Projet de mi-parcours – Reinforcement Learning
15 Mars 2021 - Caroline Boudier

I. Contexte et objectif de l'article

Dans cet article de recherche, les auteurs s'intéressent au sujet des attaques adversarielles appliquées à des problèmes de bandits. L'objectif de l'article est de montrer comment des algorithmes classiques en théorie des bandits (UCB et ϵ-greedy) peuvent facilement être induits en erreur par un acteur externe malveillant. Leur idée est de pointer du doigt la fragilité de ces algorithmes afin de pouvoir par la suite proposer des stratégies de défense efficaces notamment dans des contextes critiques tels que les essais médicaux ou la recommandation de contenu. Ce rapport est décomposé en deux grandes parties : une première dédiée à la synthèse de l'article et une seconde consacrée à une implémentation dans le cas ϵ-greedy.

II. Démarche et résultats

a) Introduction du problème et notations

On se place dans un contexte classique de bandits : un agent doit faire une succession de choix parmi plusieurs options lui rapportant chacune un gain aléatoire avec pour but de maximiser sa récompense totale. Plus précisément (et afin d'introduire les notations) l'univers est composé de K bras $\{1,2,\dots K\}$ dont la récompense est modélisée comme une loi gaussienne centrée respectivement en $\mu_1, \mu_2 \dots \mu_K$ et d'écart type σ. On suppose sans perte de généralité que K est un bras sous-optimal (c'est-à-dire que $\mu_K \leq \max_{i=1:K} \mu_i = \mu^*$).

Les agents Bob et Alice s'affrontent alors sur une partie en T tours où chaque tour s'organise comme suit :

Pour t =1,2 ….T
- Bob choisit un bras I_t
- Une récompense associée r_t^0est générée d'après la loi $\mathcal{N}(\mu_{I_t}, \sigma^2)$
- Alice observe I_t et r_t^0 et décide alors d'une attaque α_t (qui peut être nulle)
- Bob obtient une récompense finale $r_t = r_t^0 - \alpha_t$

L'objectif d'Alice est d'induire Bob en erreur en lui faisant croire que K est le meilleur bras, et ce en minimisant la somme totale de ses attaques. En notant $N_i(t)$ le nombre total de fois que le bras i a été tiré à l'étape t on considèrera qu'Alice réussit si $N_K(T) = T - o(T)$ le tout en minimisant le coût d'attaque $\sum_{t=1}^{T} |\alpha_t|$. L'objectif de Bob quant à lui est de minimiser son regret, exprimé comme : $\mathbb{E}\left[\sum_{t=1}^{T}(\mu^* - \mu_{I_t})\right]$ où μ^* est la moyenne théorique associée au meilleur bras.

Ni Bob, ni Alice ne connaissent la vraie distribution des bras, ils n'ont accès qu'à des moyennes empiriques mises à jour au fur et à mesure de la partie. On notera ainsi $\widehat{\mu_i^0}(t) = N_i(t)^{-1} \sum_{t:I_t=i} r_t^0$ la moyenne empirique du bras i pré-attaque à l'étape t (observée seulement par Alice) et $\hat{\mu}_i(t) = N_i(t)^{-1} \sum_{t:I_t=i} r_t$ la moyenne empirique du bras i post-attaque (la seule à laquelle Bob a accès).

b) Cas oracle

Dans un premier temps et afin de disposer d'un point de référence on s'intéresse au cas dit « oracle » où Alice connaît la véritable distribution des bras et Bob utilise n'importe quel algorithme dont le regret est $O(\log(T))$.

Dans ce scénario, Alice peut adopter une stratégie optimale en ajustant les récompenses pour induire Bob en erreur. Si Bob tire le bras K ou un bras moins performant que K alors Alice n'attaque pas, sinon Alice attaque le bras d'autant plus fortement que celui-ci surperforme par rapport à K. Il lui suffit donc de choisir, à chaque tour t une attaque telle que $\alpha_t = \Delta_{I_t}^{\epsilon} = \max\{\mu_{I_t} - \mu_K + \epsilon, 0\} \times \mathbb{I}[I_t \neq K]$. On peut alors montrer que l'attaque d'Alice réussit (on obtient bien $\mathbb{E}[N_K(T)] = T - o(T)$) avec un coût total d'attaque de l'ordre de $O(\sum_{i=1}^{K-1} \Delta_i^{\epsilon} \times \log(T))$ quelque quoi l'algorithme utilisé par Bob.

c) Cas général

On se place désormais dans le cas plus général où Alice n'a pas connaissance de la distribution (scénario plus plausible dans un contexte pratique). Les auteurs démontrent qu'il est possible d'obtenir une performance légèrement dégradée par rapport au cas oracle mais tout de même efficace (selon les critères définis en (a)).

Attaque sur ϵ-greedy

On s'intéresse tout d'abord au cas où Bob utilise l'algorithme ϵ-greedy. C'est-à-dire que selon un schéma d'exploration défini par $\{\epsilon_t\}_{1\leq t\leq T}$ (où la suite (ϵ_t) est décroissante) Bob choisit ses bras de la manière suivante :

$I_1 \dots I_K = K, \dots, 1$ *(Bob teste une première fois chacun des bras en commençant par K)*
Pour t de K+1 à T :
- $I_t = \mathcal{U}(\mathrm{K})$ avec probabilité ϵ_t (*exploration uniforme*)
- $I_t = \underset{i}{\operatorname{argmax}}\, \widehat{\mu_i}(t-1)$ avec probabilité $1-\epsilon_t$ (*exploitation*)

Ici Alice n'a pas de prise sur (ϵ_t). Son seul levier d'action est donc la phase d'exploitation où son objectif sera de faire en sorte que Bob tire très souvent le bras K. Cela revient à dire que la moyenne empirique du bras K qu'observe Bob doit à tout instant, être la meilleure des moyennes empiriques avec forte probabilité (puisque c'est selon ce critère que Bob choisit le bras dans les phases d'exploitation). Concrètement, pour un tour t, si un bras différent de K est choisi (par exemple dans le cadre de l'exploration) l'idée pour Alice va être de dégrader artificiellement la moyenne empirique de ce bras pour la faire tomber en dessous de la moyenne empirique du bras K.

Pour définir l'attaque adéquate, l'approche utilisée par les auteurs est la suivante. Dans un premier temps on se fixe une inégalité à respecter sur les moyennes empiriques pour induire Bob en erreur puis on utilise la relation liant les moyennes empiriques (observées par Bob) du bras I_t entre les étapes $t-1$ et t[1]. Ces deux équations permettent alors immédiatement de trouver l'attaque à appliquer sur la récompense du bras I_t afin de satisfaire l'inégalité.

Ici on veut obtenir l'inégalité $\widehat{\mu_{I_t}}(t) \leq \widehat{\mu_K}(t) - 2\beta\big(\mathrm{N_K}(t)\big)$[2] ce qui permet de déduire la stratégie d'attaque suivante : $\alpha_t = \left[\widehat{\mu_{I_t}}(t-1) \times \mathrm{N_{I_t}}(t-1) + r_t^0 - \mathrm{N_{I_t}}(t)\big(\widehat{\mu_K}(t) - 2\beta(\mathrm{N_K}(t))\big)\right]_+ \; \forall\, t \in [1,T]$ si $I_t \neq \mathrm{k}$

On obtient sous cette stratégie des résultats très satisfaisants (**Théorème 1**). Tout d'abord, sous l'événement E[3] (événement se réalisant avec la probabilité $1-\delta$ qui correspond aux conditions sous lesquelles notre coût se comporte bien), Alice force Bob à utiliser le bras K dans l'intégralité des tours d'exploitation. D'autre part en prenant $\epsilon_t = cK/t$, le coût total de l'attaque d'Alice est en $\hat{O}\left(\left(\sum_{i=1}^{K}\Delta_i\right) \times logT + \sigma K \times \sqrt{\log(T)}\right)$[4]. Ce coût dépend des termes Δ_i définis comme $\max_i(\mu_i - \mu_K, 0)$ ce qui semble raisonnable : plus la moyenne théorique du bras que l'on veut pousser est faible par rapport aux meilleurs bras plus les attaques devront être importantes afin de compenser l'écart.

Finalement, sous les conditions peu contraignantes du Théorème 1, les auteurs parviennent à calculer une borne inférieure pour $N_K(T)$ et une borne supérieure pour le coût $\sum_{t=1}^{T}|\alpha_t|$ qui garantissent le succès de l'attaque (à savoir $N_K(T) = T - o(T)$ et $\sum_{t=1}^{T}|\alpha_t| = O(\log(T))$ avec une probabilité supérieure à $1-\delta$.

---

[1] $\widehat{\mu_{I_t}}(t) = N_i(t)^{-1}\left[\widehat{\mu_{I_t}}(t-1) \times \mathrm{N_{I_t}}(t-1) + r_t^0 - \alpha_t\right]$

[2] où $\beta(N)$ est défini comme $\sqrt{\frac{2\sigma^2}{N}\log\left(\frac{\pi^2 K N^2}{3\delta}\right)}$, δ choisi dans $\left[0, \frac{1}{2}\right]$

[3] E : $= \{\forall i, \forall t > K : \left|\hat{\mu}_i^0(t) - \mu_i\right| < \beta\big(N_i(t)\big)\}$

[4] La notation $\hat{O}$ indique qu'on ignore les termes en $\log(\log(T))$

Attaque sur UCB

On s'intéresse ensuite au cas où Bob utilise l'algorithme UCB. C'est-à-dire que Bob choisit ses bras de la manière suivante.

$I_1 \dots I_K = 1, \dots, \mathrm{K}$ *(Bob teste une première fois chacun des bras)*
Pour t de K+1 à T :

- $I_t = arg\max_i \hat{\mu}_i(t-1) + 3\,\sigma\sqrt{\log(t)/Ni(t-1)}$ *(on choisit le bras pour lequel la borne supérieure de la récompense empirique est la plus élevée).*

L'objectif d'Alice est donc ici de faire en sorte que la borne supérieure du bras K soit le plus souvent possible la plus élevée parmi les bras.

La stratégie proposée par les auteurs est qu'Alice n'attaque pas pendant les K premiers tours puis attaque seulement pour les bras différents de K à partir du K-ème tour. On procède de façon similaire à l'approche utilisée pour le cas ϵ-greedy. Pour un bras $i \neq k$ choisi au tour t, Alice choisit l'attaque α_t minimale permettant de satisfaire la relation : $\hat{\mu}_i(t) \leq \hat{\mu}_K(t-1) - 2\beta\big(N_K(t-1)\big) - \Delta_0$ où Δ_0 est un paramètre positif choisi par Alice. Cela revient à prendre comme valeur de l'attaque : $\alpha_t = \Big[\mathrm{N_{I_t}}(t)\widehat{\mu_{I_t}^0}(t) - \sum_{t' \leq t-1 : I_{t'}=i} \alpha_{t'} - \mathrm{N_{I_t}}(t) \times \big(\widehat{\mu_K}(t-1) - 2\beta\big(N_K(t-1)\big) - \Delta_0\big)\Big]_+$ $\forall t$ si $I_t \neq$k.

Le paramètre Δ_0 est un paramètre d'ajustement pour Alice qui détermine à quel point on veut « éloigner » les moyennes empiriques des bras $\{1, \dots, K-1\}$ de celle du bras K. Plus il augmente plus l'attaque est performante (c'est-à-dire plus le nombre de fois que Bob tire K augmente) mais plus le coût d'attaque augmente. Les auteurs recommandent de choisir $\Delta_0 = O(\sigma)$, valeur avec laquelle on obtient un coût d'attaque en $\hat{O}\Big(\big(\sum_{i=1}^{K-1}\Delta_i\big) \times \log(T) + \sigma K \times \log(T)\Big)$. Cette borne est légèrement moins satisfaisante que celle obtenue avec l'attaque contre ϵ-greedy (pour laquelle le deuxième terme était en $\sqrt{\log(T)}$*).* Mais il est à noter que si l'on connaît dès le départ l'échéance T et en prenant alors $\Delta_0 = O\big(\sigma\sqrt{\log(T)}\big)$ on peut optimiser le coût en $\hat{O}\big(\sigma K \times \sqrt{\log(T)}\big)$, ce qui permet de se départir des termes Δ_i.

Finalement, le **Théorème 2** garantit à nouveau qu'avec une probabilité d'au moins $1-\delta$, l'attaque est une réussite, c'est-à-dire $N_K(T) = T - o(T)$ et $\sum_{t=1}^{T} |\alpha_t| = O(\log(T))$.

d) Simulations

Suite à ces démonstrations théoriques, les auteurs appliquent leur stratégie sur des cas de simulation avec deux bras (un bras optimal et un « mauvais bras » qu'Alice va pousser à Bob). Ces simulations confirment bien les hypothèses démontrées au préalable comme le montrera la partie (III) pour le cas ϵ-greedy.

En comparant le comportement avec et sans attaque on confirme bien que l'attaque conduit l'agent à choisir le mauvais bras à la quasi-totalité des tours alors même qu'il aurait très peu été choisi sans attaque. On peut également confirmer l'influence de Δ_i : plus il augmente, plus le coût de l'attaque est important (puisqu'Alice doit compenser un fort écart de performance théorique entre le bras optimal et le bras qu'elle tente de pousser). Enfin l'influence de σ est aussi à prendre en compte : plus il augmente, plus le coût d'attaque augmente pour Alice.

III. Implémentation dans le cas ϵ-greedy

Afin de compléter ce résumé, on se propose d'implémenter l'algorithme d'attaque proposé par les auteurs dans le cas epsilon-greedy. Le détail du code est disponible dans le fichier .ipynb joint.

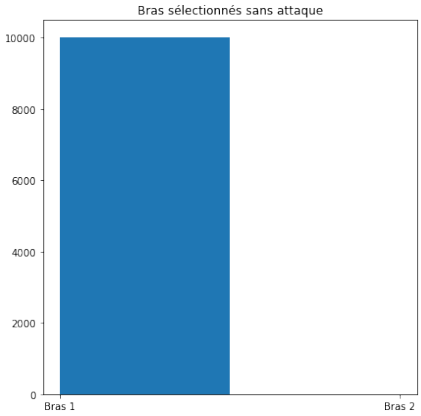
On se place dans la même configuration que dans l'article : Bob est face à deux bras avec $\mu_1 = \Delta_1 > 0$ et $\mu_2 = 0$. Alice a pour cible le deuxième bras (bras sous optimal).

a) ***Vérification des propriétés sur un premier exemple***

Dans un premier temps, on fixe $\mu_1 = \Delta_1 = 2$, $\delta = 0.025$, $T = 10\,000$ et $\epsilon_t = \frac{1}{t}\ \forall t$ et on compare les résultats avec et sans attaque sur des données simulées.

Le résultat est sans appel et confirme les démonstrations des auteurs. Sans attaque Bob joue seulement 4 fois le deuxième bras, mais lorsqu'il est soumis aux attaques d'Alice il inverse totalement son comportement en jouant 9 998 fois sur 10 000 le second bras (bras ciblé par Alice) comme le montre la *figure 1*. En traçant la courbe du nombre de fois que le bras K est tiré en fonction du temps (*figure 2*), on retrouve bien la relation linéaire souhaitée : $N_K(T) = T - o(T)$.

Figure *1 : Bras sélectionnés avec et sans attaque*



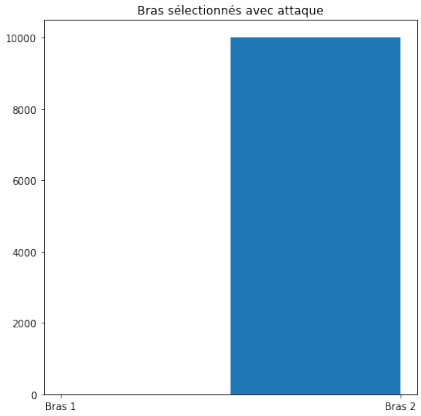
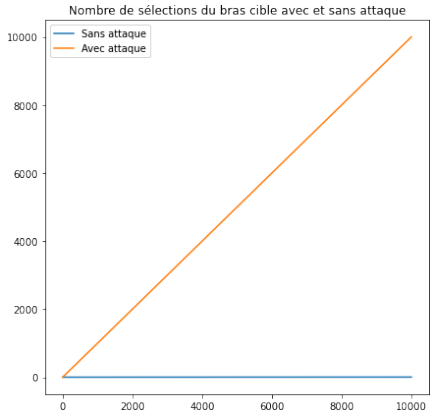


Figure 2 : Sélection du bras ciblé par Alice avec et sans attaque



Même si les regrets des deux scénarios (attaque, pas d'attaque) ne sont pas comparables en tant que tels (puisqu'Alice manipule les récompenses) il est intéressant d'observer que sans attaque le regret immédiat converge très rapidement vers 0 (le meilleur bras est identifié rapidement et la probabilité d'exploration décroit très vite puisque $\epsilon_t = \frac{1}{t}$). Dans le second scénario, l'attaque force le regret à converger vers la valeur sous-optimale de $\Delta_1 = 2$ (*Figure 3*). Dans cet exemple, le coût d'attaque pour Alice (*Figure 4*) est très faible (valeur totale de 7) et ne connait que deux paliers (Alice n'a attaqué que lors des deux tours où Bob a sélectionné le premier bras).

Figure *3 : Convergence du regret immédiat avec et sans attaque*

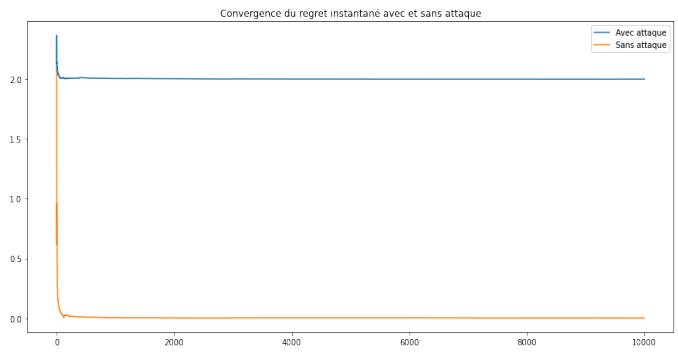
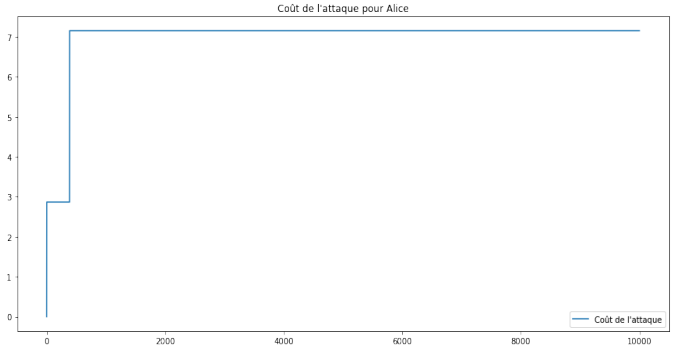


Figure 4 : Coût de l'attaque (cumulé) pour Alice



b) ***Influence des paramètres Δ_1 et σ***

On s'intéresse aux paramètres Δ_1 (qui traduit l'écart de performance entre le meilleur bras et le bras sous-optimal qu'Alice veut pousser à Bob) et σ (l'écart type des gaussiennes simulant les récompenses). Pour étudier leur impact sur le coût d'attaque pour Alice on réalise n simulations (de T tours chacune) pour trois valeurs différentes du paramètre (détails *figure 5*). On trace ensuite le coût moyen cumulé d'attaque pour Alice pour chacun de ces scénarios (*Figure 6 et 7*).

Figure *5 : Paramètre des simulations*

Scénario	Δ_1	σ	T	δ	ϵ_t	n
Δ_1 varie	1, 2 et 5	0.1	1000	0.025	1/t	100
σ varie	2	0.1, 0.2 et 0.5	1000	0.025	1/t	100

Tout d'abord, en traçant les coûts cumulés en fonction du logarithme de t, on retrouve bien la propriété attendue à savoir $\sum_{t=1}^{T} |\alpha_t| = O(\log(T))$. On constate également que le coût d'attaque augmente d'autant plus que Δ_1 augmente (Alice doit davantage attaquer afin de dégrader les récompenses du meilleur bras). La situation est similaire lorsque σ augmente (Alice doit augmenter ses récompenses afin de compenser son incertitude plus grande sur la moyenne théorique du meilleur bras).

Figure 6 : *Coût d'attaque pour différentes valeurs de* Δ_1

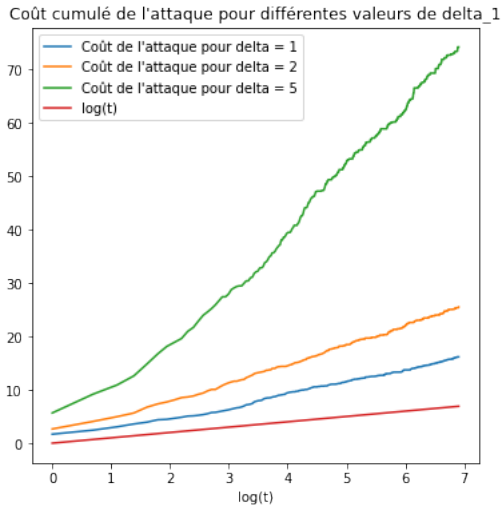
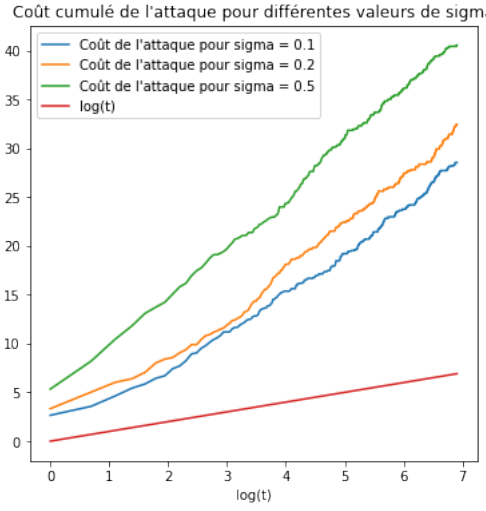


Figure 7 : *Coût d'attaque pour différentes valeurs de* σ



c) Piste pour une stratégie de défense de la part de Bob

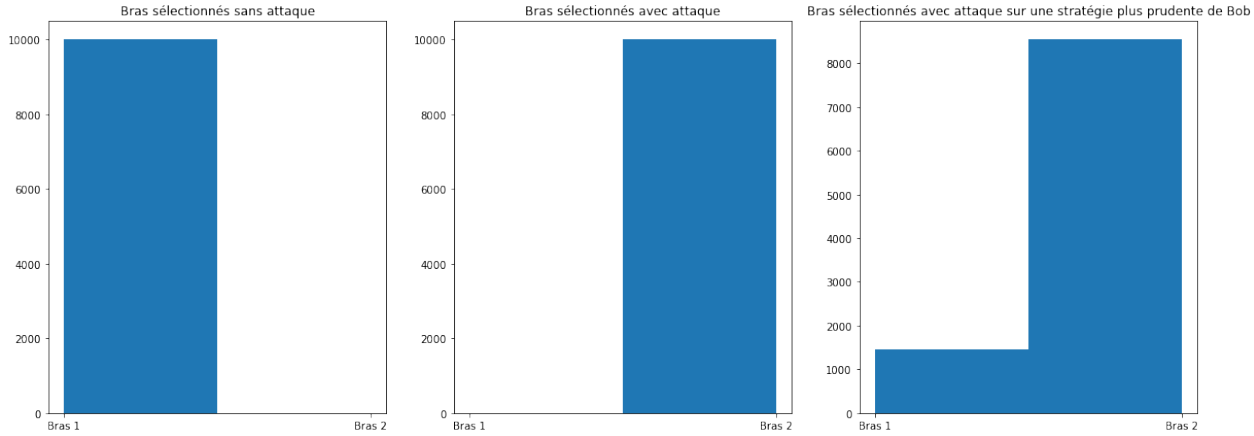
En guise de conclusion et pour donner une autre perspective sur le problème on s'interroge sur la possibilité pour Bob de contrecarrer les plans d'Alice.

Dans les scénarios étudiés, Bob choisit le bras sous-optimal dans la quasi-totalité des cas. Cela est dû bien sûr à la stratégie d'Alice qui dégrade volontairement les récompenses du meilleur bras – chose contre laquelle Bob ne peut rien faire. Mais un autre paramètre qui rentre en ligne de compte est la stratégie d'exploration choisie par Bob. En effet la probabilité d'exploration pour un tour t est ici $\epsilon_t = \frac{1}{t}\ \forall t$ ce qui implique qu'après quelques tours la probabilité d'explorer devient très faible et on se contente de jouer le meilleur bras sans discontinuer (en l'occurrence le mauvais bras).

On se propose alors d'observer l'impact d'une stratégie plus prudente de la part de Bob dans laquelle la probabilité d'exploration reste fixe au cours de la partie ($\epsilon_t = 0.3\ \forall t$). On simule une partie en 10 000 tours et on compare à présent trois scénarios : le scénario sans attaque avec $\epsilon_t = \frac{1}{t}\ \forall t$, le scénario avec attaque et $\epsilon_t = \frac{1}{t}\ \forall t$ et le scénario avec attaque avec $\epsilon_t = 0.3\ \forall t$.

La *figure 8* présente la répartition des bras choisis par Bob. Comme on s'y attendait, l'approche plus prudente de Bob permet mécaniquement d'augmenter le nombre de sélections du premier bras (avec $\epsilon_t = 0.3\ \forall t$, on obtient environ 30% de tours d'exploration, et sur chacun de ces tours le premier bras a une chance sur deux d'être sélectionné, il est donc logique qu'il soit sélectionné dans environ 15% des cas).

Figure 8 : *Bras sélectionnés avec et sans attaque pour la stratégie classique et avec attaque pour la stratégie prudente*



En traçant les courbes de regret immédiat (*Figure 9*), on comprend que cette stratégie n'est pas idéale pour Bob puisqu'il s'expose à des attaques de plus en plus fortes d'Alice faisant très fortement augmenter son regret immédiat qui ne converge même pas. Le regret total est multiplié par 108 pour Bob (par rapport à son regret total en cas d'attaque avec $\epsilon_t = \frac{1}{t}\ \forall t$). Mais dans le même temps le coût d'attaque pour Alice est multiplié par 34 840 et on observe immédiatement qu'il ne respecte plus la condition de succès $\sum_{t=1}^{T} |\alpha_t| = O(\log(T))$ (*Figure 10*).

Figure 9 : *Regret immédiat*

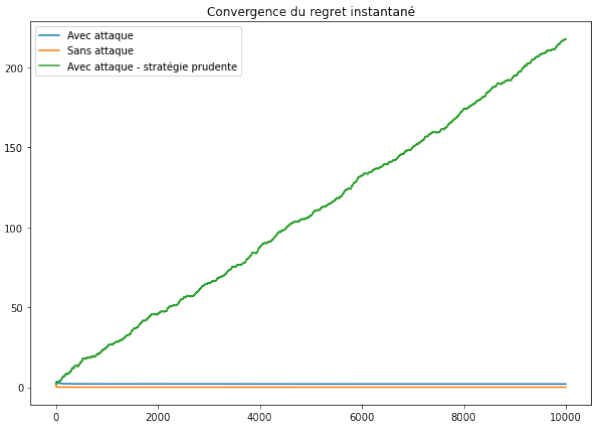
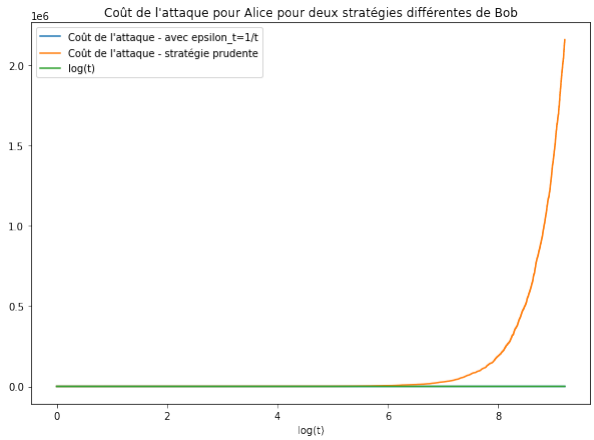


Figure 10 : Coût de l'attaque (cumulé) pour Alice



Ainsi au prix – certes - d'une forte dégradation de sa performance, Bob parvient à empêcher Alice de maintenir un coût d'attaque raisonnable. Cette première simulation donne l'intuition que dans des scénarios où le budget d'attaque serait limité, des approches plus aléatoires et exploratoires permettraient de déjouer ou tout du moins de contrecarrer la stratégie d'un adversaire malicieux.

IV. Conclusion

En conclusion les auteurs de l'article ont démontré qu'il était possible pour un adversaire malicieux de contourner les algorithmes ϵ-greedy et UCB afin de pousser leur utilisateur à tirer presque systématiquement un bras sous-optimal et ce avec un coût logarithmique par rapport au temps.

Notre implémentation a permis de confirmer ces assertions tout en donnant une intuition sur des premières stratégies de défense. Il serait également intéressant d'étendre cette étude à la fois à des algorithmes plus complexes (Thompson, bandits contextuels) mais aussi à des objectifs d'attaques différentes (maximiser le regret, éviter certains bras etc).