Un Algorithme pour le Problème des Bandits Manchots avec Stationnarité par Parties

Robin Allesiardo[1,2] et Raphaël Féraud[1]

[1]Orange Labs, 2 av. Pierre Marzin, 22300 Lannion
[2]TAO - INRIA, LRI, Université Paris-Sud, CNRS, 91405 Orsay

Résumé

Dans le problème des bandits manchots, un joueur possède le choix entre plusieurs bras possédant des espérances de gain différentes. Son but est de maximiser la récompense obtenue après T essais. Il doit alors explorer pour estimer les récompenses de chaque machine tout en exploitant le bras qu'il estime le meilleur. C'est le dilemme exploration/exploitation. Dans le cas des bandits *adverses*, la séquence de récompenses d'une machine est choisie à l'avance par un adversaire. En pratique, la notion de meilleur bras sur le jeu complet est trop restrictive pour des applications comme l'optimisation publicitaire où la meilleure publicité peut changer au cours du temps. Dans ce papier, nous considérons une variante du problème *adverse*, où le jeu est divisé en un nombre inconnu de périodes à l'intérieur desquelles les récompenses sont tirées dans des distributions stochastiques. Entre chaque période, le meilleur bras peut changer. Nous présentons un algorithme utilisant l'exploration constante d'EXP3 pour détecter les changements de meilleur bras. Son analyse montre, que sur un jeu divisé en N segments où le meilleur bras change, l'algorithme proposé possède un regret en $O(N\sqrt{T \log T})$.

1 Introduction

Le problème des bandits manchots est un jeu répétitif où, à chaque tour, un joueur choisit l'un des K bras (ou actions) puis reçoit la récompense associée à l'action choisie. Pour trouver l'action la plus profitable, le joueur doit explorer les différents choix possibles mais doit aussi exploiter l'action qu'il estime la meilleure de manière à maximiser sa récompense cumulée. La qualité de la stratégie du joueur est mesurée en terme de regret, qui est la différence entre la récompense cumulée du joueur et celle qu'aurait obtenue une stratégie supposée optimale comme ≪ *jouer uniquement le bras possédant la plus haute espérance de récompense* ≫.

Dans le cas *stochastique*, les récompenses de chaque bras sont tirées de manière indépendante dans une distribution associée à chaque bras. Le plus ancien algorithme, le Thompson Sampling [Tho33], est basé sur des idées Bayésiennes. À chaque pas de temps, une action est choisie en échantillonnant les récompenses a posteriori estimées pour chaque bras. Cette politique atteint, en espérance, un regret cumulé logarithmique [AG12] et est optimal asymptotiquement [KKM12]. L'algorithme UCB [ACBF02] calcule un *intervalle de confiance supérieur* pour chaque bras et joue le bras possédant la plus haute estimation. L'analyse de cet algorithme montre que son regret cumulé possède une borne supérieure logarithmique, ce qui est optimal.

Dans le cas *adverse*, les récompenses sont choisies à l'avance par un adversaire. Cette formulation a été longuement étudiée par Auer et al [ACBFS02] ainsi que par Cesa-Bianchi et Lugosi [CBL06]. Les algorithmes adverses les plus populaires proviennent de la famille EXP3 [ACBFS02]. EXP3 atteint un regret cumulé en $O(\sqrt{T})$, ce qui est optimal dans le cas adverse. Le problème des bandits manchots avec stationnarité par parties considère que des points de rupture sont choisis en avance par un adversaire [GM11]. Entre ces points, les récompenses sont tirées depuis des distributions stochastiques. Cet énoncé est approprié pour des applications comme l'optimisation publicitaire, où les publicités disponibles peuvent changer au cours du temps.

Résumé de la contribution

Pour être capable d'appréhender les changements de meilleur bras, notre contribution combine l'algorithme de bandit adverse EXP3 avec un détecteur de rupture. Cet algorithme, appelé EXP3.R, utilise l'exploration constante d' EXP3 pour maintenir des estimateurs non biaisés

de la récompense moyenne de chaque bras. Quand un changement de meilleur bras est détecté, les poids d'EXP3 sont réinitialisés. L'analyse de notre algorithme montre que son regret cumulé est borné par $O(N\sqrt{T \log T})$.

2 Travaux précédents

Le principal point faible d'EXP3 provient du fait qu'il est construit pour trouver un unique meilleur bras sur le jeu entier. Cette notion d'unique meilleur bras est trop restrictive pour des applications comme l'optimisation publicitaire. Pour s'affranchir de cette restriction, EXP3.S [ACBFS02] utilise une méthode de régularisation sur les estimateurs des récompenses pour oublier le passé et faciliter les changements de bras. L'analyse d'EXP3.S le compare à une politique optimale capable de jouer N bras différents au cours du jeu. Face à elle, le regret d'EXP3.S est borné par $O(\sqrt{NT \log T})$. Dans DISCOUNTEDUCB [KS06], un facteur de réduction est utilisé sur les récompenses d'UCB pour l'adapter à la non-stationnarité. Une autre variante utilise une fenêtre sur les estimateurs d' UCB. Ce deux adaptations ont été analysées dans [GM11] et sont bornées par $O(\sqrt{MT \log T})$, où M est le nombre de changements de moyenne. La borne inférieure en $\Omega(\sqrt{T})$ pour les bandits manchot avec stationnarité par parties a été démontré dans [GM11] et les précédents algorithmes l'atteignent à un facteur $\sqrt{\log T}$ près. Une autre approche proposée pour les ≪ *jeux à gratter* ≫ réinitialise EXP3 à chaque fois qu'un bras apparait ou disparait [FU13]. Pour utiliser cette approche sur notre problème, il est nécessaire de savoir quand les changements se produisent.

La détection du changement a été étudiée [HPC12] pour différentes applications comme la détection du spam, de la fraude, la météorologie ou bien la finance. Les concepts surveillés dépendent de l'application. Par exemple, dans la classification en ligne, les variations de performances du modèle sont souvent utilisées pour détecter les dérives de concept : dans [GMCR04], les auteurs considèrent que l'erreur en classification est une variable aléatoire de Bernoulli et dans [Las02], des exemples sont collectés à différents intervalles de temps et utilisés comme ensemble d'apprentissage et de validation. Pour les problèmes de bandits à information partielle, dans META-EVE [HBG+06], la récompense moyenne de la meilleure action estimée est surveillée. Une statistique de Page-Hinkley est utilisée pour tester si la série de récompense provient d'une unique loi statistique. Le point faible de cette approche est de ne pas gérer le cas où une action sous-optimale devient optimale sans que la récompense du meilleur bras ne change. Des intervalles de confiance sont utilisés dans [YM09] pour détecter les changements de récompense moyenne. Cet algorithme possède une borne en $O(N \log T)$ où $N-1$ est le nombre de changements durant le jeu. Pour arriver à une telle borne, l'algorithme utilise des ≪ *informations cachées* ≫ acquises sans impact sur le regret. La divergence de Kullback-Leiber peut aussi être utilisée comme détecteur [YYC13, BLB10].

3 Politique optimale et Regret cumulé

Définition. De manière similaire à SW-UCB [GM11], nous définition le problème des bandits manchots avec stationnarité par partie par un ensemble de K actions, où $1 \leq k \leq K$ est l'index de chaque action, et une séquence de vecteurs récompense $\mathbf{x}(t) = (x_1(t), ..., x_k(t))$ de taille T où $x_k(t) \in \{0, 1\}$. Chaque récompense $x_k(t)$ est tirée depuis une distribution de Bernoulli de moyenne $\mu^k(t)$. Les $M-1$ pas de temps où $\exists k,\ \mu(t)^k \neq \mu^k(t+1)$ sont appelés points de rupture. Un adversaire choisit les pas de temps où les ruptures se produisent et tire ensuite les séquences de récompenses. L'action jouée au temps t est noté $k(t)$. Le but d'un algorithme A est de maximiser le gain cumulé à l'horizon de temps T défini par :

$$G_A = \sum_{t=1}^{T} x_{k(t)}(t)\,. \tag{1}$$

La politique optimale. La séquence de vecteurs récompense est divisée en $N \leq M$ séquences appelées segments. S est l'index du segment incluant les tours $[T_S, T_{S+1}[$. Un segment S commence quand $\max_k \mu^k(T_S) \neq \max_k \mu^k(T_S - 1)$. Le bras optimal sur le segment S est noté k_S^* avec :

$$k_S^* = \arg\max_k \sum_{t=T_S}^{T_{S+1}-1} \mu^k(t)\,. \tag{2}$$

Le gain de la politique optimale est noté G^* et défini par :

$$G^* = \sum_{S=1}^{N} \sum_{t=T_S}^{T_{S+1}-1} x_{k_S^*}(t)\,. \tag{3}$$

Le regret de l'algorithme A face à la politique optimale est :

$$R(A) = G^* - G_A\,. \tag{4}$$

Notez que

$$G^* \leq \sum_{S=1}^{N} \max_k \sum_{t=T_S}^{T_{S+1}-1} x_k(t)\,, \tag{5}$$

c'est à dire qu'après le tirage par l'adversaire, la séquence de récompenses d'un bras sous-optimal peut être supérieure à la séquence du bras avec la plus haute récompense moyenne.

4 Algorithme

Tandis-que les autres algorithmes ont une approche passive consistant à oublier le passé [ACBFS02, KS06, GM11], nous proposons une stratégie active consistant à réinitialiser les estimateurs de l'algorithme quand un changement de meilleure action est détecté. Premièrement, nous décrirons l'algorithme adverse EXP3 [ACBFS02] qui sera utilisé par l'algorithme proposé EXP3.R entre chaque détection. Nous décrirons ensuite le détecteur de dérive de concept utilisé pour détecter les changements de meilleur bras. Pour finir, nous combinerons les deux pour obtenir l'algorithme EXP3.R.

Algorithm 1 EXP3

Le paramètre $\gamma \in [0,1]$ contrôle l'exploration. La probabilité de choisir k au tour t est :

$$p_k(t) = (1-\gamma)\frac{w_k(t)}{\sum_{i=1}^{k} w_i(t)} + \frac{\gamma}{K}\,. \tag{6}$$

Les poids $w_k(t)$ de chaque action k sont :

$$w_k(t) = \exp\left(\frac{\gamma}{K}\sum_{j=t_r}^{t}\frac{x_k(j)}{p_k(j)}[\![k=k(j)]\!]\right)\,, \tag{7}$$

où t_r est le pas de temps où l'algorithme a été initialisé.

L'algorithme EXP3 (voir Algorithme 1) minimise le regret face au meilleur bras en utilisant un estimateur non biaisé des récompenses cumulées au temps t pour calculer les probabilités de choisir chaque action. Même si cette stratégie peut être vu comme optimale dans un cas adverse, dans beaucoup d'applications pratiques, la non-stationnarité à l'intérieur d'une période de temps est faible et est uniquement remarquable sur le long terme. Si un bras est performant sur une période de longue taille mais devient très mauvais ensuite, l'algorithme EXP3 peut nécessiter un

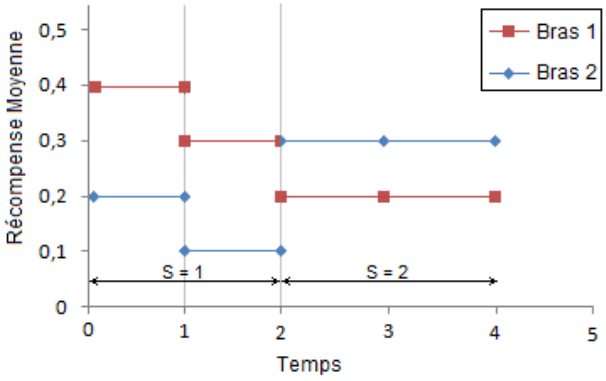

FIGURE 1 – Ce jeu comporte trois changements de moyennes mais la politique optimale ne changerait de bras qu'une seule fois. Ici, $M = 3$ et $N = 2$.

nombre d'itérations égal à la taille de la première période avant de changer de bras majoritairement joué. Couplé avec un détecteur de changement, l'algorithme EXP3 a plusieurs avantages. Premièrement, dans un environnement non-stationnaire, une exploration constante est nécessaire pour détecter les changement. Cette exploration est donnée naturellement par l'algorithme. Deuxièmement, le nombre de changements de moyennes est supérieur au nombre de changements de meilleurs bras ($M \geq N$) (voir Figure 1). Cela signifie que le nombre de ré-initialisations requises par EXP3 est plus petit que celui nécessaire à un algorithme stochastique comme UCB. Troisièmement, EXP3 est robuste face aux non détections ou aux non-stationnarités locales.

Le détecteur de dérive (voir Algorithme 2) utilise des intervalles de confiance pour estimer les espérances de récompenses sur la période de temps précédente. La distribution de probabilité des actions dans EXP3 est une mixture entre une distribution de Gibbs et une distribution uniforme. Nous appelons γ-observation, une observation sélectionnée via la distribution uniforme. Les paramètres γ, H et δ définissent un nombre minimal de γ-observations nécessaires à l'utilisation d'un détecteur d'une précision ϵ avec une probabilité d'au moins $1-\delta$. Ces paramètres seront fixés plus tard dans l'analyse (voir Corollaire 1) et la validité du test est prouvée dans le Lemme 1. Nous notons $\hat{\mu}^k(I)$ la moyenne empirique des récompense obtenues par le bras k sur l'intervalle I en utilisant uniquement les γ-observations et $\Gamma_{\min}(I)$ le plus petit nombre de γ-observations parmi tous les bras sur l'intervalle I. Le détecteur est appelé uniquement quand $\Gamma_{\min}(I) \geq \frac{\gamma H}{K}$. Celui-ci lève une détection quand l'action $k_{\max}$, qui est celle possédant la plus haute probabilité $p_k(t)$ (voir Algorithme 1), est éliminée par une

autre sur l'intervalle courant.

Algorithm 2 DetectionDeDerive(I)

Paramètres : Intervalle courant I
$k_{\max} = \arg\max_k p_k(t)$
$\epsilon = \sqrt{\frac{K \log(\frac{1}{\delta})}{2\gamma H}}$
return $\left[\!\left[\exists k, \quad \hat{\mu}^k(I) - \hat{\mu}^{k_{\max}}(I) \geq 2\epsilon\right]\!\right]$

L'algorithme EXP3.R est obtenu en combinant EXP3 et le détecteur de dérive. Dans un premier temps une instance d' EXP3 est initialisée et utilisée pour sélectionner les actions. Des γ-observation sont alors collectée jusqu'à ce que $\Gamma_{\min}(I) \geq \frac{\gamma H}{K}$. Lorsque ce compte est atteind, le test de détection est exécuté. Si, dans l'intervalle correspondant, la moyenne empirique d'un bras est supérieure de 2ϵ à meilleure action actuelle alors une détection est levée. Dans ce cas, les poids de l'algorithme EXP3 sont réinitialisés. Commence alors un nouvel intervalle de collecte en préparation du prochain test. Ces étapes sont répétées jusqu'à la fin du jeu (voir Algorithme 3).

Algorithm 3 EXP3 avec Ré-initialisation

Paramètres : Réels δ, γ et entier H
$I = 1$
for each $t = 1, ..., T$ **do**
  Appeler EXP3 sur le pas de temps t
  if $\Gamma_{\min}(I) \geq \frac{\gamma H}{K}$ **then**
    if $DetectionDeDerive(I)$ **then**
      Réinitialiser EXP3
    end if
    $I = I + 1$
  end if
end for

Avec une précision ϵ, seules les différences supérieures à 4ϵ sont détectées avec une haute probabilité. De la même manière que [YM09] nous utilisons l'hypothèse 1 pour s'assurer que tout les changements de meilleur bras sont détectés avec une haute probabilité.

Hypothèse 1. *Durant chacune des M périodes stationnaires, la différence entre la récompense moyenne du bras optimal et celle de n'importe quel autre bras est au moins de 4ϵ avec*

$$\epsilon = \sqrt{\frac{K \log(\frac{1}{\delta})}{2\gamma H}}\,. \tag{8}$$

4.1 Analyse

Dans cette section nous analysons le détecteur de rupture et bornons l'espérance du regret de l'algorithme EXP3.R.

Le lemme 1 garantit que quand l'hypothèse 1 est valide et que l'intervalle I est inclus dans l'intervalle S alors, avec une haute probabilité, une détection sera levée si et seulement si le bras optimal k_S^* élimine un bras sous-optimal.

Lemme 1. *Quand l'hypothèse 1 est valide et que $I \subseteq S$ alors, avec une probabilité d'au moins $1 - 2\delta$, pour tout $k \neq k_S^*$:*

$$\hat{\mu}^{k_S^*}(I) - \hat{\mu}^k(I) \geq 2\sqrt{\frac{K \log(\frac{1}{\delta})}{2\gamma H}} \Leftrightarrow \mu^{k_S^*}(I) \geq \mu^k(I)\,. \tag{9}$$

Démonstration. Nous justifions notre test de détection en considérant les γ-observations comme des tirages sans remplacement dans une urne. Plus précisément, quand toutes les observations nécessaires sont collectées, la procédure de détection est lancée. Durant l'intervalle de collecte, les récompenses sont tirées depuis $1 \leq m \leq M$ différentes distributions de moyenne $\mu_0^k(I), ..., \mu_m^k(I)$. Nous appelons t_i le pas de temps où la réponse moyenne commence à être $\mu_i^k(I)$ et t_{m+1} le pas de temps où le détecteur est appelé. A posteriori, chaque $x_k(t)$ a une probabilité $(t_{i+1} - t_i)/(t_{m+1} - t_0)$ d'avoir été tiré depuis la distribution de moyenne $\mu_i^k(I)$. L'espérance moyenne de l'urne contenant les récompenses correspondant à l'action k est :

$$\mu^k(I) = \sum_{i=1}^{m} \frac{t_{i+1} - t_i}{t_{m+1} - t_0} \mu_i^k(I)\,. \tag{10}$$

A chaque pas de temps, par hypothèse, la récompense moyenne du meilleur bras est supérieure à celle de n'importe quel autre bras d'au moins 4ϵ . Par conséquent, la différence entre la récompense moyenne de l'urne du bras optimal k^* et celle d'un autre bras k est au moins de 4ϵ si le meilleur bras ne change pas au cours de l'intervalle.

$$\mu^k(I) \leq \sum_{i=1}^{m} \frac{t_{i+1} - t_i}{t_{m+1} - t_0} (\mu_i^{k_S^*} - 4\epsilon) \leq \mu^{k_S^*}(I) - 4\epsilon\,. \tag{11}$$

Les arguments suivant montrent l'équivalence entre une détection et l'optimalité de k_S^* avec une forte probabilité.

En utilisant l'inégalité de Serfling [Ser], nous avons :

$$P(\hat{\mu}^k(I) + \epsilon \geq \mu^k(I))) \leq e^{\frac{-2n\epsilon^2}{1 - \frac{n-1}{U}}} \leq e^{-2n\epsilon^2} = \delta\,, \tag{12}$$

où $n = \frac{\gamma H}{K}$ est le nombre d'observations et U la taille de l'urne. Nous utilisons δ pour décrire la quantité $P(\hat{\mu}^k(I) + \epsilon \geq \mu^k(I)))$. Nous avons, avec une probabilité d'au moins

$1-2\delta$, $\hat{\mu}^k(I)+\epsilon \geq \mu^k(I)$ et $\hat{\mu}^{k^*_S}(I)-\epsilon \leq \mu^{k^*_S}(I)$. Par conséquent, si $\hat{\mu}^{k^*_S}(I)-\hat{\mu}^k(I) \geq 2\epsilon$ alors, avec une haute probabilité, $\mu^{k^*}(I) \geq \mu^k(I)$. Le détecteur utilise l'intervalle de confiance supérieur pour l'actuel meilleur bras estimé et l'inférieur pour le bras candidat. Lorsque les deux intervalles sont valides, dans le pire des cas, chacun des estimateurs sont éloignés de la vraie moyenne de ϵ. Le détecteur ajoute ensuite ϵ à chaque estimateurs. L'inéquation (11) assure que tout les changements de meilleur bras sont détectés. □

Le théorème 1 borne l'espérance du regret cumulé d'EXP3.R.

Théorème 1. *Pour tout $K > 0$, $0 < \gamma \leq 1$, $0 \leq \delta < \frac{1}{2}$, $H \geq K$ et $N \geq 1$, quand l'hypothèse 1 est valide, l'espérance du regret d'*EXP3.R *vérifie*

$$G^* - \boldsymbol{E}[G_{\textit{EXP3.R}}] \leq (e-1)\gamma T + \frac{\left(N-1+\frac{K\delta T}{H}+K\delta\right)K\log(K)}{\gamma} + (N-1)HK\left(\frac{1}{1-2\delta}+1\right). \quad (13)$$

Démonstration. Premièrement, nous obtenons la structure principale de la borne. Dans ce qui suit, $L(T)$ décrit l'espérance du nombre d'intervalle après un changement de meilleure action avant sa détection et $F(T)$ l'espérance du nombre de fausses détections jusqu'alors. En utilisant les même arguments que [YM09], nous déduisons la forme de la borne depuis la borne classique d'EXP3. Il y a $N-1$ changements de meilleur bras, ainsi le nombre de ré-initialisations à l'horizon T est borné supérieurement par $N-1+F(T)$. Le regret d'EXP3 sur ces périodes est $(e-1)\gamma T+\frac{K\log K}{\gamma}$ [ACBFS02]. Alors que notre politique optimale joue le bras ayant la plus haute récompense moyenne, la politique d'EXP3 joue le bras ayant la plus haute récompense cumulée sur la période $[T_S, T_{S+1})$. Comme

$$\sum_{t=T_S}^{T_{S+1}-1} x_{k^*_S}(t) \leq \max_k \sum_{t=T_S}^{T_{S+1}-1} x_k(t), \quad (14)$$

le gain de notre politique optimale sur une période est borné supérieurement par le gain de la politique optimale d'EXP3. En sommant sur chaque période nous obtenons $(e-1)\gamma T+\frac{(N-1+F(T))K\log K}{\gamma}$.
Le regret comprend aussi le délais entre le changement de meilleur bras et sa détection. Pour évaluer l'espérance de la taille de l'intervalle séparant chaque appel du test, nous considérons un algorithme fictif collectant uniquement les observations d'un bras avant de passer au suivant jusqu'à avoir collecté toutes les observations. Les γ-observations sont tirées avec une probabilité $\frac{\gamma}{K}$ et $\frac{\gamma H}{K}$ observations sont nécessaires par action. L'espérance du nombre d'échecs avant de réussir à collecter $\frac{\gamma H}{K}$ γ-observations suit une loi binomiale négative d'espérance

$$\frac{\gamma H}{K}(1-\frac{\gamma}{K})\frac{K}{\gamma} = H - \frac{\gamma H}{K}. \quad (15)$$

L'espérance du nombre de tirages à la fin de la collecte est le nombre de succès plus le nombre espéré d'échecs :

$$\frac{\gamma H}{K} + H - \frac{\gamma H}{K} = H. \quad (16)$$

En sommant sur tout les bras, nous obtenons une espérance totale de HK. Étant donné que notre algorithme peut collecter les observations de n'importe quel bras à n'importe quel moment, sur une même séquence de tirages, notre algorithme aura fini sa collecte avant l'algorithme précédemment décrit. Par conséquent, l'espérance du temps entre chaque appel du test est bornée supérieurement par HK et inférieurement par H, l'espérance du temps de collecte pour un bras. Il y a $N-1$ changements de meilleure action et la détection se produit au plus $\lceil L(T)\rceil HK$ tour après le changement. Pour finir, il y a aussi au plus $N-1$ intervalles où les changements de meilleur bras se produisent. Durant ces intervalles, nous n'avons pas de garanties sur le comportement du test. Dans le pire des cas, le test ne détecte pas de changement et le regret instantané est de 1.

$$G^* - \mathbf{E}[G_{\text{EXP3.R}}] \leq (e-1)\gamma T + \frac{(N-1+F(T))K\log K}{\gamma} + (N-1)HK(\lceil L(T)\rceil+1). \quad (17)$$

Nous bornons maintenant $F(T)$ et $L(T)$. Les intervalles de confiance sont valides avec une probabilité d'au moins $1-\delta$ et ils sont utilisés K fois à chaque test. En utilisant l'inégalité de Boole, nous déduisons que $F(T) \leq K\delta(\frac{T}{H}+1)$. $L(T)$ est la première occurrence de l'évènement DÉTECTION après un changement de meilleur bras. Quand un tel changement se produit, le lemme 1 garantit que la détection se produira avec une probabilité d'au moins $1-2\delta$. Nous avons $L(T) \leq \frac{1}{1-2\delta}$.

$$G^* - \mathbf{E}[G_{\text{EXP3.R}}] \leq (e-1)\gamma T + \frac{\left(N-1+\frac{K\delta T}{H}+K\delta\right)K\log K}{\gamma} + (N-1)HK\left(\frac{1}{1-2\delta}+1\right). \quad (18)$$

□

Dans le corollaire 1 nous optimisons les paramètres de la borne obtenue dans le théorème 1.

Corollaire 1. *Pour tout $K \geq 1$, $T \geq 10$, $N \geq 1$ et $C \geq 1$, quand l'hypothèse 1 est vérifiée, l'espérance du regret d'un* EXP3.R *exécuté avec comme paramètres*

$$\delta = \sqrt{\frac{\log T}{KT}}, \gamma = \sqrt{\frac{K \log K \log T}{T}} \text{ et } H = C\sqrt{T \log T} \tag{19}$$

est

$$G^* - \boldsymbol{E}[G_{EXP3.R}] \leq (e-1)\sqrt{TK \log K \log T} + N\sqrt{TK \log K} + (C+1)K\sqrt{T \log K} + 3NCK\sqrt{T \log T}. \tag{20}$$

Suivant C, la précision ϵ est :

$$\epsilon = \sqrt{\frac{1}{2C}} \sqrt{\frac{\log \sqrt{\frac{KT}{\log T}}}{\log T}} \sqrt{\frac{K}{\log K}} \leq \sqrt{\frac{1}{2C}} \sqrt{\frac{K}{\log K}}. \tag{21}$$

Démonstration. Nous fixons les paramètres $\delta = \sqrt{\frac{\log T}{KT}}$ et $H = C\sqrt{T \log T}$ dans le théorème 1

$$G^* - \mathbf{E}[G_{\text{EXP3.R}}] \leq (e-1)\gamma T + \frac{(N - 1 + (C+1)\sqrt{K})K \log K}{\gamma} + 3(N-1)CK\sqrt{T \log T}. \tag{22}$$

Avec $\gamma = \sqrt{\frac{K \log K \log T}{T}}$ nous obtenons :

$$G^* - \mathbf{E}[G_{\text{EXP3.R}}] \leq (e-1)\sqrt{TK \log K \log T} + N\sqrt{TK \log K} + (C+1)K\sqrt{T \log K} + 3NCK\sqrt{T \log T}. \tag{23}$$

□

Discussion

La borne supérieure obtenue est en $O(N\sqrt{T \log T})$ et est éloignée de la borne inférieure d'un facteur $\sqrt{\log T}$. De la même manière, les regrets cumulés de SW-UCB et d'EXP3.S sont bornés respectivement par $O(\sqrt{MT \log T})$ et $O(\sqrt{NT \log T})$. Pour attendre ces bornes, la connaissance du nombre maximal de changement est nécessaire. Dans le cas contraire, les bornes deviennent $O(M\sqrt{T \log T})$ et $O(N\sqrt{T \log T})$. Lors des applications pratiques, les algorithmes sont déployés durant de longues périodes de temps, N est petit et $H \ll T$.

5 Simulations

Nous considérons deux problèmes, durant chacun 10^8 tours avec 20 bras différents. L'algorithme EXP3.R est comparé à cinq autres algorithmes de l'état de l'art (voir Figure 1) et le regret cumulé est une moyenne sur 100 exécutions indépendantes.

Les paramétrages des différents algorithmes sont montrés en table 1. Leur paramétrage est réglé sur le problème 1. Dans META-EVE[HBG+06] δ correspond à l'amplitude des changements ne devant pas lever une alarme et λ contrôle le taux de faux positifs. Durant l'exécution, λ est réglé via un méta-bandit utilisant α pour augmenter λ et β pour le réduire.

Probleme 1. Un bras sur trois possède une récompense moyenne de 0,7, premier bras inclus. Ces bras sont appelés *bras constants*. Un index, définissant le bras optimal, est initialisé sur un *bras constant* et est incrémenté tous les 2×10^7 tours. La récompense moyenne du bras optimal est de 0,8 sauf dans le cas où l'index est sur un *bras constant* ; dans ce cas elle reste de 0,7. Les autres bras possèdent une récompense de 0,2.

Sans surprise, les regrets cumulés de EXP3 et UCB, non adaptés aux changements de bras, sont très élevés. META-EVE est compétitif mais souffre de la constance des récompenses de moyenne 0,7. Quand un tel bras est optimal, étant donné que l'algorithme n'explore pas, les changements ne sont pas remarqués. La variance de cet algorithme est très haute (voir Table 1). Les changements sont détectés lors de certaines exécutions et l'algorithme est alors très performant mais sur d'autres exécutions, les changements passent totalement inaperçus, menant à un regret cumulé très élevé. Les comportements d'EXP3.S et d'EXP3.R sur ce problème sont très similaires mais le facteur de régularisation d'EXP3.S empêche la convergence totale sur durant les phases stationnaires. Finalement, SW-UCB obtient le plus petit regret, usant à son avantage les longues périodes de stationnarité ainsi que sa convergence plus rapide.

Probleme 2. L'index définissant le bras optimal est initialisé sur un bras de manière aléatoire et est incrémenté tout

Algorithme	Paramètre	Valeur	Probleme 1	Probleme 2
EXP3	γ	10^{-2}	$6,1 \times 10^6 \pm 10^5$	$8 \times 10^6 \pm 10^4$
EXP3.S	γ α	10^{-2} 2×10^{-5}	$8,8 \times 10^5 \pm 10^4$	$\mathbf{3.5 \times 10^5 \pm 5 \times 10^4}$
EXP3.R	γ δ H	10^{-2} 10^{-3} 3×10^5	$\mathbf{7,1 \times 10^5 \pm 10^5}$	$\mathbf{2,9 \times 10^5 \pm 10^5}$
UCB	$\emptyset$	$\emptyset$	$4.8 \times 10^6 \pm 10^6$	$7,2 \times 10^6 \pm 10^6$
SW-UCB	W	8×10^5	$\mathbf{2,2 \times 10^5 \pm 10^4}$	$3,5 \times 10^6 \pm 10^5$
Meta-Eve	δ λ α β	10^{-3} 100 $1 + 10^{-2}$ $1 - 10^{-2}$	$1,3 \times 10^6 \pm 8 \times 10^5$	$1,1 \times 10^6 \pm 9,5 \times 10^5$

TABLE 1 – Les différents algorithmes testés, leurs paramétrages ainsi que leurs regrets cumulés sur deux problèmes. Le regret cumulé est moyenné sur 100 exécutions indépendantes.

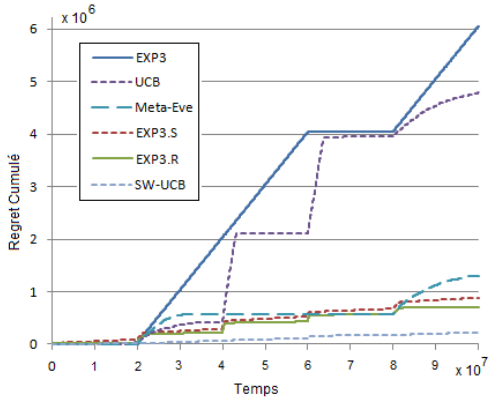

FIGURE 2 – Le regret cumulé en fonction du temps des différents algorithmes sur le Problème 1.

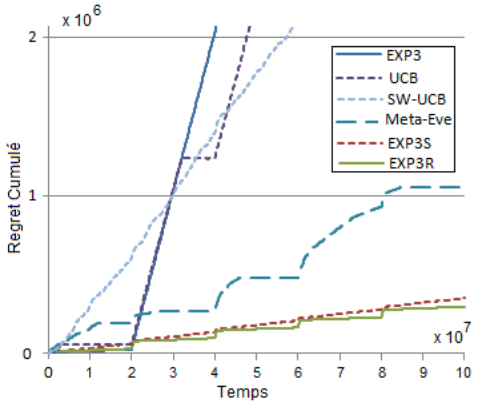

FIGURE 3 – Le regret cumulé en fonction du temps des différents algorithmes sur le Problème 2.

les 2×10^7 tours. La récompense du bras optimal change tout les 10^5 tours, suivant ce cycle : $0,6$, $0,8$ puis $0,5$. Durant les périodes de longueur 2×10^7, le bras optimal ne change pas, même si sa récompense moyenne change. Les bras sous-optimaux possèdent une récompense moyenne inférieure à celle du bras optimal de 0,1 ($0,5$, $0,7$ puis $0,4$).

Comme dans l'expérience précédente, les regrets cumulés d'EXP3 et d' UCB sont très élevés. La variance de META-EVE reste très haute et les changements de moyenne récurrents empêchent le réglage du paramètre λ. EXP3.S reste très proche d'EXP3.R mais est toujours pénalisé par son terme de régularisation. Grâce à sa stratégie active, EXP3.R converge sur les phases où le meilleur bras ne change pas, lui permettant d'obtenir le plus petit regret cumulé. Pour finir, les changements de moyennes empêchent totalement la convergence de SW-UCB.

La non-stationnarité introduit un nouveau dilemme exploration/exploitation. En plus de devoir trouver le meilleur bras, les algorithmes doivent être capables de fonctionner lorsque les distributions de récompense changent. Dans certains cas, un faible regret lors des périodes stationnaire peut devenir un handicap et empêcher la détection des changement, comme dans META-EVE. Ceci est démontré dans [GM11], le problème de bandits manchot possède une borne inférieure en $\sqrt{T}$ si les récompenses ne sont pas stationnaires. Par conséquent, tout algorithme possédant une

plus petite borne supérieure dans le cas stationnaire peut être déjoué par certains types de non-stationnarité.
Les paramètres comme les facteurs de régularisation ou la taille des fenêtres sont difficiles à régler si les non-stationnarités sont apériodiques. Lorsque les changements sont proches, les algorithmes ont besoin d'oublier le passé rapidement mais lorsque les changements sont espacés, une plus grande mémoire aurait été bénéfique. L'avantage de la stratégie active est de permettre à l'algorithme de converger lors des phases stationnaires et d'être réinitialisé uniquement lorsqu'un changement majeur est détecté.

6 Conclusion

L'algorithme proposé, EXP3.R, obtient un regret borné en $O(N\sqrt{T \log T})$. Il est aussi compétitif avec les autres algorithmes de l'état de l'art, les simulations montrant des résultats prometteurs. La nature adverse d' EXP3 le rend robuste à la non-stationnarité et le détecteur accélère les changements de meilleur bras tout en permettant à l'algorithme de converger durant les phases où celui-ci reste identique. Les travaux futurs pourront concerner l'utilisation de cet algorithme comme un méta-bandit [AFB14] pour régler les paramètres et contrôler des ensembles d'algorithmes de bandits contextuels dans un environnement non-stationnaire.

Références

[ACBF02] Peter Auer, Nicolò Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 47(2-3) :235–256, 2002.

[ACBFS02] Peter Auer, Nicolò Cesa-Bianchi, Yoav Freund, and Robert E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM J. Comput.*, 32(1) :48–77, 2002.

[AFB14] Robin Allesiardo, Raphaël Feraud, and Djallel Bouneffouf. A neural networks committee for the contextual bandit problem. In *Neural Information Processing - 21st International Conference, ICONIP 2014, Kuching, Malaysia, November 3-6, 2014. Proceedings, Part I*, pages 374–381, 2014.

[AG12] Shipra Agrawal and Navin Goyal. Analysis of thompson sampling for the multi-armed bandit problem. In *Proceedings of the 25th Annual Conference on Learning Theory (COLT)*, June 2012.

[BLB10] Hanen Borchani, Pedro Larrañaga, and Concha Bielza. Mining concept-drifting data streams containing labeled and unlabeled instances. In Nicolás García-Pedrajas, Francisco Herrera, Colin Fyfe, JoséManuel Benítez, and Moonis Ali, editors, *Trends in Applied Intelligent Systems*, volume 6096 of *Lecture Notes in Computer Science*, pages 531–540. Springer Berlin Heidelberg, 2010.

[CBL06] Nicolo Cesa-Bianchi and Gabor Lugosi. *Prediction, Learning, and Games*. Cambridge University Press, New York, NY, USA, 2006.

[FU13] Raphaël Feraud and Tanguy Urvoy. Exploration and exploitation of scratch games. *Machine Learning*, 92(2-3) :377–401, 2013.

[GM11] Aurélien Garivier and Eric Moulines. On upper-confidence bound policies for non-stationary bandit problems. In *Algorithmic Learning Theory*, pages 174–188, 2011.

[GMCR04] João Gama, Pedro Medas, Gladys Castillo, and Pedro Rodrigues. Learning with drift detection. In AnaL.C. Bazzan and Sofiane Labidi, editors, *Advances in Artificial Intelligence – SBIA 2004*, volume 3171 of *Lecture Notes in Computer Science*, pages 286–295. Springer Berlin Heidelberg, 2004.

[HBG+06] C. Hartland, N. Baskiotis, S. Gelly, O. Teytaud, and M. Sebag. Multi-armed bandit, dynamic environments and meta-bandits. In *Online Trading of Exploration and Exploitation Workshop, NIPS*, Whistler, Canada, December 2006.

[HPC12] T.Ryan Hoens, Robi Polikar, and NiteshV. Chawla. Learning from streaming data with concept drift and imbalance : an overview. *Progress in Artificial Intelligence*, 1(1) :89–101, 2012.

[KKM12] Emilie Kaufmann, Nathaniel Korda, and Rémi Munos. Thompson sampling : An asymptotically optimal finite-time analysis. In NaderH. Bshouty, Gilles Stoltz, Nicolas Vayatis, and Thomas Zeugmann, editors, *Algorithmic Learning Theory*, volume 7568 of *Lecture Notes in Computer Science*, pages 199–213. Springer Berlin Heidelberg, 2012.

[KS06] L. Kocsis and C. Szepesvári. Discounted ucb. In *2nd PASCAL Challenges Workshop*, Venice, Italy, April 2006.

[Las02] Mark Last. Online classification of nonstationary data streams. *Intell. Data Anal.*, 6(2) :129–147, April 2002.

[Ser] R.J. Serfling. Probability inequalities for the sum in sampling without replacement. In *The Annals of Statistics, Vol 2, No.1, pages = 39–48, year = 1974,.*

[Tho33] W.R. Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25 :285–294, 1933.

[YM09] Jia Yuan Yu and Shie Mannor. Piecewise-stationary bandit problems with side observations. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pages 1177–1184, New York, NY, USA, 2009. ACM.

[YYC13] L. Feng Y. Yao and F. Chen. Concept drift visualization. *Journal of Information and Computational Science*, 10(10), 2013.