La Croissance de l'Enchevêtrement dans la Cryptomonnaie IOTA

Quentin Bramas

La Croissance de l'Enchevêtrement dans la Cryptomonnaie IOTA

Quentin Bramas[1]

[1]*Laboratoire ICUBE, Université de Strasbourg, France*

IOTA fait partie des premières cryptomonnaies dont le fonctionnement est basé, non pas sur une chaine de blocs (blockchain), mais sur un graphe orienté acyclique (DAG), appelé *Tangle* (ou enchevêtrement), dont les nœuds sont les transactions et les arêtes représentent les confirmations. Cet article présente des résultats obtenus par simulation concernant la croissance du Tangle en fonction de plusieurs paramètres. Ces résultats permettent aussi de mieux comprendre les risques d'attaque de type double-dépense.

Mots-clefs : blockchain, sécurité, algorithmes distribués

1 Introduction

Les technologies « de livre de comptes distribué » (distributed ledger technology ou DLT) ont connu un interêt croissant ces dernières années, notamment depuis l'introduction en 2008 du protocole Bitcoin par Satoshi Nakamoto [Nak08], qui marque le début de leur utilisation à grande échelle. Elles sont au cœur de transactions financières qui représentent à ce jour 20 milliards de dollars par jour.

Depuis 2009, ces technologies ont petit à petit gagné le milieu de la recherche, notamment grâce à leurs liens avec les problèmes fondamentaux de l'algorithmique répartie, comme le problème du consensus en présence de pannes ou d'agents byzantins.

Plus récemment, d'autres technologies de type DLT font leur apparition. Cet article présente la technologie appelée enchevêtrement (ou Tangle) qui est à la base de l'implémentation de la cryptomonnaie IOTA [Pop16]. Le Tangle a la particularité de stocker les transactions dans un arbre orienté acyclique (DAG), et non dans une chaine de blocs, comme c'est le cas dans le protocole Bitcoin.

Travaux Connexes Le *white paper* [Pop16] qui introduit le Tangle n'est qu'une présentation rapide qui contient peu de résultats théoriques. Des simulations pour calculer le nombre moyen de transactions non confirmées ont été réalisées [Kuś16] et les résultats sont confirmés par une étude théorique que nous avons effectuée [Bra18] †. Cependant, ces simulations n'ont pas été réalisées avec les fonctions qui sont actuellement utilisées par la cryptomonnaie IOTA.

Par ailleurs, une partie du white paper est consacrée à l'étude des attaques possibles et à leur défense, mais aucune analyse théorique ou par simulation n'a été réalisée concernant la puissance nécessaire pour qu'un adversaire effectue une attaque.

Contributions Après avoir rappelé le fonctionnement du Tangle, nous présentons les résultats obtenus par simulation concernant le nombre d'extrémités (de transactions non-confirmées) au fil du temps. Notamment, les performances sur ce critère de la fonction utilisée par IOTA sont loin d'être optimales. Pour finir nous présentons la première analyse de la puissance nécessaire que doit déployer un adversaire afin de pouvoir attaquer le Tangle.

†. La version Française de cette étude [Bra18] a été soumise à AlgoTel 2018. Ce papier contient des similitudes avec le papier soumis à Algotel concernant la présentation du modèle Tangle, mais les résultats présentés ici ne sont présents ni dans la version d'AlgoTel ni dans le rapport technique [Bra18] et sont donc uniques.

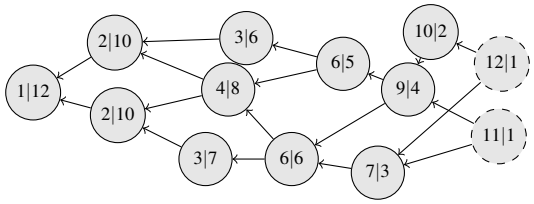

Figure 1: Un exemple de Tangle. Dans chaque site est inscrit son score et son poids cumulé. Les deux extrémités en pointillés ne sont confirmées par aucun site.

2 Modèle

On modélise l'ensemble des acteurs qui participent au protocole par un réseau de n nœuds complètement connectés. Le temps est discrétisé et à chaque instant, ou *ronde*, tous les nœuds s'activent de manière synchronisée. Ils communiquent par envoi de messages avec une latence de $h \geq 1$ ronde(s). Ainsi, à la ronde r, un nœud reçoit tous les messages envoyés par les autres nœuds à la ronde $r-h$. Chaque nœud peut générer une ou plusieurs transactions qu'il souhaite écrire dans la structure de données distribuée. Ce modèle, avec ces restrictions fortes, permet d'obtenir des résultats négatifs plus généraux, et servira de base pour l'étude des améliorations futures du protocole.

Les DLT permettent d'effectuer des opérations variées mais pour simplifier on ne considère que les transactions qui représentent un simple transfert de fond d'une adresse d'origine vers une adresse de destination.

Le DAG Les transactions sont stockées dans un graphe orienté acyclique (DAG) appelé *Tangle*. Chaque nœud stocke une copie locale du Tangle. Un nœud du Tangle, appelé *site*, représente une transaction et possède deux parents (potentiellement identiques) dans le Tangle. On dit qu'un site *confirme directement* ses parents et *confirme indirectement* ses autres ancêtres dans le DAG. Une *extrémité* du Tangle est un site qui n'a pas d'enfants i.e., qui n'est confirmé par aucun site. Le *genesis* est le seul site qui n'a pas de parents.

Pour inclure une transaction dans le Tangle, un nœud doit effectuer une preuve de travail i.e., générer une preuve de l'utilisation d'une certaine quantité de puissance de calcul. On appelle *poids*, cette quantité de travail, et on suppose que chaque site possède un poids de 1. Á partir de là, on appelle *poids cumulé*, resp. *score*, d'un site, la somme de son poids avec le poids de ces descendants (les sites qui le confirment), resp. la somme de son poids avec le poids de ces ancêtres (les sites qu'il confirme).

La figure 1 permet d'illustrer ces notations.

Algorithme de Sélection des Extrémités La sélection des parents, lors de l'ajout d'un site dans le Tangle, se fait en exécutant un *algorithme de sélection des extrémités* (ou *tip selection algorithm* ou TSA). Le TSA est un composant fondamental du protocole car c'est lui qui est responsable de la manière dont le Tangle se construit au fur et à mesure de l'ajout des sites. Principalement, le TSA doit choisir des sites qui n'entrent pas en conflit (voir Section 4) et qui n'ont pas encore d'enfants (i.e., des extrémités). Malgré tout, il est clair qu'un site peut avoir plusieurs enfants. En effet, les versions locales du Tangle pouvant être différentes au sein de deux nœuds distincts, le TSA pourrait choisir un site qui est une extrémité dans la version du Tangle local du nœud courant, mais ne pas être une extrémité dans la version locale d'un autre nœud.

Le papier d'introduction du Tangle [Pop16] présente trois TSA :

— TSA aléatoire : les deux parents sont choisis aléatoirement uniformément parmi toutes les extrémités du Tangle.

— Markov Chain Monte Carlo (MCMC) : plusieurs marches aléatoires sont exécutées à partir de certains sites vers les extrémités, en utilisant une fonction de transition qui dépend des poids cumulés des sites. Dans un site v, la probabilité de se déplacer vers un site u est donnée par la formule suivante :

$$p_{v,u} = \exp(-\alpha(w(v) - w(u))) / \sum_{c \in C_v} \exp(-\alpha(w(v) - w(c))$$

— MCMC Logarithmique (LMCMC) : similaire à MCMC mais avec la probabilité de transition sui-

vante :

$$p_{v,u} = (w(v) - w(u))^{-\alpha} / \sum_{c \in C_v} (w(v) - w(c))^{-\alpha}$$

Ce TSA est actuellement utilisé dans la cryptomonnaie IOTA avec le paramètre $\alpha = 3$.

3 Simulations

Chaque simulation consiste en une suite de rondes où à chaque ronde r le nombre de sites émis suit une loi de Poisson de paramètre λ. Pour chaque site, le TSA (random, MCMC ou LMCMC) est appelé afin de déterminer ses parents. Le TSA n'a connaissance que des sites générés aux rondes $r-h$ où h est la latence du réseau. On suppose $\lambda = 100$ et $h \in \{1,2\}$. Le cas $h = 1$ a déjà été réalisé [Kuś16] pour les TSA aléatoire et MCMC.

Les Figures 2 et 3 présentent le nombre d'extrémités à chaque ronde i.e., le nombre de sites qui ne sont pas encore confirmés.

Avec $h = 1$, on observe que, pour des valeurs de α plus petites que 0.1, les performances de l'algorithme LMCMC sont bonnes, c'est-à-dire que le nombre d'extrémités reste stable au fil du temps. En revanche l'algorithme MCMC, même avec de faibles valeurs de α ne permet pas d'obtenir des résultats convenables. Il faut remarquer que la valeur $\alpha = 3$ utilisée actuellement par IOTA avec l'algorithme LMCMC génère des résultats peu satisfaisant.

Avec $h = 2$, la Figure 3 montre des résultats similaires au cas $h = 1$, mais où le nombre d'extrémités est multiplié par un facteur légèrement inférieur à 2. L'intérêt ici est de montrer que les algorithmes aléatoire et LMCMC sont impactés de la même manière par l'augmentation de la latence.

Dans les deux cas ($h = 1$ ou 2) les résultats de l'algorithme aléatoire correspondent aux valeurs théoriques [Bra18].

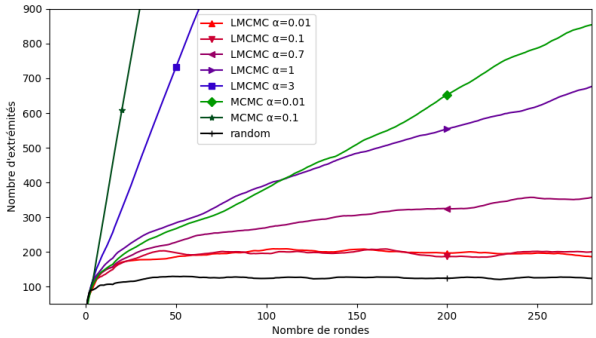

Figure 2: Nombre d'extrémités en fonction du nombre de rondes. $\lambda = 100$, $h = 1$.

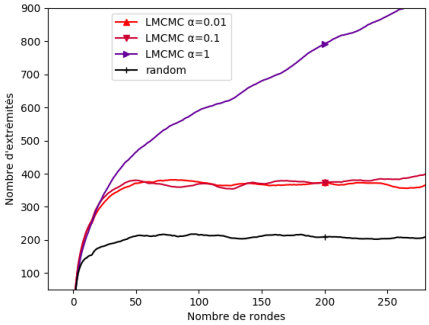

Figure 3: Nombre d'extrémités en fonction du nombre de rondes. $\lambda = 100$, $h = 2$.

4 Attaque de la Chaine Parasite

Une chaine parasite est un ensemble de sites reliés à un DAG. Généralement on parle de chaine parasite lorsque ces sites sont générés par un adversaire et entrent en conflit avec les sites du DAG principal. Par exemple, un adversaire peut commencer par émettre une transaction légitime pour acheter un bien ou un service, puis, après réception de son bien, émettre une chaine parasite contenant une transaction qui contredit sa première transaction (voir Figure 4. Si la chaine parasite est choisie par les nœuds honnêtes alors l'adversaire récupère son argent et le marchant n'est pas payé.

Pour notre étude on considère un réseau où chaque ronde, le nombre de site issus suit une loi de Poisson de paramètre $\lambda = 5$. Nous supposons une latence $h = 1$ et utilisons le TSA LMCMC avec $\alpha = 3$, similaire à celui utilisé dans IOTA. Pour simplifier on considère que l'attaque (et donc la simulation) commence sur le premier site du Tangle. Après avoir attendu un certain nombre R de rondes, on génère un nombre de sites adverses (les sites noirs de la Figure 4) tel que la probabilité qu'ils soient sélectionnés par le TSA est plus grande que $1/2$. On choisit ici d'adopter la stratégie suivante : chaque nœud adverse choisit comme parent

le nœud adverse précédent et le nœud cible du tangle. Cette méthode a la particularité à la fois de maximiser le nombre d'enfants adverses du site cible, et leur poids.

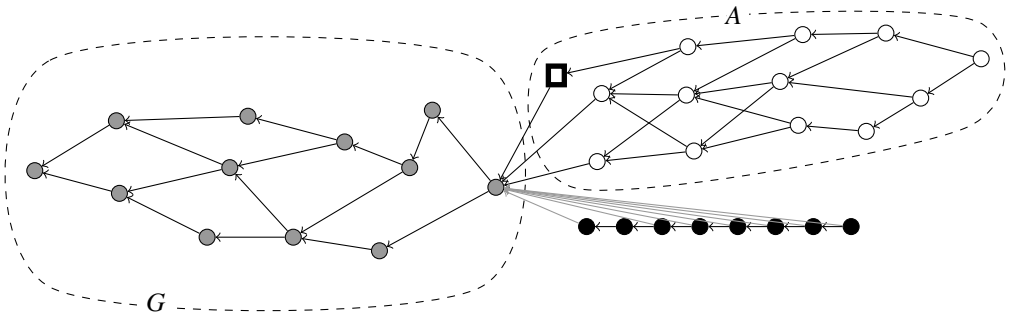

Figure 4: Une chaine parasite (nœuds noirs) est générée par un adversaire pour contredire une transaction (carré noir). Bien qu'elle contienne moins de transactions, la chaine parasite peut avoir l'avantage sur les transactions générées par les nœuds honnêtes A.

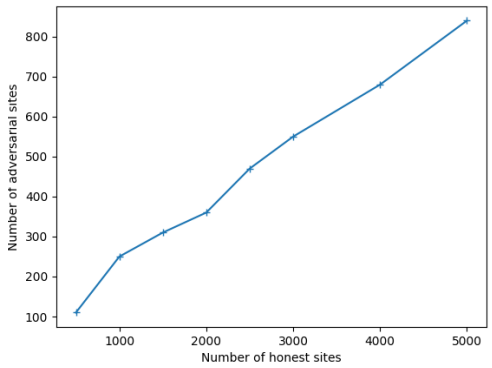

Figure 5: Nombre de nœuds que doit contenir la chaine parasites pour avoir une probabilité d'être sélectionnée plus grande que le DAG principal, par rapport au nombre de site dans le DAG principal.

La courbe de la Figure 5 montre une croissance linéaire, avec une pente de 0.15 environ, du nombre de sites adverses par rapport aux sites honnêtes. Ce qui signifie qu'avec seulement 15% de la capacité de hachage des nœuds honnêtes, un adversaire peut générer une chaine parasite qui a une plus grande probabilité d'être sélectionnée.

Conclusion Nous avons montré que les performances des algorithmes de sélections d'extrémités sont hétérogènes et que l'algorithme utilisé par IOTA est loin d'être l'optimal. De plus, la valeur élevée du paramètre $\alpha = 3$, ne permettait pas de protéger le Tangle contre les attaques de chaines parasites, puisque une faible fraction des sites permettrait à un attaquant de générer une chaine parasite. Il faut rappeler que ces attaques restent impossibles à mettre en place contre la version actuelle de IOTA, due à la présence d'un nœud spécial dans le réseau jouant le rôle de coordinateur et validant de manière centralisée les transactions.

Références

[Bra18] Quentin Bramas. The Stability and the Security of the Tangle. Research report, ICUBE, February 2018. URL : `https://hal.archives-ouvertes.fr/hal-01716111`.

[Kuś16] B. Kuśmierz. The first glance at the simulation of the tangle : discrete model, 2016. URL : `http://iota.org/simulation_tangle-preview.pdf`.

[Nak08] Satoshi Nakamoto. Bitcoin : A peer-to-peer electronic cash system. 2008.

[Pop16] S Popov. The tangle. white paper, 2016. URL : `https://iota.org/IOTA_Whitepaper.pdf`.