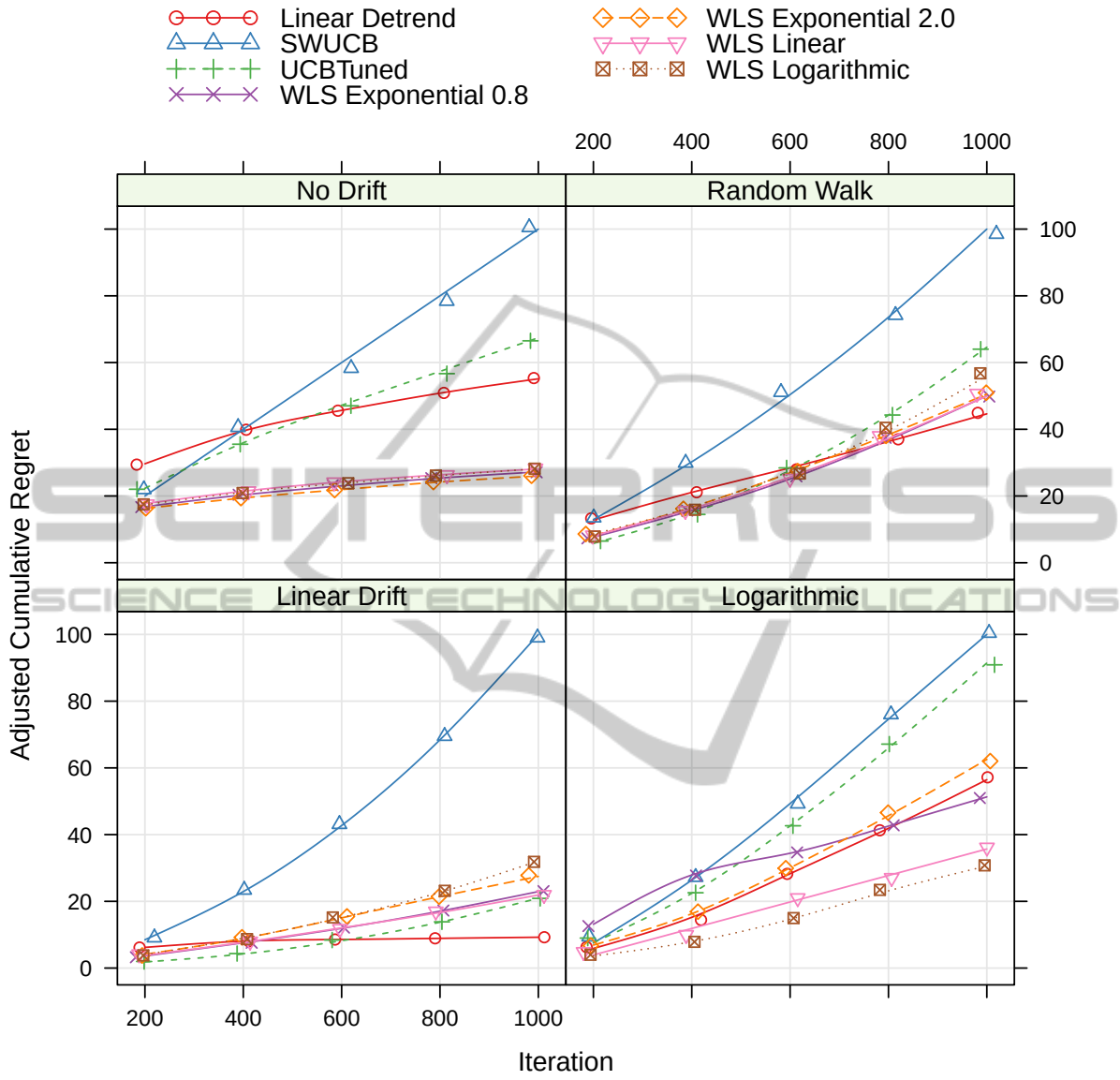

Figure 2: Adjusted average cumulative regret of selected algorithms over 1,000 replicates of all worlds and true drift forms.

tional forms including random walk, exponential random walk, logarithmic, linear (in varying degree), exponential and periodic drift (sinusoidal over varying periods). Each form of drift is parameterized by a randomly drawn real number constrained to be within the same order of magnitude as the arm payoffs in its simulation world which determines the scale of the parameterization. To validate our results against plausible modelling error, our simulator validates each algorithm in the PASCAL challenge environment (Hussain et al., 2006) and in the unbiased replay technique of Li et al. (2011).

We present the combined algorithm, parameterized in degrees of autoregression, detrending and functional form of our weighted least squares discounting process in pseudocode in Figure 1. Of the n data points, the first p must be collected using another method (uniformly at random, in our case) to provide enough degrees of freedom to fit the regression model with p variables.

5 EXPERIMENTAL RESULTS

In the results presented, we omit ε-greedy, UCB1,

DiscountedUCB and others as they were strictly outperformed by UCB-Tuned or SW-UCB for all parameter choices tested. We also show only four representative drifting worlds due to space constraints. Across all true worlds, we find in general that a detrending term congruent with the *true drift* form (e.g. *linear detrend* in the linear drift quadrant of Figure 2) outperforms all other strategies in the long run, producing a *zero-regret strategy* (Vermorel and Mohri, 2005) for restless bandits where the functional form of restlessness is known. Similarly, we find that utilizing a weighting function which closely approximates the true drift performs well in most cases. Surprisingly, we find that linear detrending is an effective technique for handling the *random walk*, a result that is robust to variations in the step type and scale of the random walk. Unintuitively, WLS techniques also perform strongly even in the case when there is no drift.

In these experiments, we find no convincing evidence for a general application for detrending in polynomial degree greater than one or autoregression of any level in our model. Both autoregression and higher degree polynomials strictly reduce regret *if* the true world trend is autoregressive or determined, even partially, by the chosen form. We find the linear weighted least squares technique (weights set to the inverse of t) to be the most robust technique over all experiments, suggesting it is the strongest technique in the case of no *a priori* information on the form of drift: having the lowest mean total regret (20.8), lowest standard deviation across all drift types (11.8) and the lowest 75th (worst-) percentile regret (26.6). Due to space constraints and difficulties reproducing some challenge results, we do not present the PASCAL challenge data here, however, our preliminary results show similar promise with the weighted least squares technique.

6 CONCLUSION

In this work, we have implemented and experimented with integrating time series techniques and weighted least squares with the highly successful Thompson sampling technique to extend the state of the art in handling restless regression bandits. We present evidence that weighted least squares techniques provide a strong solution for ongoing multi-armed bandit optimization in an uncertain-stationarity world even without an *a priori* understanding of the modality of drift. The technique presented allows bandits with context to handle drift in diverse form and operationalizes monotonic discounting in a simple, easy to implement regression framework. This provides a viable solution to the ongoing online marketing experimentation problem. Future work will explore how contextual factors improve results for web optimization, perform real world experiments on online marketing optimization, and derive formal bounds for the interaction between weighted least squares and optimistic Thompson sampling.

REFERENCES

Agrawal, S. and Goyal, N. (2012). Analysis of Thompson sampling for the multi-armed bandit problem. *CoRR*, abs/1111.1797.

Auer, P. (2003). Using confidence bounds for exploitation-exploration trade-offs. *The Journal of Machine Learning Research*, 3:397–422.

Auer, P., Cesa-Bianchi, N., and Fischer, P. (2002). Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256.

Chapelle, O. and Li, L. (2011). An empirical evaluation of Thompson sampling. In *Advances in Neural Information Processing Systems*, pages 2249–2257.

Dani, V., Hayes, T. P., and Kakade, S. M. (2008). Stochastic linear optimization under bandit feedback. In *COLT*, pages 355–366.

Durand, A. and Gagné, C. (2014). Thompson sampling for combinatorial bandits and its application to online feature selection. In *AI Workshops*.

Eckles, D. and Kaptein, M. (2014). Thompson sampling with the online bootstrap. *arXiv:1410.4009*.

Filippi, S., Cappe, O., Garivier, A., and Szepesvári, C. (2010). Parametric bandits: the generalized linear case. In *Advances in Neural Information Processing Systems*, pages 586–594.

Garivier, A. and Moulines, E. (2008). On upper-confidence bound policies for non-stationary bandit problems. *arXiv preprint arXiv:0805.3415*.

Granmo, O.-C. and Glimsdal, S. (2013). Accelerated Bayesian learning for decentralized two-armed bandit based decision making with applications to the Goore Game. *Applied Intelligence*, 38(4):479–488.

Hartland, C., Gelly, S., Baskiotis, N., Teytaud, O., Sebag, M., et al. (2006). Multi-armed bandit, dynamic environments and meta-bandits. In *NIPS*.

Hussain, Z., Auer, P., Cesa-Bianchi, N., Newnham, L., and Shawe-Taylor, J. (2006). Exploration vs. exploitation PASCAL challenge.

Kaufmann, E., Korda, N., and Munos, R. (2012). Thompson sampling: An asymptotically optimal finite-time analysis. In *Algorithmic Learning Theory*, pages 199–213. Springer.

Kocsis, L. and Szepesvári, C. (2006). Discounted UCB. *2nd PASCAL Challenges Workshop*.

Lai, T. L. and Robbins, H. (1985). Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1):4–22.

Li, L., Chu, W., Langford, J., and Schapire, R. E. (2010). A contextual-bandit approach to personalized news article recommendation. In *WWW*, pages 661–670. ACM.

Li, L., Chu, W., Langford, J., and Wang, X. (2011). Unbiased offline evaluation of contextual-bandit-based news article recommendation algorithms. In *WSDM*, pages 297–306. ACM.

May, B. C., Korda, N., Lee, A., and Leslie, D. S. (2012). Optimistic Bayesian sampling in contextual-bandit problems. *The Journal of Machine Learning Research*, 13(1):2069–2106.

Mellor, J. and Shapiro, J. (2013). Thompson sampling in switching environments with Bayesian online change detection. In *AISTATS*, pages 442–450.

Minka, T. P. (2001). *A family of algorithms for approximate Bayesian inference*. PhD thesis, MIT.

Pavlidis, N. G., Tasoulis, D. K., and Hand, D. J. (2008). Simulation studies of multi-armed bandits with covariates. In *UKSIM*, pages 493–498. IEEE.

Rusmevichientong, P. and Tsitsiklis, J. N. (2010). Linearly parameterized bandits. *Mathematics of Operations Research*, 35(2):395–411.

Scott, S. L. (2010). A modern Bayesian look at the multi-armed bandit. *Applied Stochastic Models in Business and Industry*, 26(6):639–658.

Strens, M. (2000). A Bayesian framework for reinforcement learning. In *ICML*, pages 943–950.

Tikhonov, A. (1963). Solution of incorrectly formulated problems and the regularization method. In *Soviet Math. Dokl.*, volume 5, pages 1035–1038.

Vermorel, J. and Mohri, M. (2005). Multi-armed bandit algorithms and empirical evaluation. In *ECML 2005*, pages 437–448. Springer.

Watkins, C. J. C. H. (1989). *Learning from delayed rewards.* PhD thesis, University of Cambridge.

Wyatt, J. (1998). Exploration and inference in learning from reinforcement.