Grégory Bonnet, Laurent Vercouter, Damien Lelerre

Confidentialité dans les systèmes de réputation

Volume 3, nº 5-6 (2022), p. 671-689.

DOI not yet assigned

© Les auteurs, 2022.

Cet article est diffusé sous la licence
Creative Commons Attribution 4.0 International License.
http://creativecommons.org/licenses/by/4.0/

La Revue Ouverte d'Intelligence Artificielle est membre du Centre Mersenne pour l'édition scientifique ouverte
www.centre-mersenne.org
e-ISSN : pending

Revue Ouverte d'Intelligence Artificielle
Volume 3, n° 5-6, 2022, 671-689

Confidentialité dans les systèmes de réputation

Grégory Bonnet[a], Laurent Vercouter[a], Damien Lelerre[a]

[a] Normandie Université, UNICAEN, ENSICAEN, CNRS, GREYC, Caen, France
E-mail : gregory.bonnet@unicaen.fr, laurent.vercouter@insa-rouen.fr, damien.lelerre@unicaen.fr.

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 étudié 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. — Systèmes multi-agents, Systèmes de confiance et de réputation, Confidentialité.

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* [9, 14, 20, 24, 27, 30]. 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 [18]. 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 [3, 11, 22, 28]. Néanmoins, il reste possible de corréler le moment de la publication d'un témoignage anonyme avec celui d'une transaction pour identifier son auteur. Une solution consistant à ne plus rendre les témoignages publics n'est pas non plus satisfaisante car elle réduit la transparence du calcul des réputations et augmente la vulnérabilité face à des attaques telles que la production de faux témoignages et la collusion. Dans cet article, 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, notamment lorsque les agents ont des buts compétitifs ou lorsque la qualité du service est proportionnelle au coût pour le fournisseur. 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. Une hypothèse sous-jacente à un système de réputation est donc que les agents soient capables localement d'évaluer la qualité d'une transaction. Selon le système utilisé, l'évaluation peut être binaire (bonne ou mauvaise) ou graduée sur une échelle discrète ou continue.

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 [10, 14, 20, 24, 27]. 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 [6, 13, 15, 19, 26, 31] et ont été positionnées les unes par rapport aux autres [9, 24, 30]. Dans la suite de cet article, nous considérons les deux fonctions suivantes, à savoir BetaReputation [13] et EigenTrust [15], 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 mentionnées précédemment. Dans les deux cas, les agents évaluent les transactions comme un couple (r, s) où r est la part « positive » de la transaction et s la part négative. Le calcul de ces part est classiquement laissé à la discrétion du concepteur du système. Toutefois dans toute la suite de cet article, nous utilisons la méthode préconisée par Jøsang et Ismail [13]. Une transaction est évaluée par une valeur $v \in [-1, 1]$ puis (r, s) est calculé comme suit : $r = \frac{1+v}{2}$ et $s = \frac{1-v}{2}$.

[(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.

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_i représente 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 selon l'évaluation de a_i. Lors de la prise en compte d'une nouvelle transaction (r, s) à l'instant t, ces deux valeurs sont mises à jour avec un facteur d'oubli $\lambda \in [0, 1]$:

$$r^t_{i,j} = \lambda r^{t-1}_{i,j} + r \tag{2.1}$$

$$s^t_{i,j} = \lambda s^{t-1}_{i,j} + s \tag{2.2}$$

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 :

$$Rep^i_j = \frac{R^i_j - S^i_j}{R^i_j + S^i_j + 2} \tag{2.3}$$

où R^i_j (resp. S^i_j) représente la somme de toutes les transactions des agents a_k avec a_j évaluées positivement (resp. négativement) pondérées par la confiance que l'agent a_i a envers ces agents a_k. Formellement :

$$R^i_j = \sum_{a_k \notin \{a_i, a_j\}} \frac{2r_{i,k} \times r_{k,j}}{(s_{i,k} + 2)(r_{k,j} + s_{k,j} + 2) + 2r_{i,k}} \tag{2.4}$$

$$S^i_j = \sum_{a_k \notin \{a_i, a_j\}} \frac{2r_{i,k} \times s_{k,j}}{(s_{i,k} + 2)(r_{k,j} + s_{k,j} + 2) + 2r_{i,k}} \tag{2.5}$$

EigenTrust. EigenTrust est une fonction de réputation *globale* et *par rang* où chaque agent représente là-aussi 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. Contrairement à BetaReputation, EigenTrust n'utilise pas de facteur d'oubli. EigenTrust s'appuie ensuite sur une matrice de confiance normalisée C où chaque élément $c_{i,j}$ est défini comme suit :

$$c_{i,j} = \frac{\max(r_{i,j} - s_{i,j}, 0)}{\sum_{a_j \notin \{a_i\}} \max(r_{i,j} - s_{i,j}, 0)} \tag{2.6}$$

Dans le cas où le dénominateur de l'équation 2.6 est nul, $c_{i,j}$ est fixé à une valeur par défaut (typiquement 1 sur le nombre d'agents du système). 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

[(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.

un vecteur de réputation *a priori* :

$$Rep^{(t+1)} = (1 - \alpha) \times C^T \times Rep^{(t)} + \alpha \times \vec{p} \tag{2.7}$$

Il est à noter que, de manière générale, EigenTrust converge plus vite vers des réputations stables que BetaReputation. En effet, EigenTrust est un système de réputation global : tous les agents et leurs témoignages participent au calcul de la réputation de chacun (i.e. si deux agents a_i et a_j témoignent sur un agent a_k alors la réputation de a_k est calculée à partir de ces deux témoignages). BetaReputation est un système personnalisé : dans le même exemple, la réputation de a_k est calculée du point de vue d'un agent donné, supposons ici a_i et ne tient pas compte des témoignages formulés par des agents avec qui a_i n'a pas interagit, en l'occurence a_j. Ainsi, EigenTrust prend en compte en général plus de témoignages et converge alors plus vite.

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 [5, 29].

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, UCB (pour *Upper Confidence Bound*) choisit l'agent a_j qui maximise une valeur v_j avec :

$$v_j^i = Rep_j^i + \sqrt{\frac{2 \times \ln(1 + r_{i,j} + s_{i,j})}{1 + \Sigma_{a_k \in N}(r_{k,j} + s_{k,j})}} \tag{2.8}$$

Cette valeur v_j^i est calculée à chaque fois qu'un agent a_i doit décider avec qui interagir. Elle est composée d'un premier terme d'exploitation indiquant l'intérêt à choisir l'agent en fonction de sa réputation et d'un second terme d'exploration qui donne de l'intérêt aux agents qui n'ont reçu que peu d'observations comparativement aux autres. Cela permet alors une exploration contextuelle qui dépend de la quantité d'information dont le décideur dispose sur les autres. En effet, tant que le décideur dispose de peu d'information, le second terme domine le premier et conduit l'agent à

sélectionner ceux sur lesquels il a le moin 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^i_j tirée selon une fonction exponentielle normalisée :

$$p^i_j = \frac{e^{\beta.Rep^i_j}}{\Sigma_{a_k \in N} e^{\beta.Rep^i_j}} \tag{2.9}$$

Le paramètre $\beta \in \mathbb{R}$ est une température inverse. Si $\beta = 0$, la distribution de probabilité est une 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. Bien que l'anonymisation soit nécessaire, elle n'est cependant pas suffisante pour assurer la confidentialité 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.

Intuitivement, anonymiser et confidentialiser un système de réputation est un problème analogue à anonymiser et confidentialiser une base de données. En effet, les témoignages des agents sont des éléments d'information qui peuvent être enregistrés dans une base centralisée ou distribuée entre les agents. Dans la littérature concernant les systèmes de réputation, de nombreux travaux ont proposé d'anonymiser les témoignages en se fondant sur des protocoles cryptographiques pour s'assurer de la non-répudiation des témoignages [3, 11, 22, 28]. Toutefois, comme pour les bases de données, ceci ne permet pas de garantir la confidentialité des témoignages. À moins de restrictions particulières du système, un agent, sachant les interactions qu'il a eu avec d'autres agents, peut observer la dynamique de sa réputation et en déduire les témoignages qui ont pu être formulés envers lui. Une étape de confidentialisation qui va plus loin que la simple anonymisation est donc aussi nécessaire.

Quelques travaux se sont déjà intéressés à la confidentialisation des témoignages pour un système de réputation. Hasan *et al.* [11] indiquent qu'une solution consisterait à bruiter les témoignages afin de rendre plus difficile leur déduction à partir de la

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

[4] *Privacy preservation*.

dynamique de la réputation. Toutefois, les auteurs se sont bornés à ce constat. De manière intéressante, Huang *et al.* [12] 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.

3.1. Bruit sur les témoignages

L'ajout de bruit est une méthode courante pour confidentialiser une base de données [8, 21]. 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 [8] 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.

Restent alors trois méthodes de bruitage classiques qui sont l'ajout de bruit additif, multiplicatif et logarithmique [21]. Ces méthodes consistent à transformer une donnée originale Y en une donnée Z telle que, respectivement, $Z = Y + \epsilon$, $Z = Y \times \epsilon$ et $Z = \ln(Y \times e^{\epsilon})$. Ici, ϵ est une valeur aléatoire tirée selon une une loi normale, centrée sur 0 pour le bruit additif et centrée sur 1 pour les deux autres méthodes. L'intérêt de chaque méthode dépend fortement du type de données considérées. Toutefois, comme Hasan *et al.*, nous optons dans la suite de cet article pour un *bruit additif* car, pour les autres méthodes, une valeur nulle pour Y pose problème (soit parce qu'il s'agit d'un élément absorbant, soit parce que la fonction de bruit n'est plus définie). En effet, bien que les autres méthodes puissent être pertinentes, la méthode du bruit additif nous permet de ne pas faire d'hypothèse sur les systèmes de réputation que nous étudions, en restreignant par exemple les valeurs de réputation autorisées.

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.

Proposition 3.1.
La probabilité p_{conf} que deux témoignages valides générés respectivement à t_1 et t_2 soient confondus est :

$$p_{conf} = \max\left(0, \left(\frac{(max - min) - |t_2 - t_1|}{(max - min)}\right)^2\right) \tag{3.1}$$

où max et min sont respectivement les bornes inférieures et supérieures de l'intervalle dans lequel est tiré le délai aléatoire ϵ.

Démonstration. Dans le délai temporel, un témoignage arrivé à l'instant t est validé à l'instant $t + \epsilon$ où ϵ est tiré uniformément entre *min* et *max* où $min \leqslant max$. Ce témoignage a donc un *intervalle de validation* $[t + min; t + max]$. Comme illustré

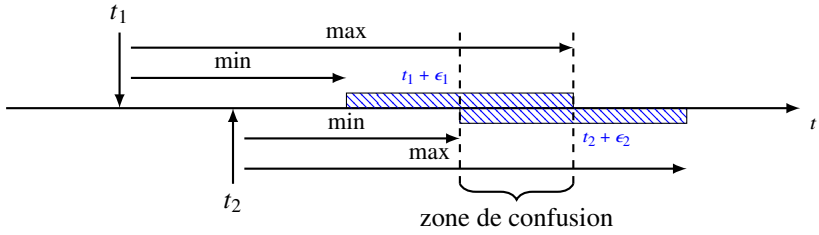


Figure 3.1 – Illustration des intervalles de validation et des zones de confusion

dans la figure 3.1, lorsque deux témoignages sont générés aux instants t_1 et t_2, leurs intervalles de validation peuvent s'intersecter. Nous nommons cette intersection *zone de confusion*. Nous avons alors deux cas de figure :

(1) Si l'un des deux témoignages est validé en dehors de la zone de confusion (ce qui est toujours le cas si cette intersection est vide), alors il est possible de déterminer de quel témoignage il s'agit en connaissant les dates de génération des témoignages et les valeurs *min* et *max*. Par élimination, il est donc possible d'identifier également le second témoignage même si celui-ci est validé dans la zone de confusion.
(2) Si les deux témoignages sont validés dans la zone de confusion, il est impossible de dire quelle validation correspond à quel témoignage. Ainsi les deux témoignages peuvent être pris l'un pour l'autre.

La zone de confusion entre deux témoignages datés t_1 et t_2 a pour taille $max(0, (max - min) - |t_1 - t_2|)$. Comme la durée du délai ϵ est tirée de façon uniforme, la probabilité qu'un témoignage généré à l'instant t_1 soit validé dans sa zone de confusion avec un témoignage généré à l'instant t_2 est :

$$max\left(0, \frac{(max - min) - |t_2 - t_1|}{(max - min)}\right) \tag{3.2}$$

Comme il est nécessaire que les deux témoignages soient validés dans la zone de confusion pour qu'ils puissent être confondus, la probabilité que le témoignage daté t_2 soit validé dans la zone de confusion est la même que pour le témoignage daté t_1, et ces deux événements sont indépendants. Ainsi, la probabilité que deux témoignages puissent être confondus est :

$$p_{\text{conf}} = max\left(0, \left(\frac{(max - min) - |t_2 - t_1|}{(max - min)}\right)^2\right) \tag{3.3}$$

□

Remarquons que pour une même durée entre deux témoignages, la probabilité de non-confusion est inversement proportionnelle au carré de la taille de l'intervalle de validation. 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 témoignage peut rester indéfiniment en attente.

3.3. Un système de réputation confidentiel

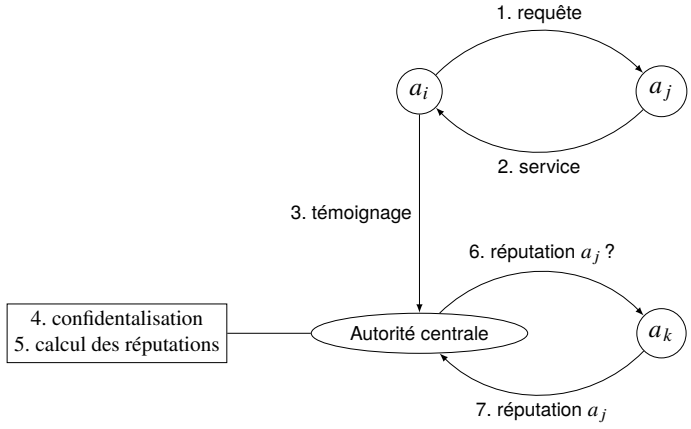


Figure 3.2 – Protocole de confidentialisation

Le système de réputation à témoignage confidentiel que nous proposons est illustré en figure 3.2. 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.

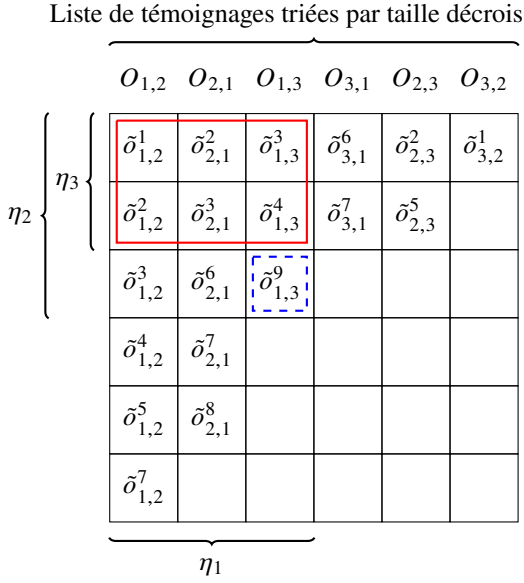


FIGURE 3.3 – Ensemble de témoignages bruités éligibles pour validation

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 $O_{i,j}$. L'autorité centrale validera ensuite ces témoignages par blocs. 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, $\eta_4 \leqslant \eta_1$ et $\eta_3 \leqslant \eta_2$. Les η_k sont tous tirés uniformément lors de chaque validation dans des intervalles définis en tant que paramètres du système. La figure 3.3 illustre ce mécanisme. Chaque colonne dans la figure correspond à une liste de témoignage d'un agent a_i envers un agent a_j. Ici, l'autorité centrale est paramétrée avec $\eta_1 = \eta_2 = 3$ et a tirée $\eta_3 = \eta_4 = 2$. Nous sommes au pas de temps de 9 et 3 agents ont déjà interagi. Lorsque le témoignage bruité $\tilde{o}^9_{1,3}$ (entouré en bleu sur la figure) est transmis à l'autorité centrale, les témoignages entourés en rouge sur la figure sont éligibles à la validation. L'autorité centrale tire au hasard deux listes parmi $O_{1,2}$, $O_{2,1}$ et $O_{1,3}$ et validera un

des ensembles de témoignages suivant : $\{\tilde{o}^1_{1,2}, \tilde{o}^2_{1,2}, \tilde{o}^2_{2,1}, \tilde{o}^3_{2,1}\}$, $\{\tilde{o}^1_{1,2}, \tilde{o}^2_{1,2}, \tilde{o}^3_{1,3}, \tilde{o}^4_{1,3}\}$ ou $\{\tilde{o}^2_{2,1}, \tilde{o}^3_{2,1}, \tilde{o}^3_{1,3}, \tilde{o}^4_{1,3}\}$.

Cette procédure rend peu probable, même pour un agent connaissant les paramètres du système, d'associer au hsard un témoignage avec une transaction, et de faire une association qui se révélerait correcte.

Proposition 3.2.

La probabilité pour un agent omniscient a d'associer correctement au hasard un témoignage reçu avec une transaction est :

$$p_{id} = \frac{1}{\eta_3.|\mathcal{A}^*|} \tag{3.4}$$

où $\mathcal{A}^*$ est l'ensemble des listes de témoignages envers a, éligibles à la validation.

Démonstration. Supposons un agent a ayant une connaissance parfaite de l'état de l'autorité centrale, c'est-à-dire connaissant la valeur courante des paramètres η_k et des témoignages en attente de validation pour l'ensemble des agents du système. Notons $\mathcal{A}^* = \{O_{i,a} : |O_{i,a}| \geqslant \eta_2\}$ l'ensemble de listes de témoignages envers l'agent a éligibles à la validation. Supposons qu'à un instant donné, l'agent a reçoive O^a témoignages. Il sait alors que ces témoignages viennent de $\frac{O^a}{\eta_4}$ listes distinctes tirées uniformément parmi $\mathcal{A}^*$. Ainsi, la probabilité qu'une liste $O_{i,a} \in \mathcal{A}^*$ ait été tirée sachant O^a est :

$$p(O_{i,a}|O^a) = \frac{|O^a|}{\eta_4.|\mathcal{A}^*|} \tag{3.5}$$

Sachant que quelles que soient les listes qui ont été tirées, le nombre de témoignages validés est le même pour chacune, la probabilité que l'agent sache de quelle liste provient un témoignage $o^a \in O^a$ donné est :

$$p(o^a \in O_{i,a} \mid O^a) = \frac{p(O_{i,a} \mid O^a)}{\sum\limits_{O_{j,a} \in \mathcal{A}^*} p(O_{j,a} \mid O^a)} = \frac{1}{|\mathcal{A}^*|} \tag{3.6}$$

Comme η_3 témoignages sont validés dans chaque liste, la probabilité que l'agent a associe correctement le témoignage est alors de :

$$p_{\text{id}} = \frac{1}{\eta_3.|\mathcal{A}^*|} \tag{3.7}$$

Ainsi, le seul cas où la probabilité de confusion du témoignage est nulle est lorsque $\eta_2 = 1$ et que l'agent n'interagit qu'avec un unique agent ($|\mathcal{A}^*| = 1$). □

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 [13] : chaque témoignage bruité

validé $\tilde{o}_{i,j}$ donne un $(\tilde{r}_{i,j}, \tilde{s}_{i,j})$ comme suit et tous les couples $(\tilde{r}_{i,j}, \tilde{s}_{i,j})$ validés sont sommés pour être ensuite utilisé dans le calcul de la réputation.

$$\tilde{r}_{i,j} = \frac{1 + \tilde{o}^t_{i,j}}{2} \tag{3.8}$$

$$\tilde{s}_{i,j} = \frac{1 - \tilde{o}^t_{i,j}}{2} \tag{3.9}$$

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 1 000 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 4.1 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.

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 4.1 – Récapitulatif des paramètres d'expérimentation

[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.

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.

La distance est la *distance moyenne de Kentall-tau* entre deux fonctions de rang [16]. 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 [5, 29].

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 4.1 et 4.2), et le délai provoque un décalage dans la convergence de la distance et du délai (voir figures 4.3 et 4.4). 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 4.4.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.

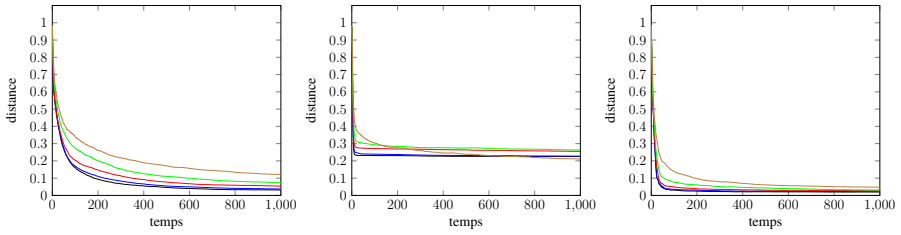
Avec le délai, le terme d'exploitation prend le pas et le terme d'exploration n'arrive pas à le rattraper. La politique UCB devient alors proche d'une politique 0-gloutonne. La combinaison du bruit et du délai provoque naturellement à la fois une augmentation de la distance et du regret, ainsi qu'un 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 4.5.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.

4.4. Résultats sur EigenTrust

Sur EigenTrust, le bruit et le délai ont des effets similaires mais plus contrastés à ceux sur BetaReputation. 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.

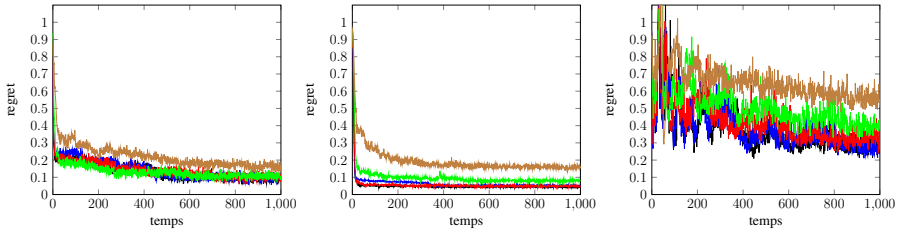
---

[6] Deux agents a_i et a_j forment une paire discordante pour a_k si $Rep_i^k > Rep_j^k$ et $f_i < f_j$ (où f_i (resp. f_j est la qualité de l'agent a_i (resp. a_j).



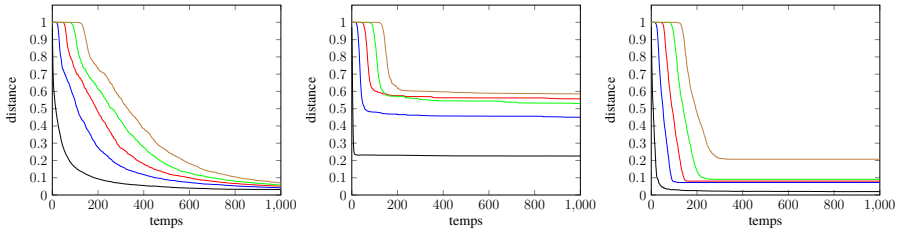
(a) ϵ-gloutonne (b) β-softmax (c) UCB

FIGURE 4.1 – Impact du bruit sur la distance pour BetaReputation



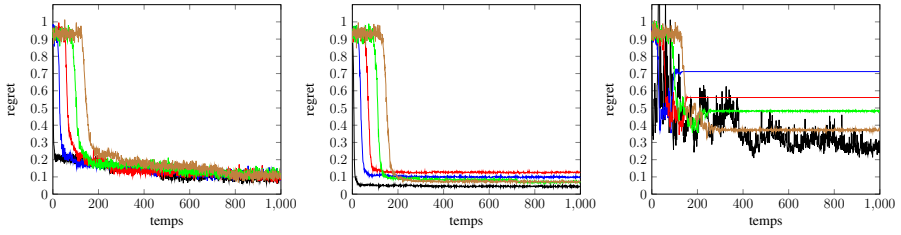
(a) ϵ-gloutonne (b) β-softmax (c) UCB

FIGURE 4.2 – Impact du bruit sur le regret pour BetaReputation



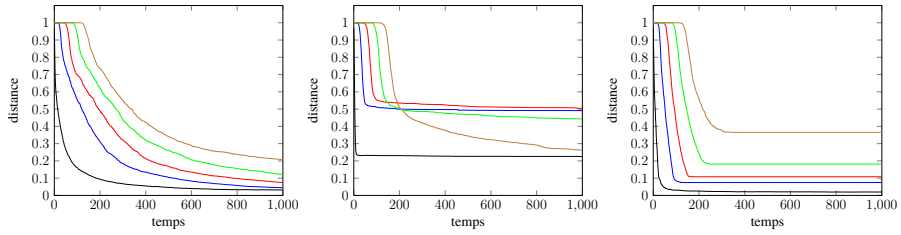
(a) ϵ-gloutonne (b) β-softmax (c) UCB

FIGURE 4.3 – Impact du délai sur la distance pour BetaReputation



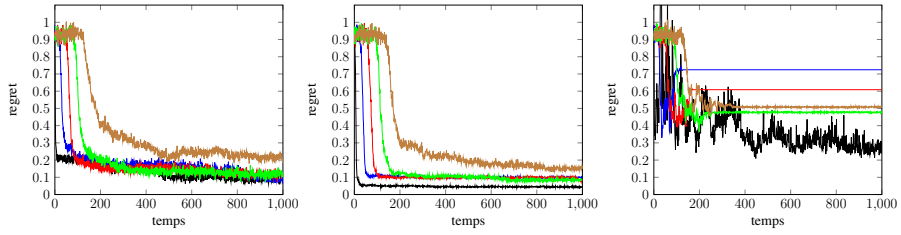
(a) ϵ-gloutonne (b) β-softmax (c) UCB

FIGURE 4.4 – Impact du délai sur le regret pour BetaReputation



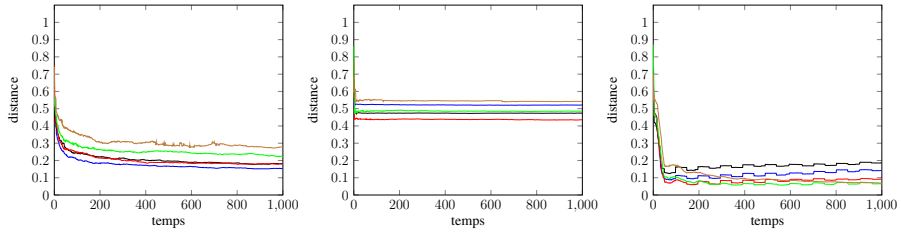
(a) ϵ-gloutonne (b) β-softmax (c) UCB

FIGURE 4.5 – Impact du bruit et du délai sur la distance pour BetaReputation



(a) ϵ-gloutonne (b) β-softmax (c) UCB

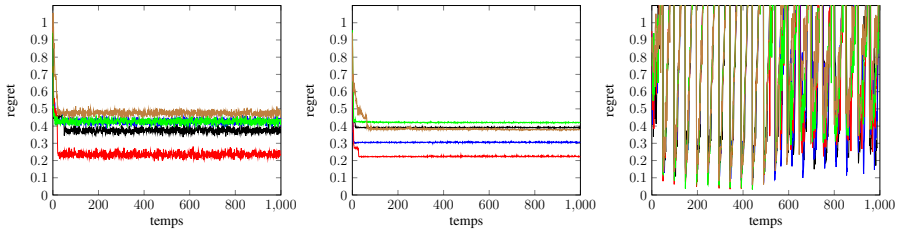
FIGURE 4.6 – Impact du bruit et du délai sur le regret pour BetaReputation



(a) ϵ-gloutonne (b) β-softmax (c) UCB

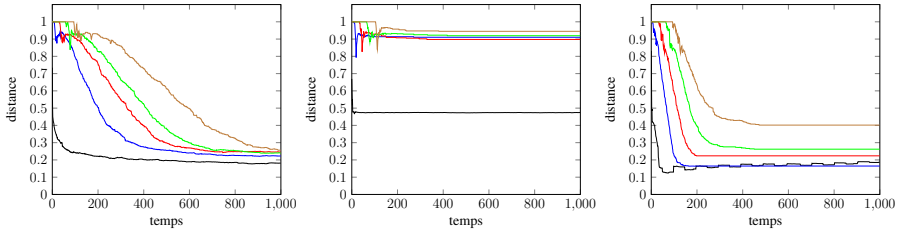
FIGURE 4.7 – Impact du bruit sur la distance pour EigenTrust

Les figures 4.8.a et 4.8.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 4.10.c et 4.12.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.



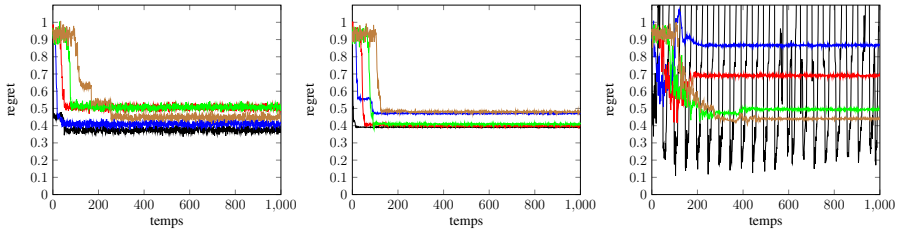
(a) ϵ-gloutonne (b) β-softmax (c) UCB

FIGURE 4.8 – Impact du bruit sur le regret pour EigenTrust



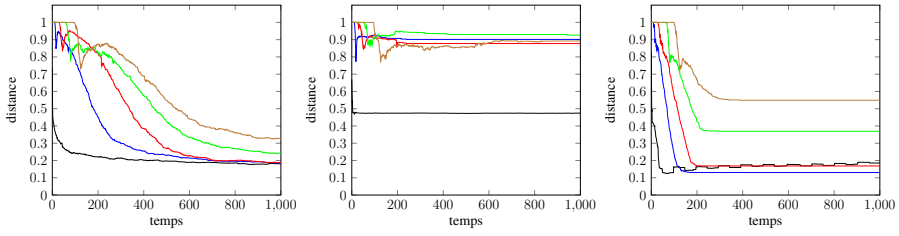
(a) ϵ-gloutonne (b) β-softmax (c) UCB

FIGURE 4.9 – Impact du délai sur la distance pour EigenTrust



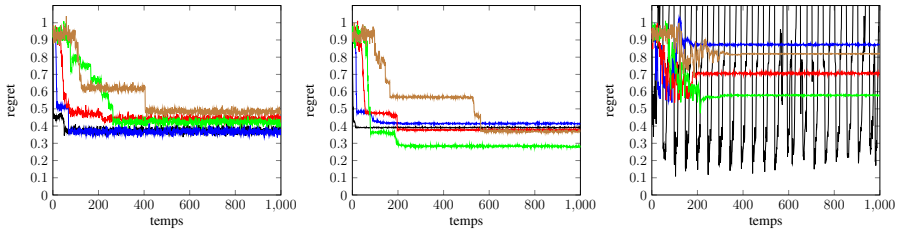
(a) ϵ-gloutonne (b) β-softmax (c) UCB

FIGURE 4.10 – Impact du délai sur le regret pour EigenTrust



(a) ϵ-gloutonne (b) β-softmax (c) UCB

FIGURE 4.11 – Impact du bruit et du délai sur la distance pour EigenTrust



(a) ϵ-gloutonne (b) β-softmax (c) UCB

FIGURE 4.12 – Impact du bruit et du délai sur le regret pour EigenTrust

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.

Étudier des variantes de notre mécanisme de confidentialisation, comme utiliser des bruits multiplicatifs ou logarthimiques, ou bien mélanger aléatoirement les listes de témoignages en attente de validation, ainsi que leurs effets sur d'autres systèmes de réputation est une perspective de recherche. Une seconde piste de recherche serait de réaliser une étude théorique pour caractériser l'impact du bruit et du délais au pire cas ou en moyenne. Par exemple, EigenTrust est une fonction de réputation qui repose sur le calcul des valeurs propres d'une matrice de confiance, et il serait intéressant d'analyser notre approche sous l'angle de la théorie des perturbations matricielles.

Parmi les autres perspectives à ce travail, nous envisageons d'étudier une version décentralisée où plusieurs agents maintiennent leur propre système local de réputation. En effet, nos expérimentations ont été menées dans cet article en considérant un système centralisé de réputation commun à l'ensemble des agents. La décentralisation apporte des bénéfices en terme d'efficacité, de passage à l'échelle et de robustesse. Mais elle nécessite également de prendre en compte l'hétérogénéite des valeurs de réputation, des fonctions de calcul et dans notre cas des mécanismes de confidentialité. Il sera alors nécessaire de synchroniser ces systèmes de réputation et d'étudier l'impact de l'ajout d'un bruit et d'un délai de diffusion qui peut être paramétré de manière différente au sein de chaque agent.

BIBLIOGRAPHIE

[1] C. C. AGGARWAL & P. S. YU, « A General Survey of Privacy-Preserving Data Mining Models and Algorithms », in *Privacy-Preserving Data Mining*, Springer, 2008, p. 11-52.

[2] E. Androulaki, S. G. Choi, S. M. Bellovin & T. Malkin, « Reputation Systems for Anonymous Networks », in *International Symposium on Privacy Enhancing Technologies Symposium*, 2008, p. 202-218.

[3] R. Aringhieri, E. Damiani, S. De Capitani Di Vimercati, S. Paraboschi & P. 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** (2006), nº 4, p. 528-537.

[4] D. Artz & Y. Gil, « A Survey of Trust in Computer Science and the Semantic Web », *Web Semantics : Science, Services and Agents on the World Wide Web* **5** (2007), nº 2, p. 58-71.

[5] P. Auer, N. Cesa-Bianchi & P. Fischer, « Finite-Time Analysis of the Multi-Armed Bandit Problem », *Machine Learning* **47** (2002), nº 2–3, p. 235-256.

[6] J. Carbo, J. M. Molina & J. Davila, « Comparing Predictions of SPORAS vs. a Fuzzy Reputation System », in *3rd International Conference on Fuzzy Sets and Fuzzy Systems*, vol. 200, 2002, p. 147-153.

[7] A. Cheng & E. Friedman, « Sybilproof Reputation Mechanisms », in *ACM SIGCOMM Workshop on Economics of Peer-to-peer Systems*, 2005, p. 128-132.

[8] C. Dwork, « Differential Privacy », in *33rd International Colloquium on Automata, Languages and Programming*, 2006, p. 1-12.

[9] J. Granatyr, V. Botelho, O. Lessing, E. E. Scalabrin, J.-P. Barthes & F. Enembreck, « Trust and Reputation Models for Multi-Agent Systems », *ACM Computing Surveys* **48** (2015), nº 2, article no. 27 (42 pages).

[10] T. Grandison & M. Sloman, « A Survey of Trust in Internet Applications », *Communications Surveys & Tutorials, IEEE* **3** (2000), nº 4, p. 2-16.

[11] O. Hasan, L. Brunie & E. Bertino, « Preserving Privacy of Feedback Providers in Decentralized Reputation Systems », *Computers and Security* **31** (2012), nº 7, p. 816-826.

[12] K. L. Huang, S. S. Kanhere & W. Hu, « A Privacy-Preserving Reputation System for Participatory Sensing », in *37th Annual IEEE Conference on Local Computer Networks*, 2012, p. 10-18.

[13] A. Jøsang & R. Ismail, « The Beta Reputation System », in *15th Bled Conference on Electronic Commerce*, 2002, p. 324-337.

[14] A. Jøsang, R. Ismail & C. Boyd, « A Survey of Trust and Reputation Systems for Online Service Provision », *Decision support systems* **43** (2007), nº 2, p. 618-644.

[15] S. D. Kamvar, M. T. Schlosser & H. Garcia-Molina, « The Eigentrust Algorithm for Reputation Management in P2P Networks », in *12th International Conference on World Wide Web*, 2003, p. 640-651.

[16] M. Kendall, « A New Measure of Rank Correlation », *Biometrika* **30** (1938), p. 81-89.

[17] J. J. Kim & W. E. Winkler, « Multiplicative Noise for Masking Continuous Data », Tech. report, Statistical Research Division, US Bureau of the Census, Washington D.C, 2003.

[18] R. A. Malaga, « Information Systems Research Methods, Epistemology, and Applications », chap. The Retaliatory Feedback Problem : Evidence from eBay and a Proposed Solution, p. 342-349, Hershey, 2009.

[19] Z. Malik & A. Bouguettaya, « Rateweb: Reputation Assessment for Trust Establishment Among Web Services », *International Journal on Very Large Data Bases* **18** (2009), nº 4, p. 885-911.

[20] S. Marti & H. Garcia-Molina, « Taxonomy of Trust: Categorizing P2P Reputation Systems », *Computer Networks* **50** (2006), nº 4, p. 472-484.

[21] K. Mivule, « Utilizing Noise Addition for Data Privacy, an Overview », *11th International Conference on Information and Knowledge Engineering* (2012), p. 65-71.

[22] X. Oscar Wang, W. Cheng, P. Mohapatra & T. Abdelzaher, « ARTSense: Anonymous Reputation and Trust in Participatory Sensing », in *32nd International Conference on Computer Communications*, 2013, p. 2517-2525.

[23] L. Page, S. Brin, R. Motwani & T. Winograd, « The PageRank Citation Ranking : Bringing Order to the Web », Tech. report, Stanford InfoLab, 1999.

[24] I. Pinyol & J. Sabater-Mir, « Computational Trust and Reputation Models for Open Multi-Agent Systems : A Review », *Artificial Intelligence Revue* **40** (2013), p. 1-25.

[25] D. B. Rubin, « Statistical Disclosure Limitation », *Journal of Official Statistics* **9** (1993), nº 2, p. 461-468.

[26] J. Sabater, M. Paolucci & R. Conte, « Repage: Reputation and Image Among Limited Autonomous Partners », *Journal of Artificial Societies and Social Simulation* **9** (2006), n° 2, p. 3.

[27] J. Sabater & C. Sierra, « Review on Computational Trust and Reputation Models », *Artificial Intelligence Review* **24** (2005), n° 1, p. 33-60.

[28] A. Singh & L. Liu, « TrustMe: Anonymous Management of Trust Relationships in Decentralized P2P Systems », in *International Conference on Peer-to-Peer Computing*, 2003, p. 142-149.

[29] T. Vallée, G. Bonnet & F. Bourdon, « Multi-Armed Bandit Policies for Reputation Systems », in *13th International Conference on Practical Applications of Agents and Multi-Agent Systems*, 2014, p. 279-290.

[30] H. Yu, Z. Shen, C. Leung, C. Miao & V. Lesser, « A Survey of Multi-Agent Trust Management Systems », *IEEE Access* **1** (2013), p. 35-50.

[31] R. Zhou & K. Hwang, « Powertrust: A Robust and Scalable Reputation System for Trusted Peer-to-Peer Computing », *IEEE Transactions on Parallel and Distributed Systems* **18** (2007), n° 4, p. 460-473.

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. — Multi-agent systems, Trust and reputation systems, Confidentiality.

Manuscrit reçu le 9 juillet 2021, révisé le 31 janvier 2022, accepté le 14 mars 2022.