La Stabilité 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. Il existe peu de travaux qui analysent le *Tangle* et aucun ne modélise la construction du DAG au fil d'une arrivée discrète des transactions. Cet article propose un modèle permettant d'analyser le Tangle et le rôle que joue la preuve de travail dans sa sécurité. Notamment, nous prouvons que les acteurs honnêtes doivent produire un nombre de blocs qui correspond à leur puissance de calcul afin d'éviter des attaques similaires à ce qui se fait dans la blockchain, et ce, pour toute méthode de construction du DAG respectant des hypothèses raisonnables.

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 contient beaucoup de résultats ne sont pas formellement prouvés et se concentre sur un modèle continue. Des résultats obtenus par simulation [Kuś16] dans le modèle discret donne des résultats. Une partie du white paper est consacrée à l'étude des attaques possibles et à leur défense. Cependant, aucune analyse formelle n'est réalisée, laissant planer le doute sur la nécessité de certaines hypothèses. Á notre connaissance, aucun autre article de recherche ne s'est penché sur cette question.

Contributions Après avoir modélisé le Tangle, nous analysons le nombre moyen de rondes avant la première confirmation d'une transaction, ainsi que le nombre moyen de transactions non confirmées au fil du temps. Nous utilisons ici un modèle discret. Ensuite, nous montrons que l'hypothèse d'un ensemble de nœuds honnêtes possédant la majorité de la puissance de calcul et soumettant *constamment* des transactions est nécessaire pour prévenir les attaques de double dépense. Cela peut concerner les futures implémentations de la cryptomonnaie IOTA. Il faut noter que IOTA n'est pas concernée à ce jour car elle utilise un nœud coordinateur validant régulièrement et de manière centralisée les transactions, un comportement qui n'est pas défini dans le protocole Tangle.

Un rapport technique [Bra18] contient les résultats complets de notre recherche, ainsi que les preuves omises.

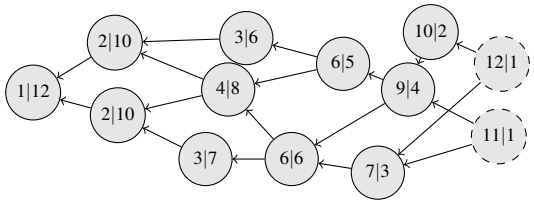

Figure 1: Un exemple de Tangle. Dans chaque site est inscrit son score et son poids cumulé. Les deux extrémités en pointillés ne sont confirmées par aucun site.

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 une ronde. 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 sommet 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. À 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 version 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. Cependant, un parent a ne peut pas confirmer l'autre b, car la présence du parent a prouve que le nœud courant savait que b n'était pas une extrémité. De plus, on pourra supposer que lors de l'ajout de deux sites successifs par un même nœud, des parents différents doivent être choisis (après la sélection des premiers parents, ces-derniers ne sont plus des extrémités), donc de manière globale, le nombre d'enfants par site ne peut pas excéder le nombre de nœuds dans le réseau.

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

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

— 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 : similaire à MCMC mais avec la probabilité de transition suivante :

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

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

3 Performance du Tangle

Dans cette section, nous étudions le nombre moyen de rondes avant la première confirmation d'une extrémité. Nous supposons les mêmes hypothèses que dans les travaux précédents [Pop16] : (i) le TSA aléatoire est utilisé, (ii) on considère que le nombre de sites ajoutés à chaque ronde suit une loi de Poisson de paramètre λ, et (iii) on suppose une latence de $h \geq 1$ dans le réseau i.e., les messages envoyés au temps t sont reçus au temps $t+h$. Nos résultats confirment les résultats obtenus par simulation [Kuś16].

Théorème 1. *Avec $h = 1$, la suite $(N_t)_t$, où N_t est le nombre d'extrémités à la ronde t, est une chaine de Markov qui possède une distribution stationnaire. Ça fonction de transition est :*

$$P_{N \to N'} = \sum_{k=|N'-N|}^{N'} \frac{N! \lambda^k e^{-\lambda}}{N^{2k}(N'-k)!k!} \left\{ \begin{matrix} 2k \\ N - N' + k \end{matrix} \right\}$$

Le théorème permet de calculer la distribution stationnaire du nombre N d'extrémités. Par ailleurs, on remarque qu'à chaque ronde, une proportion λ / N_{avg} des extrémités est confirmée en moyenne, ce qui implique qu'il faut en moyenne N_{avg}/λ rondes avant la première confirmation d'une transaction.

Nous avons calculé par approximation numérique la distribution stationnaire du nombre d'extrémités pour différentes valeur de λ et la Figure 2 en présente les valeurs moyennes N_{avg} divisées par λ, c'est-à-dire, le nombre moyen de rondes avant la première confirmation d'un site.

Lorsque λ tends vers l'infini, des simplifications supplémentaires peuvent être réalisées afin d'obtenir la valeur exacte du nombre moyen d'extrémités $N_{avg} = 2/(W(-2\exp(-2)) + 2)$ lorsque $h = 1$ (où W est la fonction de Lambert) et des approximations numériques lorsque $h > 1$, présentées dans le tableau 1.

h	1	2	10
N_{avg}	1.25500097λ	2.10656565λ	10.01752446λ
$Conf$	1.25500097	2.10656565	10.01752446

Tableau 1: Nombre moyen d'extrémités N_{avg} et du temps moyen de confirmation d'une extrémité, en fonction de la latence h et du paramètre λ, pour des valeurs de λ élevées.

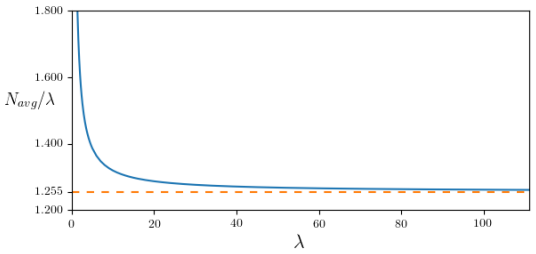

Figure 2: Nombre moyen de rondes avant première confirmation en fonction de λ ($h = 1$)

4 Attaque de la Double Dépense

L'attaque de la double dépense dans les DLT consiste à générer deux transactions utilisant la même source de fond, mais avec deux destinataires distincts. Le but pour l'attaquant est de faire croire a chaque destinataire que la transaction le concernant est valide et que l'autre ne l'est pas.

Pour simplifier, l'un des destinataires peut être un vendeur de voiture et le deuxième est l'attaquant lui même. Dans ce cas, le but pour l'attaquant est d'attendre que le vendeur soit convaincu de la validité de sa transaction et donne les clefs de la voiture. Ensuite, l'attaquant génère plusieurs transactions rendant la première transaction invalide. Lorsque tous les nœuds du réseau considèrent la première transaction invalide,le vendeur aura donné sa voiture sans avoir reçu ces gains.

Dans le cas du Tangle, les travaux précédents n'ont pas étudié formellement la sécurité et laissent entendre que la preuve de travail présent dans les transactions ainsi qu'un TSA adapté suffisent à éviter les attaques de double dépense. Dans cette section nous prouvons que ce n'est pas forcément le cas.

Un attaquant est un nœud qui peut générer des sites ayant des parents arbitraires et il peut décider de ne pas les diffuser immédiatement aux autres nœuds. On définit la notion de *majorité assidue de nœuds honnêtes* comme étant l'hypothèse que les nœuds honnêtes possèdent la majorité de la puissance de calcul *et* qu'ils génèrent de manière continue des transactions. Sans cette hypothèse, nous montrons que toute transaction générée par un attaquant peut être invalidée par la suite. Ce résultat est fort car il est vrai quelque soit l'algorithme de sélection des extrémités choisis, à partir du moment où il existe une manière de générer des sites qui maximise les chances d'être sélectionnés par le TSA. Dans ce cas, on dit que le TSA possède une version maximale.

Définition 1. *La version maximale $\mathcal{T}_{det}$ d'un TSA $\mathcal{T}$ est un TSA qui vérifie : pour tout Tangle G, il existe $N_G \in \mathbb{N}$ tel que pour toute extension $G_{\mathcal{T}}$ de taille n en conflit avec l'extension G_{det} de taille $n+N_G$ on a :*

$$\mathbb{P}[\mathcal{T}(G_{\mathcal{T}} \cup G_{det}) \in G_{det}] \geq \frac{1}{2}$$

Les TSA précédemment définis possèdent tous une version maximale.

Lemme 1. *Les trois TSA existants (aléatoire, MCMC et MCMC logarithmique) possèdent une version maximale.*

Schéma de la preuve. Soit G un Tangle contenant l extrémités et A une extension de G de taille n.

La version maximale du TSA aléatoire choisi simplement comme parents des sites de G (qui ne sont pas forcément des extrémités). L'ensemble A_{det} des sites ainsi générés sont des extrémités. Si $\mathcal{T}$ est exécuté, la probabilité de choisir un site appartenant à A_{det} est donc plus grande que celle de choisir un site de A, lorsque $|A_{det}| > |A|$.

La version maximale de MCMC (et de MCMC logarithmique) est plus difficile à définir. Le point clé est qu'il existe un sous-ensemble du Tangle par lequel la marche aléatoire doit passer (une coupe du graphe). Le but est donc de choisir des parents dans ce sous-ensemble afin d'augmenter le nombre d'enfants directs, et leur poids cumulé, augmentant ainsi la probabilité que la marche aléatoire se dirige vers un des sites de A_{det}. □

Théorème 2. *Sans l'hypothèse d'une majorité assidue et si le TSA possède un version maximale, toute transaction générée par un attaquant peut être invalidée.*

Schéma de la preuve. Lorsqu'un attaquant diffuse dans le réseau une transaction a, il peut générer, sans la diffuser, une transaction conflictuelle $\bar{a}$. Il peut ensuite utiliser toute sa puissance de calcul pour générer des sites qui confirment $\bar{a}$, en utilisant la version maximale du TSA. À tout moment, notamment lorsque le bénéficiaire de a pense avoir effectivement reçu les fonds, l'adversaire peut diffuser l'ensemble des sites dans le réseau. Par définition, les nœuds honnêtes ont plus de chance de choisir la branche du Tangle qui contient $\bar{a}$ et de considérer a comme invalide. □

Conclusion Notre formalisation du Tangle a conduit a deux résultats importants : (i) une analyse théorique des performances comme le nombre moyen de rondes avant la confirmation d'une transaction. et (ii) la preuve que les nœuds honnêtes doivent constamment utiliser leur puissance de calcul afin de sécuriser le protocole. Ce dernier point peut paraître évident, mais il n'existe aucun mécanisme incitant les nœuds à le respecter. Nous apportons ici la preuve de son importance.

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`.