Confidentialité dans les systèmes de réputation

Grégory Bonnet[a] gregory.bonnet@unicaen.fr
Laurent Vercouter[b] laurent.vercouter@insa-rouen.fr
Damien Lelerre[a] damien.lelerre@unicaen.fr

[a]Normandie Université, UNICAEN, ENSICAEN, CNRS, GREYC, 14000 Caen, France

[b]Normandie Université, UNIROUEN, UNIHAVRE, INSA Rouen, LITIS, 76000 Rouen, France

Résumé

Dans les systèmes de réputation, il a été observé, qu'afin d'éviter des évaluations vengeresses, les témoignages étaient majoritairement positifs, réduisant par là-même l'efficacité du système. Pour inciter les agents à diffuser tous leurs témoignages, les solutions classiques proposent d'anonymiser les témoignages. Dans la littérature, de nombreux travaux ont étudé des approches cryptographiques pour s'assurer à la fois de l'anonymat et de la non-répudiation des témoignages. Toutefois, ceci ne permet de garantir leur confidentialité en raison des corrélations entre transactions et diffusion d'un nouveau témoignage. Dans cet article, nous proposons d'étudier la faisabilité d'une autre approche dans laquelle les témoignages sont bruités et soumis à des délais de diffusion. Des résultats d'expérimentation mettent en lumière l'effet de ces perturbations sur les systèmes de réputation BetaReputation et EigenTrust.

Mots-clés : *Système de confiance et de réputation, Confidentialité*

Abstract

On reputation systems, it has been observed that testimonies are mostly positive in order to avoid vengeful evaluations. Such behaviour reduces the effectiveness of the system. In order to incite agents to publish all their testimonies, classic solutions consist in anonymizing them. In the literature, many works have proposed cryptographic approaches to ensure both anonymity and non-repudiation of testimonies. However, due to correlations between transactions and dissemination of new testimonies, it does not guarantee confidentiality. In this article, we propose to study the feasibility of another approach in which a noise is applied on testimonies and those latter are subject to delays of diffusion. Experimental results highlight the effect of these disturbances on the BetaReputation and EigenTrust reputation systems.

Keywords: *Trust and reputation system, Confidentiality*

1 Introduction

Dans un système multi-agents, il est courant qu'un agent ait besoin des services ou des ressources d'un autre agent. Deux agents peuvent effectuer des *transactions* afin que chacun obtienne ce dont il a besoin. Cependant, un agent peut être plus ou moins qualifié (ou plus ou moins honnête) dans la fourniture de ce qui lui est demandé. Il est donc dans l'intérêt d'un agent de bien choisir ceux avec lesquels il effectue des transactions. Une des approches permettant ceci est l'utilisation de *systèmes de réputation* [10, 15, 19]. Ces systèmes se fondent sur l'attribution de notes évaluant la façon dont une transaction s'est déroulée. Chaque agent est alors en mesure de développer dans un premier temps une *confiance* envers un autre agent à partir de sa propre expérience passée. Dans un second temps, les notes données par chaque agent peuvent être diffusées publiquement sous forme de *témoignages* et aggrégées par une fonction afin de construire une notion de confiance collective plus précise appelée *réputation*.

Sur les sites de vente en ligne utilisant des systèmes de réputation (et où les agents sont majoritairement des humains), il a été observé que les témoignages sont très majoritairement positifs. L'explication de ce phénomène repose sur le fait que, puisque ces notes sont publiques, il y a un risque d'un agent désire *se venger* lorqu'il reçoit une mauvaise évaluation de la part d'un tiers [13]. Pour éviter les vengeances, les agents préfèrent alors uniquement diffuser des témoignages positifs, réduisant ainsi l'efficacité du système de réputation. Afin d'inciter les agents à diffuser tous leurs témoignages, les solutions classiques proposent d'*anonymiser* les témoignages à l'aide d'un protocole cryptographique, permettant ainsi d'obtenir la réputation d'un autre agent sans connaitre les témoignages qui ont servi à calculer celle-ci [2, 7, 17, 20]. Toutefois, ne plus rendre les témoignages publics n'est pas suffisant car il pourrait être possible de déduire leur valeur de la dynamique

de la réputation. C'est pourquoi nous proposons d'étudier la faisabilité d'une autre approche, fondée sur la *confidentialisation* et dans laquelle les témoignages sont bruités et soumis à des délais, décorrélant ainsi les transactions de la diffusion de témoignages.

Le reste de cet article est structuré comme suit. Nous présentons en section 2 les systèmes de réputation, leurs propriétés principales ainsi que deux systèmes particuliers, BetaReputation et EigenTrust, qui nous serviront de référence pour nos expérimentations. La section 3 présente les méthodes de confidentialisation que nous utilisons et le système de réputation que nous proposons. Enfin, la section 4 étudie expérimentalement les effets de ces méthodes sur la performance des fonctions de réputation.

2 Systèmes de réputation

Les systèmes de réputation sont des systèmes d'évaluation distribués, et parfois décentralisés, de la fiabilité des agents. Dans cette section, nous présentons les principaux concepts associés à ces systèmes et détaillons les systèmes qui nous serviront de référence.

2.1 Principes généraux

Dans un système multi-agent, chaque agent peut posséder certaines compétences ou certaines ressources. Si un agent a besoin d'une compétence ou d'une ressource qu'il ne possède pas, il peut réaliser une *transaction* avec un agent qui la possède. Toutefois, chaque agent n'est pas qualifié de façon égale pour fournir un service. De plus, il est possible qu'il soit profitable au fournisseur de service d'en fournir délibérément un de mauvaise qualité. Aussi, lorsqu'un agent a besoin d'un service, il est dans son intérêt d'obtenir un service de bonne qualité, et donc de choisir un fournisseur fiable. Pour choisir un fournisseur, un agent peut se fonder sur ses propres observations afin de mesurer à quel point il peut avoir *confiance* en un autre agent. Au moment d'initier une transaction, un agent doit : (1) choisir un agent en fonction de la confiance qu'il a en lui ; (2) observer le résultat de la transaction ; (3) évaluer ce résultat pour mettre à jour la confiance qu'il attribue au fournisseur de service.

En fondant uniquement la confiance sur ses observations, un agent ne va pas pouvoir évaluer un autre qui lui est inconnu, tout comme il se prive des informations que d'autres agents pourraient avoir. La notion de confiance peut alors être étendue par celle de *réputation*. Cette dernière est une évaluation de la fiabilité d'un agent à partir des observations des autres agents. Pour ce faire, à chaque transaction, les agents publient [1] leurs observations sous forme de *témoignages*. Une *fonction de réputation* permet ensuite à un agent donné d'agréger ces témoignages en une valeur unique. Les fonctions de réputation peuvent être classées selon leurs propriétés [6, 10, 15, 19]. Parmi celles-ci, deux propriétés nous intéressent :

Visibilité. Une fonction de réputation peut être *globale* ou *personnalisée*. Une fonction personnalisée construit une réputation dépendante du point de vue de l'agent qui la calcule : deux agents peuvent calculer deux valeurs de réputation différentes pour un même autre agent. Une fonction est globale si la réputation d'un agent est la même quelque soit l'agent qui la calcule.

Sémantique. Une fonction de réputation peut être *par valeur* ou *par rang*. Une fonction de réputation est par valeur si les valeurs de réputation qu'elle calcule représentent une information en elles-mêmes. Par exemple, certaines fonctions retournent la valeur moyenne des notes attribuées à un agent ou la probabilité que la prochaine transaction sera de bonne qualité. En revanche, une fonction par rang n'attribue pas de sens aux valeurs de réputation en soi mais uniquement à l'ordre qu'elles impliquent entre les agents. Ainsi, un agent ayant une meilleure réputation qu'un autre sera considéré comme meilleur mais sans qu'il soit possible de le quantifier. Notons qu'une fonction par valeur permet aussi de construire un ordre sur les agents.

2.2 Systèmes de référence

De nombreuses fonctions de réputation ont été proposées dans la littérature [4, 9, 11, 14, 18, 22]. Dans la suite de cet article, nous considérons les deux fonctions suivantes, à savoir BetaReputation [9] et EigenTrust [11], car ce sont deux fonctions de référence dans la littérature tout en étant à l'opposé l'une de l'autre sur chacune des propriétés que nous avons mentionné précédemment.

BetaReputation. Cette fonction *personnalisée* et *par valeur* caractérise la réputation d'un agent comme la probabilité qu'il fournisse un service de bonne qualité. Chaque agent a_j représente

1. Ceci peut se faire de diverses façons : mise à disposition sur demande, publication auprès d'une autorité centrale, diffusion générale au système, propagation de voisinage en voisinage, etc.

sa confiance envers un agent a_j par un couple $r_{i,j} \in \mathbb{N}$ et $s_{i,j} \in \mathbb{N}$, correspondant respectivement au nombre de « bonnes » et « mauvaises » transactions avec a_j. Lors d'une nouvelle transaction (r', s') à l'instant t, ces deux valeurs sont mises à jour avec un facteur d'oubli $\lambda \in [0, 1]$:

$$r_{i,j}^t = \lambda r^{t-1} + r'$$
$$s_{i,j}^t = \lambda s^{t-1} + s'$$

Pour simplifier les notations, hormis dans les cas ambigus, nous omettons dans la suite l'exposant t. La réputation d'un agent a_j du point de vue d'un agent a_i est donnée par :

$$Rep_j^i = \frac{R_j^i - S_j^i}{R_j^i + S_j^i + 2}$$

où :

$$R_j^i = \sum_{a_k \notin \{a_i, a_j\}} \frac{2r_{i,k} \times r_{k,j}}{(s_{i,k}+2)(r_{k,j}+s_{k,j}+2)+2r_{i,k}}$$

$$S_j^i = \sum_{a_k \notin \{a_i, a_j\}} \frac{2r_{i,k} \times s_{k,j}}{(s_{i,k}+2)(r_{k,j}+s_{k,j}+2)+2r_{i,k}}$$

EigenTrust. EigenTrust est une fonction de réputation *globale* et *par rang* où chaque agent représente sa confiance envers un agent a_j par un couple $r_{i,j} \in \mathbb{N}$ et $s_{i,j} \in \mathbb{N}$ de « bonnes » et « mauvaises » transactions. EigenTrust s'appuie ensuite sur une matrice de confiance normalisée $\mathcal{C}$ où chaque élément $c_{i,j}$ est défini comme suit :

$$c_{i,j} = \frac{\max(r_{i,j} - s_{i,j}, 0)}{\sum_{a_j \notin \{a_i\}} \max(r_{i,j} - s_{i,j}, 0)}$$

La réputation des agents est un vecteur Rep où la ième composante de Rep, notée Rep_i, est la réputation de l'agent a_i. Ce vecteur est défini comme le point fixe [2] de l'équation suivante où $\vec{p}$ est un vecteur de réputation *a priori* :

$$Rep^{(t+1)} = (1-\alpha) \times \mathcal{C}^T \times Rep^{(t)} + \alpha \times \vec{p}$$

2.3 Politiques de sélection

Calculer des réputations n'est pas suffisant, encore faut-il que les agents s'en servent pour décider avec qui effectuer des transactions. Or, toujours porter son choix sur les agents qui disposent de la meilleure réputation à un instant donné peut conduire à se priver d'information sur les autres agents. Aussi, une politique de sélection doit permettre un compromis entre l'*exploitation* qui correspond au fait de sélectionner des agents ayant une bonne réputation dans l'espoir d'obtenir la meilleure transaction possible, et l'*exploration* qui consiste à choisir un agent non pas parce qu'il a bonne réputation mais pour obtenir une observation supplémentaire et mieux estimer sa fiabilité. Les trois méthodes que nous considérons sont inspirées des politiques de sélection pour le problème du bandit multi-bras [3, 21].

ϵ-gloutonne. Connue pour être simple mais peu performante, cette politique sélectionne l'agent qui dispose de la meilleure réputation avec une probabilité $(1-\epsilon)$ ou sélectionne un agent tiré aléatoirement de manière uniforme avec une probabilité ϵ. Cependant, cela peut conduire à la sélection d'un agent évalué comme peu fiable avec la même probabilité qu'un agent sur lequel aucune information n'est disponible (d'où parfois une faible performance).

UCB. Connue pour être très performante, cette politique choisit l'agent a_j qui maximise une valeur v_j avec :

$$v_j^i = Rep_j^i + \sqrt{\frac{2 \times \ln(1 + r_{i,j} + s_{i,j})}{1 + \Sigma_{a_k \in N}(r_{k,j} + s_{k,j})}}$$

Cela permet une exploration contextuelle qui dépend de la quantité d'information dont dispose l'agent sur les autres. Tant que l'agent dispose de peu d'information, le second terme domine le premier et conduit l'agent à sélectionner ceux sur lesquels il dispose du moins d'observations. Toutefois, cette politique est peu performante face à des agents non stationnaires [3].

β-softmax. Connue pour être moins performante que UCB mais plus robuste aux comportements non stationnaires, cette politique sélectionne les agents proportionnellement à leur réputation, avec une probabilité p_j^i tirée selon une fonction exponentielle normalisée :

$$p_j^i = \frac{e^{\beta . Rep_j^i}}{\Sigma_{a_k \in N} e^{\beta . Rep_j^i}}$$

Le paramètre $\beta \in \mathbb{R}$ est une température inverse. Si $\beta = 0$, la distribution de probabilité est une

2. Le calcul effectué est généralement un calcul approché à une erreur ϵ près. De plus, cette équation n'est pas garantie de converger vers un point fixe. Le facteur $\alpha \in]0, 1]$ permet de s'en assurer s'il est fixé à une valeur non nulle.

3. Un agent est non stationnaire si sa fiabilité change au fil du temps.

distribution uniforme : chaque agent est sélectionné avec la même probabilité. Si β tend vers ∞, la probabilité de sélectionner l'agent ayant la plus haute réputation tend vers 1 tandis que celles des autres tendent vers 0.

3 Confidentialiser les témoignages

Nous proposons dans cette section deux méthodes pour accroître la confidentialité des témoignages, qui s'inspirent de méthodes utilisées dans le domaine des bases de données. Lorsque des données stockées dans une base doivent rester confidentielles, la première étape est l'*anonymisation* en supprimant les informations permettant d'identifier un individu. Elle n'est cependant pas suffisante car il peut être possible reconstruire les informations cachées, par recoupement par exemple avec d'autres sources d'information. Ainsi, une seconde étape de *confidentialisation* [4] doit être mise en place [1]. Il s'agit de modifier une base de données anonymisée pour empêcher la reconstruction des identités des individus associés aux données.

Quelques travaux se sont déjà intéressés à la confidentialisation des témoignages pour un système de réputation. Hasan *et al.* [7] indiquent qu'une solution consisterait à bruiter les témoignages afin de rendre plus difficile leur déduction à partir de la dynamique de la réputation. Toutefois, les auteurs se sont bornés à ce constat. De manière intéressante, Huang *et al.* [8] ont proposé de confidentialiser les témoignages en les agrégeant avant de les communiquer. Un agent ne déclare plus avoir confiance en un agent donné mais uniquement en un groupe donné. Plus les groupes considérés sont de grande taille, plus la confidentialité est forte car il est difficile d'y distinguer deux agents.

Nous proposons de partir du constat de Hasan *et al.* et d'étudier l'effet d'un bruit appliqué aux témoignages sur les systèmes de réputation. Nous distinguons deux types de bruits : un bruit sur les témoignages eux-mêmes qui correspond au bruit classiquement utilisé pour la confidentialisation des bases de données et un bruit sur la diffusion des témoignages sous forme d'un délai. En effet, si un agent veut observer l'évolution de sa réputation pour déduire la valeur des nouveaux témoignages à son encontre, il peut supposer que ces nouveaux témoignages sont diffusés aussitôt ou presque après la transaction. L'introduction d'un délai vient décorreler ces deux événements.

4. *Privacy preservation.*

3.1 Bruit sur les témoignages

L'ajout de bruit est une méthode courante pour confidentialiser une base de données [5, 16]. Cela consiste à altérer les valeurs numériques de cette base de données afin que les informations relatives à un même individu ne puissent plus être mises en corrélation. Toutefois, l'ajout de bruit modifie le résultat des requêtes : plus les données sont altérées, plus la confidentialité est forte mais moins l'information globale sur la base est conservée. Parmi les méthodes de bruitage, le bruit différentiel [5] permet d'éviter ce problème en calculant un bruit spécifique pour chaque requête possible sur la base tout en minimisant la dissimilarité entre les résultats des requêtes sur la base bruitée et non bruitée. Si cette méthode est particulièrement intéressante, elle s'accommode mal des données dynamiques comme peuvent l'être l'ensemble des agents et leurs témoignages dans un système de réputation. Il serait nécessaire de recalculer l'ensemble de la base synthétique lors de l'arrrivée d'un nouvel agent ainsi qu'à chaque nouveau témoignage. De plus, cette méthode est nécessairement dépendante de la fonction de réputation utilisée. Nous écartons donc pour ces raisons le bruit différentiel.

3.2 Délais de diffusion

Nous proposons d'utiliser différentes méthodes appliquant un délai sur la diffusion des témoignages afin de décorréler la transaction de son évaluation. Une approche naïve consiste simplement à faire usage d'un délai « temporel ». Un témoignage généré à un instant t ne sera diffusé (nous dirons *valide* par la suite) qu'à partir de l'instant $t + \epsilon$, où ϵ est un nombre aléatoire tiré uniformément dans un intervalle donné en paramètre. Ainsi, deux témoignages valides peuvent être confondus et l'agent concerné ne peut pas savoir à quelle transaction associer chacun. Toutefois, une stratégie consistant à attendre suffisamment entre deux transactions pour voir les témoignages être validés peut mettre à mal cette approche. Afin de passer outre ce problème, nous proposons de faire usage d'un délai non plus temporel mais de ne pas valider un témoignage tant qu'un certain nombre d'autres témoignages venant de témoins distincts n'a pas été généré. Cette méthode, en exigeant une diversité de témoins, évite qu'un agent qui serait évalué majoritairement par un unique agent puisse savoir à quelle transaction associer un témoignage. Toutefois, si un agent n'est que peu sollicité, un té-

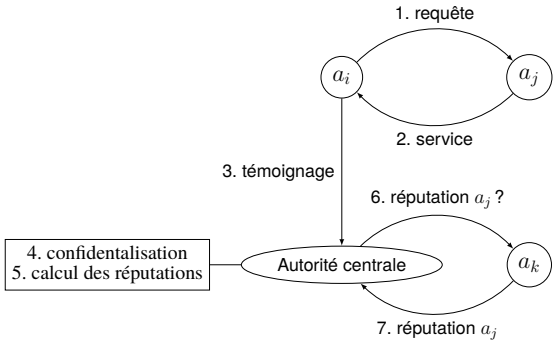


FIGURE 1 – Protocole de confidentialisation

moignage peut rester indéfiniment en attente.

3.3 Un système de réputation confidentiel

Le système de réputation à témoignage confidentiel que nous proposons est illustré en figure 1. Ici, une autorité centrale est en charge du stockage des témoignages et de l'application du bruit et du délai de diffusion. Un agent qui effectue à l'instant t une transaction avec un autre en observe la qualité et évalué celle-ci dans l'intervalle $[-1, 1]$ (-1 pour la plus mauvaise qualité possible, 1 pour la meilleure). Cette évaluation correspond à un témoignage $o_{i,j}$ qui est envoyé à l'autorité centrale. Cette dernière applique en premier lieu un *bruit additif* et enregistre le témoignage bruité $\tilde{o}_{i,j}$ tel que $\tilde{o}_{i,j} = o_{i,j} + X$ où X est une valeur aléatoire tirée selon une une loi normale centrée sur 0 écart-type ϵ. Ici, ϵ est un paramètre du système.

Ensuite, chaque témoignage bruité $\tilde{o}_{i,j}$ d'un agent a_i envers un agent a_j est mis en attente dans une liste $O_{i,j}$. Lorsque la taille de η_1 listes de témoignages en attente franchit un seuil η_2, alors un ensemble de témoignages en attente sur d'autres listes sont validés. Cela va concerner les η_3 plus anciens témoignages de η_4 listes ayant atteint ce seuil. Ici, $\eta_4 \leq \eta_1$ et $\eta_3 \leq \eta_2$, et les η_k sont tous tirés uniformément pour chaque témoignage dans des intervalles définis en tant que paramètres du système. Ainsi, si un agent participe à plusieurs transactions simultanément, chaque témoignage le concernant sera diffusé avec un délai distinct.

Enfin, lorsqu'un agent a_k demande à l'autorité centrale la réputation d'un agent, cette dernière la calcule soit avec la fonction BetaReputation, soit avec EigenTrust. Comme vu en section 2.2, ces fonctions s'appuient sur un couple $(r_{i,j}, s_{i,j})$ de "bonnes" et "mauvaises" transactions. Pour construire ce couple, l'autorité centrale utilise la méthode préconisée par Jøsang et Ismail [9] : chaque témoignage bruité valide $o_{i,j}$ donne un $(\tilde{r}_{i,j}, \tilde{s}_{i,j})$ comme suit et tous les couples $(\tilde{r}_{i,j}, \tilde{s}_{i,j})$ valides sont sommés pour être ensuite utilisé dans le calcul de la réputation.

$$\tilde{r}_{i,j} = \frac{1 + \tilde{o}^t_{i,j}}{2} \qquad \tilde{s}_{i,j} = \frac{1 - \tilde{o}^t_{i,j}}{2}$$

4 Expérimentations

Nous expérimentons dans cette section nos méthodes de confidentialisation en simulation. Nous considérons les fonctions de réputation BetaReputation et EigenTrust ainsi que les politiques de sélection ϵ-gloutonne, β-softmax et UCB. Chaque expérimentation est composée de 10 simulations de 1000 pas de temps durant lesquel 50 agents [5] sélectionnent, interagissent avec et évaluent un autre agent.

4.1 Paramètres expérimentaux

Les transactions effectuées par chaque agent varient en qualité. Cette qualité est une vérité terrain non accessible aux agents. La qualité f_j d'un agent a_j est définie par une espérance et un écart-type tirés uniformément respectivement dans $[-1, 1]$ et $[0, 0.5]$. Lors d'une transaction, la qualité observée par un agent a_i est tiré aléatoirement à partir d'une loi normale paramétrée par la qualité de l'agent a_j sollicité. Cette qualité observée correspond aux témoignages $o^t_{i,j}$ évoqués dans la section précédente. La table 1 récapitule les paramètres de bruit et de délai considérés dans nos expérimentations. Les courbes noires sur les figures pages 7, 8 et 9 indiquent les résultats sans mécanisme de confidentialisation, représentant donc le comportement nominal des fonctions de réputation étudiés.

4.2 Mesures de performance

Afin de mesurer l'influence des procédures d'anonymisation sur les systèmes de réputation, nous considérons deux métriques : une distance entre les valeurs de réputations calculées et la fiabilité réelle des agents – vérifiant que la qualité de l'évalution n'est pas perturbée – et une mesure de regret – vérifiant que la politique de sélection ne l'est pas non plus.

5. Si le nombre d'agents a bien entendu une influence sur les performances absolues des systèmes de réputation, nous avons observé qu'augmenter le nombre d'agents ou le diminuer ne modifie pas la forme générale des résultats, ni leurs performances relatives.

Écart-type ϵ du bruit	Paramètres η_k du délai	Couleur des courbes
–	–	noir
0.2	$(3, 5, 2, 3)$	bleu
0.5	$(4, 6, 3, 4)$	rouge
1.0	$(5, 7, 4, 5)$	vert
2.0	$(6, 8, 5, 6)$	marron

TABLE 1 – Récapitulatif des paramètres d'expérimentation

La distance est la *distance moyenne de Kentall-tau* entre deux fonctions de rang [12]. Cette mesure calcule le nombre de *paires discordantes* [6] entre réputation des agents et fiabilité des agents, divisant ensuite cette somme par le nombre de paires d'agents. Le *regret*, quant à lui, est la différence entre le gain qu'un agent aurait obtenu s'il avait interagit avec l'agent qui avait la meilleure réputation et le gain qu'il a effectivement obtenu en intergissant avec l'agent qui a été sélectionné [3, 21].

4.3 Résultats sur BetaReputation

Sur BetaReputation, quelle que soit la politique de sélection, le bruit et le délai pris séparément perturbent les performances comme attendu. Le bruit provoque une augmentation de la distance et du regret (voir figures 2 et 3), et le délai provoque un décalage dans la convergence de la distance et du délai (voir figures 4 et 5). Plus le bruit et le délai sont importants, plus cette perturbation est forte. Nous pouvons remarquer un effet contre-intuitif concernant l'effet du délai sur le regret de la politique UCB (voir figure 5.c) : le délai fait converger plus rapidement le regret et, plus le délai est important, plus l'augmentation du regret est faible. Ceci s'explique par le fait que le terme d'exploration de la politique UCB est facteur du nombre de témoignages.

Avec le délai, le terme d'exploitation prend le pas et le terme d'exploration n'arrive pas à le rattraper. La politique UCB devient alors proche d'une politique 0-gloutonne. La combinaison du bruit et du délai provoque naturellement à la fois une augmentation de la distance et du regret, ainsi qu'un décalage de leur convergence. Si les effets sur le regret sont assez semblables à ceux du délai seul, et en particulier pour la politique UCB, la figure 6.b montre que plus le bruit et le délai sont importants, plus vite le sysème converge vers une évaluation correcte des agents pour la politique β-softmax.

6. Deux agents a_i et a_j forment une paire discordante pour l'agent a_k si, et seulement si, $Rep_i^k > Rep_j^k$ et $f_i < f_j$.

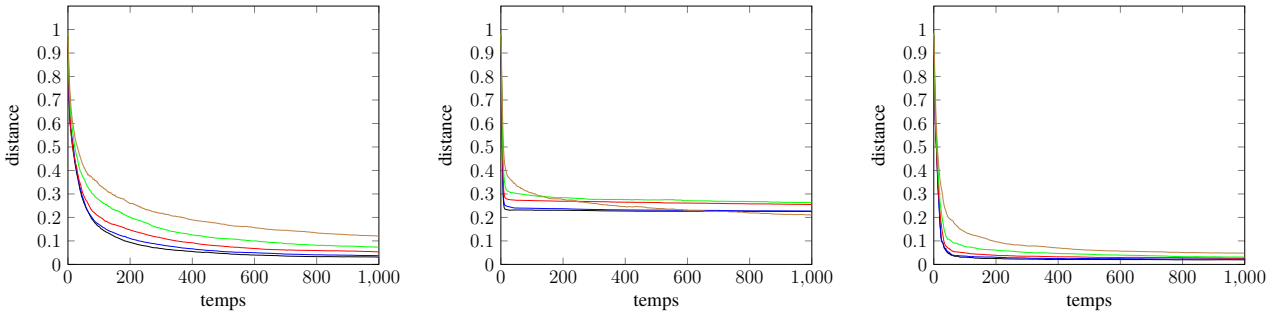
4.4 Résultats sur EigenTrust

Sur EigenTrust, le bruit et le délai ont des effets similaires mais plus contrastés à ceux sur Beta-Reputation. Dans certains cas, des effets contre-intuitifs ont lieu. En effet, EigenTrust est une fonction par rang et les valeurs de réputation des agents sont très proches les unes des autres. Cela conduit les politiques β-softmax et UCB à des comportements particuliers : sans un facteur β élevé la politique β-softmax se rapproche d'un tirage uniforme, et la politique UCB oscille car le facteur d'exploration domine périodiquement.

Les figures 9.a et 9.b montrent qu'un bruit modéré permet d'obtenir une meilleure évaluation des agents et, donc, un regret inférieur au regret sans confidentialisation. Comme les valeurs de réputation d'EigenTrust sont proches les unes des autres, de petites perturbations uniformes de ces valeurs ont pour effet de les disperser et de permettre une meilleure discrimination des agents. Enfin, pour les mêmes raisons que sur BetaReputation, un délai permet à la politique UCB, qui normalement oscille sans être efficace, de converger (voir figures 11.c et 13.c) . En effet, le délai diffuse les témoignages par paquets et une arrivée massive de témoignages valides discrimine subitement les agents, évitant au facteur d'exploration de dominer.

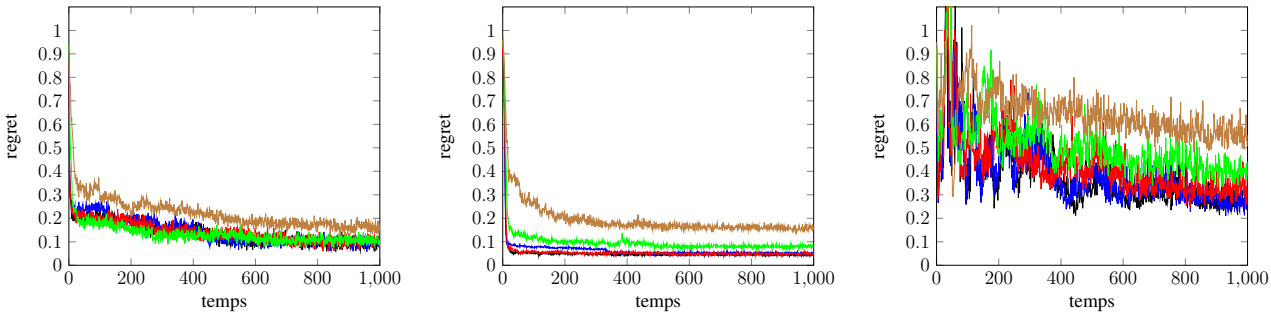
5 Conclusion

Pour conclure, les expérimentations montrent que la confidentialisation fondée sur un bruit additif et un délai de diffusion par agent peut être utilisable sur des systèmes de réputation de type BetaReputation à condition que les agents utilisent une politique de sélection de type ϵ-gloutonne. En effet, sur ce type de politique les effets du bruit et du délai sont les plus faibles et les plus attendus. De manière intéressante, le bruit et le délai peuvent être également utilisés sur EigenTrust pour améliorer les politiques ϵ-gloutonne et β-softmax ainsi que "régulariser" le comportement d'une politique UCB.



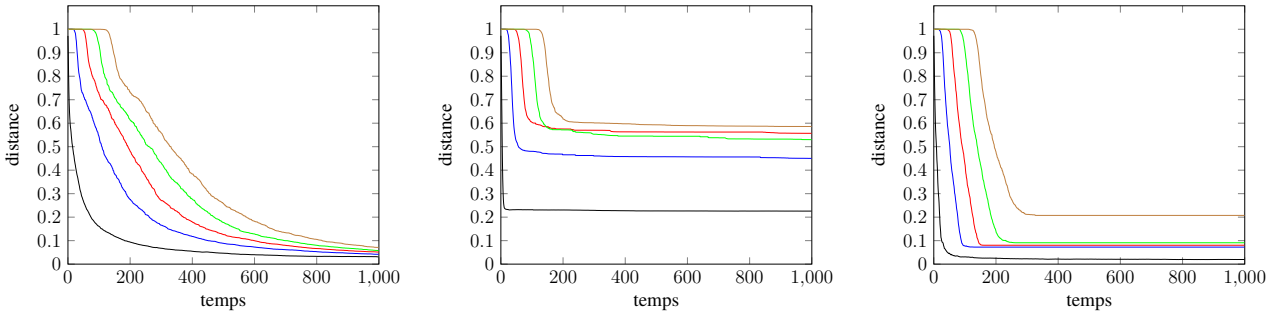
(a) ϵ-gloutonne (b) β-softmax (c) UCB

FIGURE 2 – Impact du bruit sur la distance pour BetaReputation



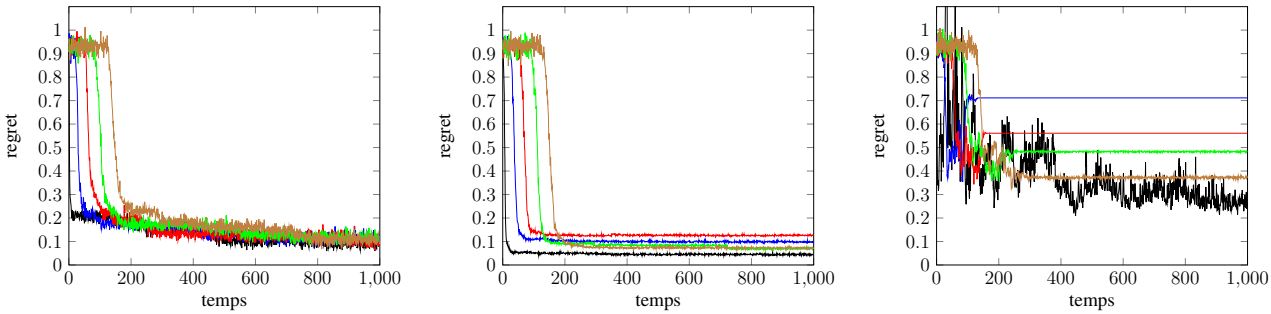
(a) ϵ-gloutonne (b) β-softmax (c) UCB

FIGURE 3 – Impact du bruit sur le regret pour BetaReputation



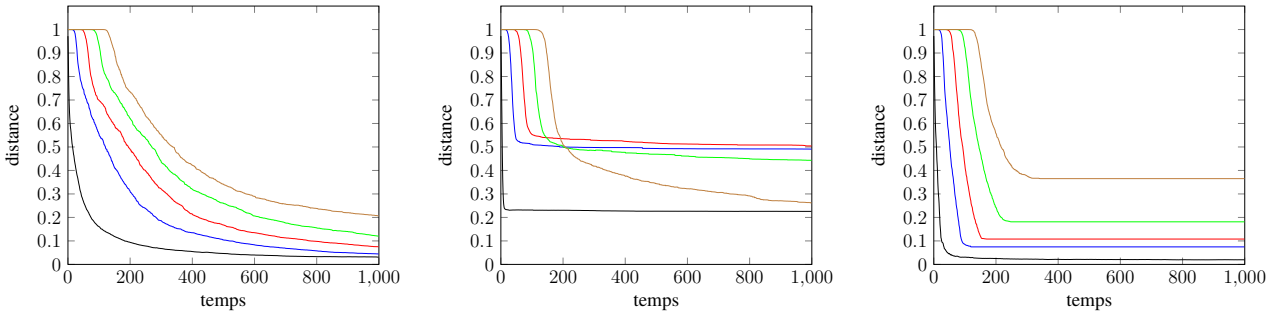
(a) ϵ-gloutonne (b) β-softmax (c) UCB

FIGURE 4 – Impact du délai sur la distance pour BetaReputation



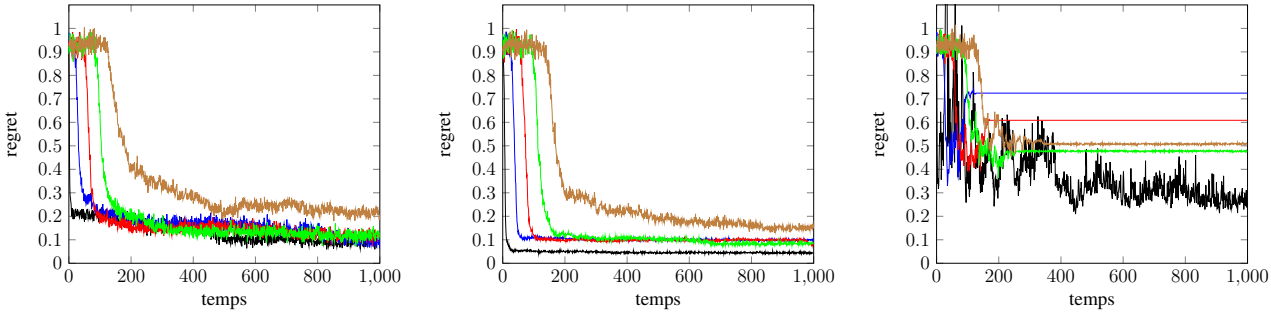
(a) ϵ-gloutonne (b) β-softmax (c) UCB

FIGURE 5 – Impact du délai sur le regret pour BetaReputation



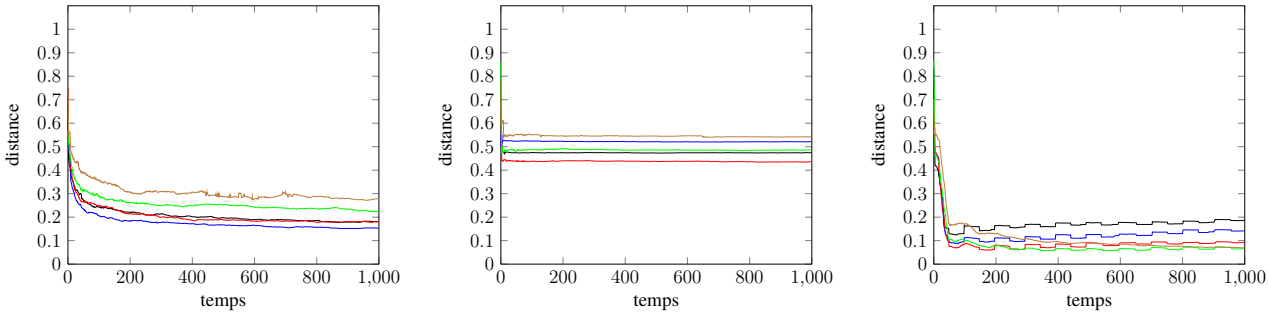
(a) ϵ-gloutonne (b) β-softmax (c) UCB

FIGURE 6 – Impact du bruit et du délai sur la distance pour BetaReputation



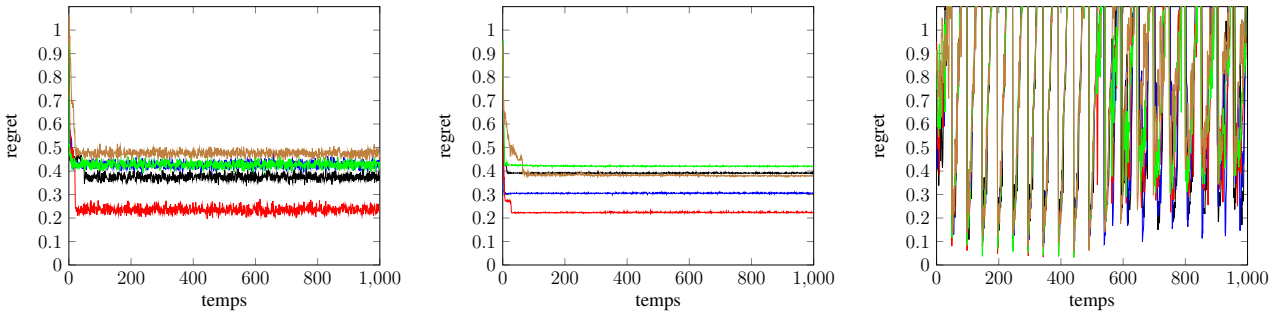
(a) ϵ-gloutonne (b) β-softmax (c) UCB

FIGURE 7 – Impact du bruit et du délai sur le regret pour BetaReputation



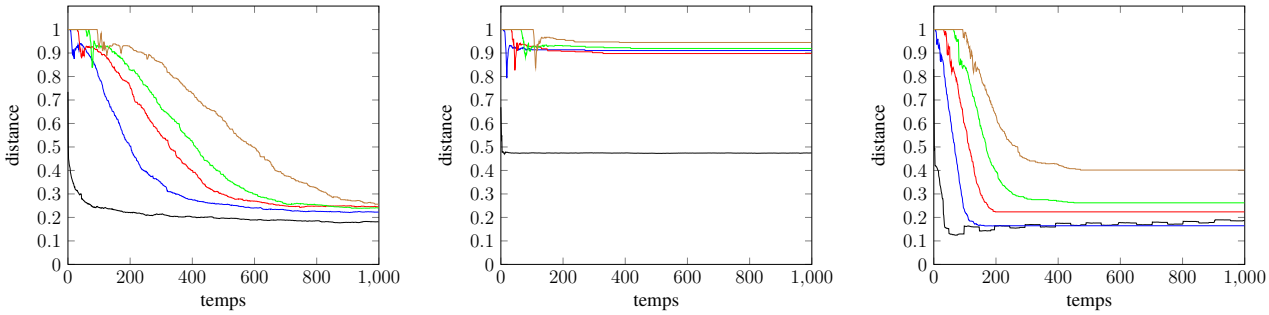
(a) ϵ-gloutonne (b) β-softmax (c) UCB

FIGURE 8 – Impact du bruit sur la distance pour EigenTrust



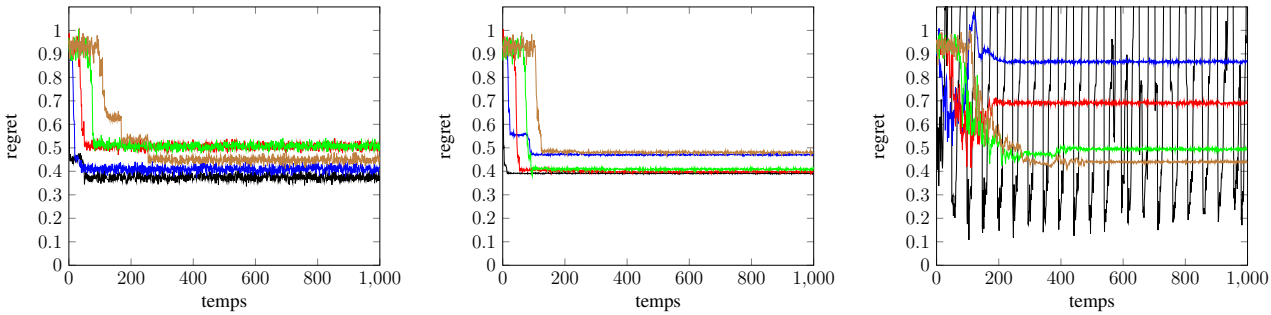
(a) ϵ-gloutonne (b) β-softmax (c) UCB

FIGURE 9 – Impact du bruit sur le regret pour EigenTrust



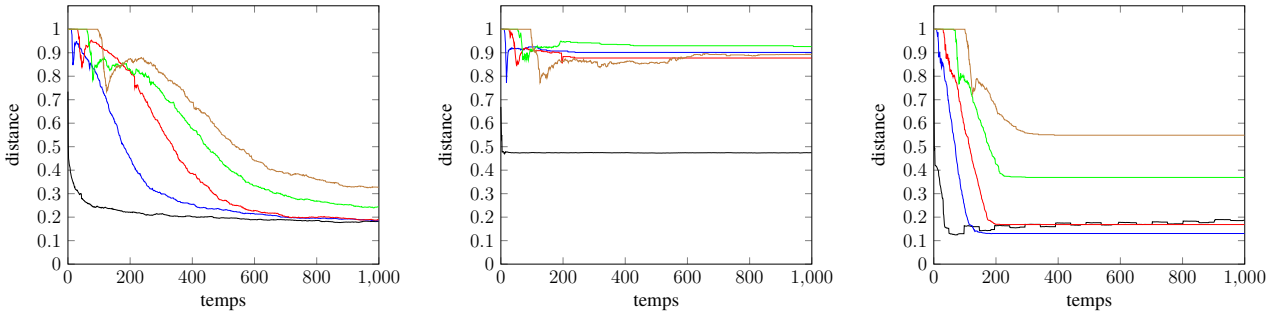
(a) ϵ-gloutonne (b) β-softmax (c) UCB

FIGURE 10 – Impact du délai sur la distance pour EigenTrust



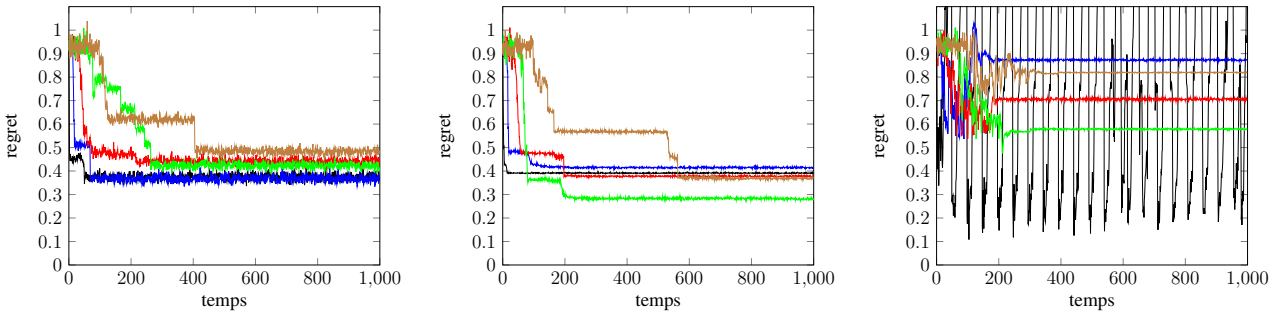
(a) ϵ-gloutonne (b) β-softmax (c) UCB

FIGURE 11 – Impact du délai sur le regret pour EigenTrust



(a) ϵ-gloutonne (b) β-softmax (c) UCB

FIGURE 12 – Impact du bruit et du délai sur la distance pour EigenTrust



(a) ϵ-gloutonne (b) β-softmax (c) UCB

FIGURE 13 – Impact du bruit et du délai sur le regret pour EigenTrust

Références

[1] Charu C. Aggarwal and Philip S. Yu. A general survey of privacy-preserving data mining models and algorithms. In *Privacy-Preserving Data Mining*, pages 11–52. Springer, 2008.

[2] Roberto Aringhieri, Ernesto Damiani, Sabine De Capitani Di Vimercati, Stefano Paraboschi, and Pierangelo Samarati. Fuzzy techniques for trust and reputation management in anonymous peer-to-peer systems. *Journal of the American Society for Information Science and Technology*, 57(4) :528–537, 2006.

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

[4] Javier Carbo, Jose M Molina, and Jorge Davila. Comparing predictions of SPORAS vs. a fuzzy reputation system. In *3rd International Conference on Fuzzy Sets and Fuzzy Systems*, volume 200, pages 147–153, 2002.

[5] Cynthia Dwork. Differential privacy. In *33rd International Colloquium on Automata, Languages and Programming*, pages 1–12, 2006.

[6] Tyrone Grandison and Morris Sloman. A survey of trust in Internet applications. *Communications Surveys & Tutorials, IEEE*, 3(4) :2–16, 2000.

[7] Omar Hasan, Lionel Brunie, and Elisa Bertino. Preserving privacy of feedback providers in decentralized reputation systems. *Computers and Security*, 31(7) :816 – 826, 2012.

[8] Kuan L. Huang, Salil S. Kanhere, and Wen Hu. A privacy-preserving reputation system for participatory sensing. In *37th Annual IEEE Conference on Local Computer Networks*, pages 10–18, 2012.

[9] Audun Jøsang and Roslan Ismail. The beta reputation system. In *15th Bled Conference on Electronic Commerce*, pages 324–337, 2002.

[10] Audun Jøsang, Roslan Ismail, and Colin Boyd. A survey of trust and reputation systems for online service provision. *Decision support systems*, 43(2) :618–644, 2007.

[11] Sepandar D. Kamvar, Mario T. Schlosser, and Hector Garcia-Molina. The eigentrust algorithm for reputation management in P2P networks. In *12th International Conference on World Wide Web*, pages 640–651, 2003.

[12] Maurice Kendall. A new measure of rank correlation. *Biometrika*, 30 :81–89, 1938.

[13] Ross A. Malaga. *Information Systems Research Methods, Epistemology, and Applications*, chapter The retaliatory feedback problem : evidence from eBay and a proposed solution, pages 342–349. Hershey, 2009.

[14] Zaki Malik and Athman Bouguettaya. Rateweb : Reputation assessment for trust establishment among web services. *International Journal on Very Large Data Bases*, 18(4) :885–911, 2009.

[15] Sergio Marti and Hector Garcia-Molina. Taxonomy of trust : Categorizing P2P reputation systems. *Computer Networks*, 50(4) :472–484, 2006.

[16] Kato Mivule. Utilizing noise addition for data privacy, an overview. *CoRR*, 2013.

[17] Xinlei Oscar Wang, Wei Cheng, Prasant Mohapatra, and Tarek Abdelzaher. ARTsense : Anonymous reputation and trust in participatory sensing. In *32nd International Conference on Computer Communications*, pages 2517–2525, 2013.

[18] Jordi Sabater, Mario Paolucci, and Rosaria Conte. Repage : Reputation and image among limited autonomous partners. *Journal of Artificial Societies and Social Simulation*, 9(2), 2006.

[19] Jordi Sabater and Carles Sierra. Review on computational trust and reputation models. *Artificial Intelligence Review*, 24(1) :33–60, 2005.

[20] Aameek Singh and Ling Liu. Trustme : Anonymous management of trust relationships in decentralized P2P systems. In *International Conference on Peer-to-Peer Computing*, page 1, 2003.

[21] Thibaut Vallée, Grégory Bonnet, and François Bourdon. Multi-armed bandit policies for reputation systems. In *13th International Conference on Practical Applications of Agents and Multi-Agent Systems*, pages 279–290, 2014.

[22] Runfang Zhou and Kai Hwang. Powertrust : A robust and scalable reputation system for trusted peer-to-peer computing. *IEEE Transactions on Parallel and Distributed Systems*, 18(4) :460–473, 2007.