Multi-armed Bandit, Dynamic Environments and Meta-Bandits

Cédric Hartland, Sylvain Gelly, Nicolas Baskiotis, Olivier Teytaud, Michèle Sebag

Multi-Armed Bandit, Dynamic Environments and Meta-Bandits

C. Hartland, S. Gelly, N. Baskiotis, O. Teytaud and M. Sebag
Lab. of Computer Science – CNRS – INRIA
Université Paris-Sud, Orsay, France

Abstract

This paper presents the *Adapt-EvE* algorithm, extending the UCBT online learning algorithm (Auer et al. 2002) to abruptly changing environments. *Adapt-EvE* features an adaptive change-point detection test based on Page-Hinkley statistics, and two alternative extra-exploration procedures respectively based on smooth-restart and Meta-Bandits.

1 Introduction

The Game Theory perspective is gradually becoming more relevant and appealing to Machine Learning (ML), as quite a few application domains emphasize the incompleteness of available information in the learning game (Cesa-Bianchi & Lugosi, 2006). In some cases, the huge volume of available information enforces the use of incremental and/or anytime algorithms (Auer et al., 2002). In other cases, the dynamic nature of the application domain asks for new learning algorithms, able to estimate on the fly the relevance of the training examples, and accommodate these relevance estimates within the learning process (Kifer et al., 2004). One central question for ML in this perspective is that of the balance between Exploration and Exploitation (EvE). For instance in the multi-armed bandit problem, online learning is both concerned with finding the very best option (exploration) and playing as often as possible a good enough option (exploitation), in order to optimize the cumulated reward of the gambler (Auer et al., 2002).

This paper is about online learning in dynamic environments. While online algorithms offer some leeway for accommodating dynamic environments, empirical evidence shows that their Exploration *versus* Exploitation trade-off is not appropriate for abruptly changing environments. In order to adapt online learning to such abrupt changes in the environment, three interdependent questions must be addressed. The first one, referred to as change-point detection (Page, 1954), is concerned with deciding whether some change has occurred beyond the "natural" variations of the environment. The second, referred to as Meta-EvE, is concerned with designing a good strategy for such change moments. On one hand, the change-point detection must trigger some extra exploration; this extra exploration relates to the (partial) forgetting of the recent history. On the other hand, if the change-point detection was a false alarm, the process should quickly recover its memory and switch back to exploitation; otherwise, the extra exploration results in wasting time. Thirdly, the process should be able to adapt the change-point detection mechanism based on what happened during the Meta-EvE episodes. Typically, if the Meta-EvE episode concludes that the change-point detection was a false alarm, the detection thresholds should be increased.

The algorithm presented in this paper, called *Adapt-EvE*, relies on the UCBT algorithm proposed by (Auer et al., 2002), described in Appendix 1 for the sake of completeness. Our contribution is two-fold. Firstly, *Adapt-EvE* incorporates a change-point detection test based on the Page-Hinkley statistics (Page, 1954); parameterized after the desired false alarm detection rate, this test provably minimizes the expected time before detection

(section 2). Secondly, two alternative Meta-EvE strategies are proposed and compared. The first one, γ-restart strategy, proceeds by discounting the process memory. The second one, Meta-Bandit, formulates the Meta-Eve problem as another multi-armed bandit problem, where the two options are: i/ forgetting the whole process memory and playing UCBT accordingly; ii/ discarding the change detection and keeping the same UCBT strategy as before (section 3). Finally, the adjustment of the change point detection criterion is based on a simple multiplicative update of the underlying threshold.

Empirical validation, conducted on the EvE Challenge proposed by (Hussain et al., 2006) and discussed in section 4 demonstrates significant improvement over the baseline UCBT algorithm (Auer et al., 2002). The paper concludes with some perspectives for further research, particularly considering the case of many options.

2 Change point detection

As already mentioned, one question raised by the extension of UCBT to abruptly changing environments is that of detecting the environment changes. Let us assume that the best current option i^* is correctly identified, and let μ^* denote the expected associated reward. Three types of change can occur. In the first case, the best option remains the same but the associated reward μ^* changes (it decreases or increases); in the second case, the reward of another option increases to the point that it outperforms option i^*; in the third case, reward μ^* associated to option i^* abruptly decreases and another option becomes the best one. Only the last type of changes will be considered in this section, leaving the other two cases for further study.

If we consider the series of rewards $x_1, ...x_T$ gathered by playing the current best option i^* in the last T steps, the question is whether this series can be attributed to a single statistical law (null hypothesis); otherwise (change-point detection) the series demonstrates a change in the statistical law underlying the rewards. A most well-known criterion for testing this hypothesis is the Page-Hinkley (PH) statistics (Page, 1954; Hinkley, 1969; Hinkley, 1970; Hinkley, 1971). The PH statistical test involves a random variable m_T defined as the difference between the x_t and their average $\bar{x}_t$, cumulated up to step T; by construction, this variable should have 0 mean if the null hypothesis holds (no change has occurred). The maximum value M_T of the m_t for $t = 1 \dots T$ is also computed and the difference between M_T and m_T is monitored; when this difference is greater than a given threshold λ (depending on the desired false alarm rate), the null hypothesis is rejected i.e. the PH test concludes that a change point has occurred. Further, under some technical hypothesis, the Page-Hinkley test provably ensures the minimal expected time before detection for a given false detection rate (Lorden, 1971).

$$\bar{x}_t = \frac{1}{t} \sum_{\ell=1}^{t} x_\ell$$
$$m_T = \sum_{t=1}^{T} (x_t - \bar{x}_t + \delta)$$
$$M_T = \max\{m_t,\ t = 1 \dots T\}$$
$$PH_T = M_T - m_T$$
$$Return(PH_T > \lambda)$$

Table 1: The Page-Hinkley statistical test

The PH test involves two parameters. Parameter δ, manually adjusted in this paper, corresponds to the magnitude of changes that should not raise an alarm. Parameter λ depends on the desired false detection rate. Increasing λ will entail less false alarms, but might miss some changes. As λ directly controls the exploration-exploitation dilemma, an adaptive control of λ is proposed in section 3.3.

3 Meta Exploration *vs* Exploitation Dilemma

When the change-point detection test is positive, the question becomes to reconsider the balance between exploration and exploitation. Two alternative strategies are proposed to handle the extra-exploration control, referred to as Meta-EVE. The first strategy, γ-restart, is based on discounting the process memory (section 3.1). The second strategy, Meta-Bandit, is based on the formulation of the Meta-EVE problem as another multi-armed

bandit problem (section 3.2).
Independently, section 3.3 tackles the *a posteriori* control of the change-point detection test, through adaptively adjusting the λ parameter of the Page-Hinkley test.

Notations
In this section, $n_{i,t}$ and $\hat{\mu}_{i,t}$ respectively denote the estimation effort (initially, the number of times the i-th arm has been selected) and the average reward associated to the i-th arm at time step t; subscript t is omitted when clear from the context. The process memory, made of the $n_{i,t}$ and $\hat{\mu}_{i,t}$ for $i = 1 \dots K$, dictates the selection of the next option through the UCBT algorithm (Appendix 1).

3.1 $\gamma-$ Restart

Let T denote the current time step where the change-point detection occurs, and let $T - n_C$ denote the time step where the previous change-point detection occurred (set to 0 by default). Window time $[T - n_C, T]$ is referred to as the last episode of the process.

Smooth restart proceeds by discounting the estimation effort associated to every bandit arm. Formally, the γ-restart procedure multiplies $n_{i,T}$ by the discount γ factor ($0 < \gamma < 1$) for $i = 1 \dots K$. The average reward $\hat{\mu}_{i,T}$ is kept unchanged. In further time steps, parameters $n_{i,T+\ell}$ and $\hat{\mu}_{i,T+\ell}$ are updated as before (Appendix 1).

3.2 Meta-Bandit

The Meta-Bandit procedure models the choice of increasing exploration or discarding the change-point detection as another bandit problem. Precisely, the Meta-Bandit is concerned with selecting one among two Bandits: the *Old* Bandit considers that the change-point detection is a false alarm; it implements the UCBT algorithm based on the current process memory ($n^{\mathrm{o}}_{i,T} = n_{i,T}$; $\hat{\mu}^{\mathrm{o}}_{i,T} = \hat{\mu}_{i,T}$); the *New* Bandit considers instead that the change-point detection is correct; it accordingly implements the UCBT algorithm based on a void memory at time step T ($n^{\mathrm{n}}_{i,T} = \hat{\mu}^{\mathrm{n}}_{i,T} = 0$).

The Meta-Bandit memory involves the number of times each Bandit has been selected, respectively noted n^{n} and n^{o}, and the associated average reward $\hat{\mu}^{\mathrm{n}}$ and $\hat{\mu}^{\mathrm{o}}$, all set to 0 at time T. In every further time step $T + \ell, \ell \geq 1$, the Meta-Bandit uses UCBT to select one among the *New* and *Old* Bandits. The selected Bandit uses its own memory to select some i-th option and it accordingly gets some reward r_i. Reward r_i is used to update three parameters: i/ the reward associated to the selected Bandit; ii/ the reward associated to the i-th option for the *New* Bandit; iii/ the reward associated to the i-th option for the *Old* Bandit. Further, the Meta-Bandit increments the number of selections associated to the selected Bandit[1].

The Meta-Bandit thus gradually estimates the rewards associated to the *New* and *Old* Bandits. After MT time steps (set to 1000 in all reported experiments), the Bandit with the lowest reward is killed; the other Bandit takes in the control of the process, and the Meta-Bandit is killed too.

3.3 Adaptive change-point detection through adjusting λ

Note that one can always determine *a posteriori* whether the last change-point detection was a false alarm. In the smooth restart case, the false alarm is detected as: the best option did not change between the previous and the current episode. In the Meta-Bandit case, the false alarm amounts to: the *Old* Bandit wins.

Accordingly, the λ parameter is adjusted as follows, where $\Delta\mu$ is the difference between the reward of the best current option and the second best. Parameters α and β are experimentally adjusted.

$$\lambda := \lambda \times e$$
$$e = \begin{cases} (1 - \alpha\Delta\mu) & if\ true\ alarm \\ (1 + \beta\Delta\mu) & if\ false\ alarm \end{cases}$$

[1] In the rare cases where both Bandits would select the same option, the Meta-Bandit increments both n^{n} and n^{o}.

4 Empirical validation

Adapt-EvE involves six parameters, detailed in Table 2 together with the empirically optimal values in the context of validation, that of the EvE Pascal Challenge (Hussain et al., 2006). The sensitivity analysis is in Appendix 2.

Parameter	Role	Adjustment	Optimal value
δ	change-point detection	manual	$5\,10^{-3}$
λ	change-point detection	adaptive	100
γ	in γ-restart only	manual	.95
MT	in Meta-Bandit only	fixed	1000
α	for λ adjustment	manual	10^{-4}
β	for λ adjustment	manual	10^{-2}

Table 2: Parameters of *Adapt-EvE*

The experimental results of *Adapt-EvE* compared to the baseline UCBT (Auer et al., 2002) and the discounted UCBT proposed by L. Kocsys (2006), are reported in Table 3. For each algorithm and visitor, the regret (in thousands) is averaged over 100 independent runs.

	Baseline Algs		γ-restart		Meta-Bandit	
			Adaptive Detection		Adaptive Detection	
	UCBT	UCBT + discount	No	Yes	No	Yes
Frequent Swap	32.6 ± 0.2	19.1 ± 0.1	12.1 ± 0.1	16.6 ± 0.6	17.7 ± 1	14 ± 0.5
Long Gaussians	53.1 ± 4	7.1 ± 0.2	7.4 ± 0.4	5.6 ± 0.3	5.9 ± 0.3	6.5 ± 0.4
Daily Variation	60.2 ± 1.4	6.9 ± 0.1	6.9 ± 0.6	5 ± 0.2	13.4 ± 0.9	6.5 ± 0.3
Weekly Variation	62.2 ± 0.7	10.2 ± 0.4	7.3 ± 0.2	5.5 ± 0.1	5.9 ± 1.7	6 ± 0.2
Weekly Close Var.	21.6 ± 0.5	9.0 ± 0.2	6.6 ± 0.2	6.1 ± 0.1	6.1 ± 0.2	6.5 ± 0.2
Constant	0.4 ± 0.02	7.3 ± 0.05	0.4 ± 0.02	3.1 ± 0.1	0.4 ± 0.02	1.5 ± 0.1
Regret	230 ± 4.5	59.7 ± 0.5	40.9 ± 0.8	42.2 ± 0.7	49.5 ± 1.3	41.6 ± 0.8

Table 3: *Adapt-EvE*: Regret (in thousands) after 10^6 steps on every visitor with confidence interval at 95%, using the best parameterization for each variant, averaged over 100 runs.

The γ-restart strategy appears to be the best one in the context of the EvE Challenge, provided that parameters γ and λ are carefully adjusted. Complementary experiments and the sensitivity analysis (Appendix 2) shows that the adaptive adjustment of λ does not work well in the context of the γ-restart; further, the performances strongly depend on the values of γ and λ.
With no adaptation of the change-point detection, the Meta-Bandit is outperformed by the γ-restart although its performances are less sensitive to the δ and λ parameters. Interestingly, the Meta-Bandit enables an efficient adaptation of the λ parameter; this adaptation leads Meta-Bandit to catch up γ-restart.

5 Conclusion and Perspectives

The *Adapt-EvE* algorithm was devised for online learning in abruptly changing environments. Its good performances rely first on the use of an efficient change-point detection test, and secondly on specific (alternative) procedures devised for controlling the extra-exploration related to change-point detection, the γ-restart and Meta-Bandit. The theoretical study of these procedures is undergoing.

Further work will be concerned with incorporating prior or posterior knowledge about the periodicity of the dynamic environments. Another perspective is concerned with extending *Adapt-EvE* to Many-armed bandit problems.

References

Auer, P., Cesa-Bianchi, N., & Fischer, P. (2002). Finite time analysis of the multiarmed bandit problem. *Machine Learning*, *47*, 235–256.

Cesa-Bianchi, N., & Lugosi, G. (2006). *Prediction, learning, and games*. Cambridge University Press.

Hinkley, D. (1969). Inference about the change point in a sequence of random variables. *Biometrika*, *57*, 1–17.

Hinkley, D. (1970). Inference about the change point from cumulative sum-tests. *Biometrika*, *58*, 509–523.

Hinkley, D. (1971). Inference in two-phase regression. *Journal of the American Statistical Association*, *66*, 736–743.

Hussain, Z., Auer, P., Cesa-Bianchi, N., Newnham, L., & Shawe-Taylor, J. (2006). Exploration vs. exploitation pascal challenge. http://www.pascal-network.org/Challenges/EEC.

Kifer, D., Ben-David, S., & Gehrke, J. (2004). Detecting change in data streams. *Proc. VLDB'04* (pp. 180–191). Morgan Kaufmann.

Lorden, G. (1971). Procedures for reacting to a change in distribution. *Ann. Math. Stat.*, *42*, 1897–1908.

Page, E. (1954). Continuous inspection schemes. *Biometrika*, *41*, 100–115.

Appendix 1: UCBT

In order for this paper to be self contained, this section briefly describes the UCB-Tuned (UCBT) algorithm proposed by (Auer et al., 2002) for the multi-armed bandit problem, and incorporated in *Adapt-EvE*.

Formally, let K denotes the number of options (bandit arms). The (unknown) reward associated to the i-th option is noted μ_i. Let $\hat{\mu}_i$ denote the average reward collected by the gambler for the i-th option; let n_i denote the estimation effort spent on the i-th option[2]. Let $N = \sum_{i=1}^{K} n_i$ denote the total estimation effort.

The regret $\mathcal{L}(N)$ after N estimation effort is the loss incurred by the gambler compared to the best possible strategy, i.e. investing N effort on the best option (with reward $\hat{\mu}^* = max\{\hat{\mu}_i,\ i = 1 \dots K\}$).

$$\mathcal{L}(N) = \sum_i n_i \times (\hat{\mu}^* - \hat{\mu}_i)$$

Assuming that rewards are bounded, the UCB1 algorithm ensures that the expected loss is bounded logarithmically with the estimation effort N (Auer et al., 2002), assuming that the machines are independent.

Adapt-EvE uses an algorithmic variant of UCB referred to as UCB-Tuned (UCBT) for its better empirical results (Auer et al., 2002). Let $V_i(n_i)$ denote an upper bound on the variance of the reward of the i-th machine, then equation (1) is replaced by

$$i = Argmax\{\hat{\mu}_j + \sqrt{\frac{2 \log\ N}{n_j} \times min(\frac{1}{4}, V_j(n_j))}\}$$

The above selection rule tends to decrease the exploration strength, except possibly for options with high variance.

[2]Originally, n_i is the number of times the i-th option has been selected. However, considering n_i as the estimation effort spent on the i-th option makes more sense in the context of the γ-restart strategy (section 3.1).

Initialization: For $i = 1 \dots K$, $n_i = \hat{\mu}_i = 0$. $N = 0$
Repeat
  if $n_i = 0$ for some $i \in 1 \dots K$ play i
  else play $i = \text{argmax } \{\hat{\mu}_j + \sqrt{\frac{2 \log N}{n_j}},\ j = 1 \dots K\}$
  let r be the associated reward
  Update n_i and $\hat{\mu}_i$
    $\hat{\mu}_i := \frac{1}{n_i+1} \times (n_i \hat{\mu}_i + r)$
    $n_i := n_i + 1$
  $N := N + 1$

Table 4: Algorithm UCB1

Appendix 2. Sensitivity study

The sensitivity of *Adapt-EvE* with no adaptive change detection, with respect to parameters δ and λ (controlling the false alarm rate), and γ (controlling the γ-restart), is respectively shown in Table 5.(a), (b) and (c).

δ	γ-restart	Meta-Bandit
10^{-4}	43.8 ± 2.8	48.8 ± 1.5
$5\ 10^{-4}$	42.9 ± 1.9	49.2 ± 2.4
10^{-3}	42.3 ± 1.7	50.9 ± 2.1
$5\ 10^{-3}$	39.8 ± 2	47.7 ± 4.9
10^{-2}	44.2 ± 5.3	47.9 ± 2.4

(a) *Adapt-EvE* sensitivity wrt δ

λ	γ-restart	Meta-Bandit
80.	41.9 ± 2	51.5 ± 3.6
100	39.8 ± 2	47.7 ± 4.9
150	45.7 ± 6	52.1 ± 4
250	51 ± 5	52.6 ± 5.8

(b) *Adapt-EvE* sensitivity wrt λ

γ	*Adapt-EvE* with γ-restart
0.	46 ± 2.1
0.85	41.7 ± 2.4
0.9	40.5 ± 1.9
0.95	39.8 ± 2
0.99	41.7 ± 2.5
0.999	79.9 ± 3.5

(c) *Adapt-EvE*- γ-restart sensitivity wrt γ

Table 5: Sensitivity analysis of *Adapt-EvE* wrt parameters δ, λ and γ (95% confidence interval), with NO adaptive adjustment of the λ parameter

The online regrets of all *Adapt-EvE* variants and the baseline algorithms are reported in Fig. 1.

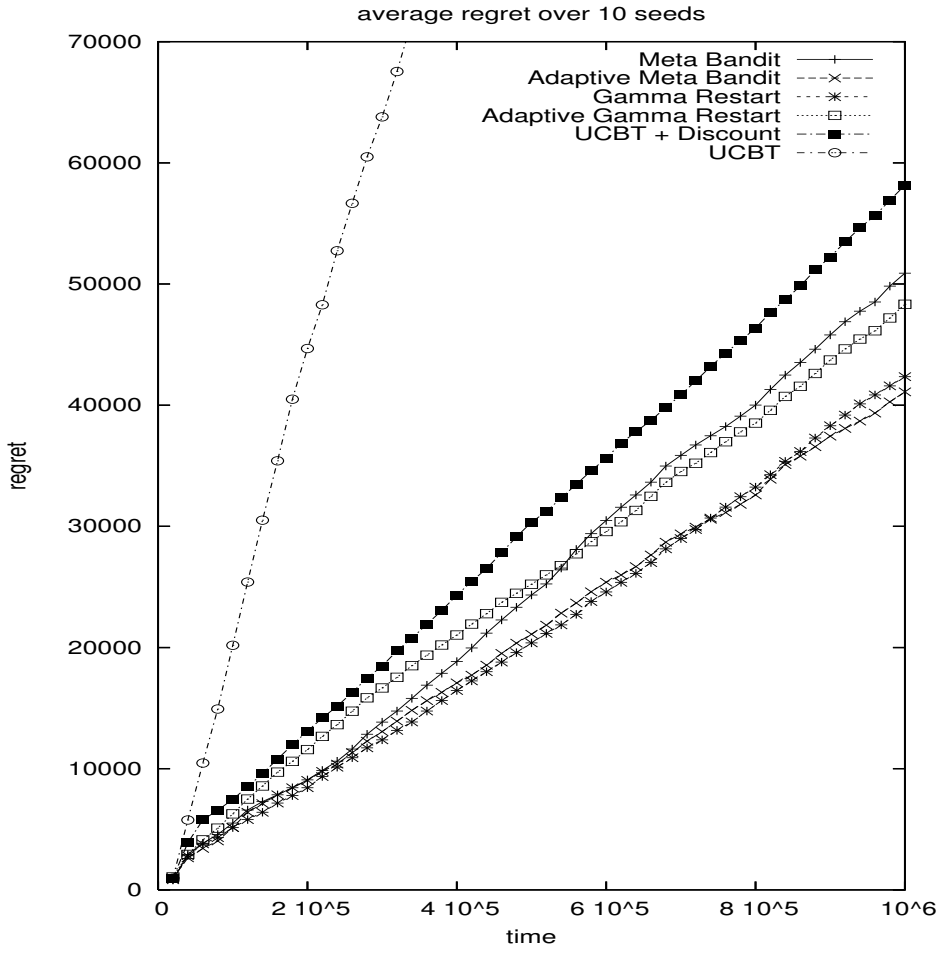

Figure 1: *Adapt-EvE*: Online regret averaged over all visitors $\times$ 10 runs, compared to baseline average regret