

IOTA-based Directed Acyclic Graphs without Orphans

P. Ferraro, C. King, and R. Shorten

Abstract—Directed Acyclic Graphs (DAGs) are emerging as an attractive alternative to traditional blockchain architectures for distributed ledger technology (DLT). In particular DAG ledgers with stochastic attachment mechanisms potentially offer many advantages over blockchain, including scalability and faster transaction speeds. However, the random nature of the attachment mechanism coupled with the requirement of protection against double-spend transactions leaves open the possibility that not all transactions will be eventually validated. Such transactions are said to be orphaned, and will never be validated. Our principal contribution is to propose a simple modification to the attachment mechanism for the Tangle (the IOTA DAG architecture). This modification ensures that all transactions are validated in finite time, and preserves essential features of the popular Monte-Carlo selection algorithm. In order to demonstrate these results we derive a fluid approximation for the Tangle (in the limit of infinite arrival rate) and prove that this fluid model exhibits the desired behavior. We also present simulations which validate the results for finite arrival rates.

I. INTRODUCTORY REMARKS

Bitcoin, and its underlying technology *Blockchain* [1]–[4], have recently become a source of great debate and controversy in both business and scientific communities. While this debate continues to rage unabated, applications of DLT technologies have begun to emerge across several domains. Applications in logistics [5], banking [6], and in sharing of data [7], are just a few of the many successes of the technology. Roughly speaking, applications of DLT technologies have hitherto focussed on two main areas: (i) as a means of (pseudo) monetary payment; and (ii) for tracking goods and services in a trustworthy manner in complex supply chains. More recently, new application arenas for DLT have been proposed. For example, DLT would appear to offer advantages in applications that require the orchestration of both humans and machines, in applications that require the orchestration of machines and other machines, and more generally in smart city applications, where the issues of social compliance and the enforcement of social contracts are at the forefront (for example discouraging traffic from breaking regulations, parking for a limited amount of time in restricted areas, etc.) [8]. For these types of applications DLT is seen as an interesting enabling mechanism for a number of reasons. First, some DLT technologies (e.g. Legicash,¹ Byteball,²

IOTA [9]) are specifically designed for high frequency micro-trading, and in this arena may be more suitable than systems like PayPal or Visa. For example, sometimes low value transactions will not be processed by vendors (note: however this is also a problem for some DLT architectures such as Bitcoin). Second, systems like PayPal or Visa may require a transaction fee, thus making their use in digital deposit based systems questionable; namely, where the entire value of the token is intended to be returned to a compliant agent. Third, in principle, DLT tokens are more like cash than other digital forms of payment. More specifically, transactions are pseudo-anonymous³ because the encrypted address is less revealing than other forms of digital payments. Card based transactions always leave a trail of what was done and when, and are uniquely associated with an individual, the time and location of the spend, and the transaction item. Thus from a privacy perspective (re. Cambridge Analytica and Facebook⁴), the use of DLT is much more satisfactory than traditional digital transactions.

Our interest in DLT stems from the question of how to enforce social contracts in a smart city context [10]. In our prior work we have used the Tangle [8] for a number of reasons. First, there are no explicit transaction fees associated with IOTA transactions, and second, IOTA is designed to support high-frequency micro transactions. In the IOTA Tangle, newly arrived transactions select two previous transactions to validate via a proof of work mechanism. A key part of the Tangle is the method for selecting these transactions. This is the so-called *tip selection* procedure. To-date, two methods have been widely discussed for tip selection. The first of these is a random selection algorithm, whereby unvalidated transactions are randomly chosen to be validated. The second algorithm is based on a random walk from the interior of the graph (the DAG) to the unvalidated transactions; this is the Monte Carlo Markov Chain (MCMC) selection method. It has been shown in a number of papers that the first of these methods leaves the DAG potentially vulnerable to *double spending attacks* by attackers (whereby the same token can be spent more than once), whereas the second of these algorithms may result in situations whereby some transactions are never validated (orphaned transactions that are simply left behind as the DAG grows [9]) but is less vulnerable to a double spending attack. The intuition behind the MCMC method is that dishonest tips are unlikely to be selected by newly arriving transactions as

Pietro Ferraro and Robert Shorten are with the School of Electrical and Electronic Engineering of the University College Dublin, Belfield, Ireland.

Christopher King is with the Department of Mathematics, Northeastern University, Boston, MA 02115 USA.

¹<https://legi.cash/>

²<https://byteball.org/>

³<https://laurecettennant.com/papers/anonymity-iota.pdf>

⁴<https://www.bbc.com/news/topics/c81zyn0888lt/facebook-cambridge-analytica-data-scandal>

they lie on low probability random paths when the MCMC algorithm is initialised from the interior of the DAG graph. Thus, selecting multiple dishonest tips (necessary for the validation of these transaction) is a very low probability event. Our principal contribution in this note is to propose a new *hybrid tip selection algorithm* that resolves the dichotomy of double spend avoidance and orphans and results in a Tangle where all transactions are validated in finite time. As we shall see, our newly proposed algorithm gives rise to an IOTA Tangle with the property that all transactions are validated in finite time, and at the same time preserves the feature that selection of multiple dishonest transactions is unlikely.

Specific Contribution The work presented in this manuscript builds on a prior work in which mathematical models of the IOTA Tangle with the random tip selection algorithm were developed and validated. The contributions of this paper are as follows.

- We extend the model presented in [8] to include Monte Carlo-inspired tip selection algorithms, giving rise (in the fluid limit) to delayed partial differential equation (PDE) based approximations of the IOTA Tangle.
- Given the PDE model, we propose a new tip selection algorithm, and prove that for this algorithm, and this model, that all transactions are eventually validated.

These results are formalised as a theorem based on a PDE fluid limit. Furthermore, we validate these predictions, and demonstrate the efficacy of the new tip selection procedure using a precise agent based simulation of the Tangle.

Paper structure This paper is organised as follows: Section II provides a qualitative description of the Tangle and of the double spending attack. Simulations show the asymptotic behaviour of the Tangle when the Monte Carlo Markov Chain algorithm is used. Section III discusses a hybrid selection algorithm and the theorem that represents the main contribution of the paper. Section IV describes the mathematical modelling of the fluid limit for the Tangle, based on a set of delayed partial differential equations. Finally, Section V summarizes the results of this work and outlines future lines of research.

II. THE TANGLE

In this paper we are interested in a particular DLT architecture that makes use of DAGs to achieve consensus about the shared ledger. A DAG is a finite connected directed graph with no directed cycles. It consists of a finite number of vertices and directed edges, such that there is no directed path that connects a vertex v with itself. An example of a DAG is depicted in Figure 1.

The Tangle is a particular instance of a DAG based DLT [9]. Its objective, according to the original paper, is to provide a cryptocurrency for the IoT industry which has no fees and has low energy consumption. The Tangle is basically a DAG

where each vertex or *site* represents a transaction (we will use interchangeably the terms site, transaction and vertex), and where the graph represents the ledger. Before being added to the tangle, a new transaction must first approve m (normally two) transactions in the tangle. All yet unapproved sites are called *tips* and the set of all unapproved transactions is called the *tips set*. New transactions will select sites from the tips set for approval (they are not obliged to do so but it is reasonable to expect them to), although since the approval process takes some time (see comments below concerning Proof of Work), these sites may no longer be tips when the transaction is added to the tangle. Each successful approval is represented by an edge of the graph, where a directed edge from site i to site j means that i directly approves j . If there is a directed path (but not a single edge) from i to j we say that j is indirectly approved by i (e.g., see Figure 2). The core metric of the Tangle is the *Cumulative weight* of a transaction: this value represents the total number of vertices that approve, directly or indirectly, a given site. Figure 3 shows an example of how the Cumulative Weight changes in time. The first transaction in the Tangle is called the *genesis* site (namely, the transaction where all the tokens were sent from the original account to all the other accounts) and all transactions indirectly approve it (and therefore its cumulative weight, at every time, is equal to the number of sites in the Tangle). Furthermore, in order to prevent malicious users from spamming the network, the approval step requires a Proof of Work (PoW). This step is less computationally intense than its Blockchain counterpart [12] - [15], and can be easily carried out by common IoT devices (e.g., smartphones, smart appliances, etc.). As mentioned above the Proof of Work introduces a delay for new transactions before they are added to the tangle.

In what follows, we assume that there is a simple way to verify whether the tips selected for approval by a new transaction are consistent with each other and with all the sites directly or indirectly approved by them (this verification occurs during the approval step). If verification fails, the selection process must be re-run until a set of consistent transactions is found. This consistency property is needed in order to prevent malicious users from tampering with the ledger by means of a *double spending* attack (this is described in more detail in the next Section).

As a final note, to illustrate the time evolution of the Tangle, Figure 4 shows an instance of the Tangle with three new incoming sites (upper panel). The green block (the leftmost) is the genesis transaction, blue blocks are transactions that have already been approved, red blocks represent the current tips of the Tangle and grey blocks are new incoming vertices. Immediately after being issued, a new transaction tries to attach itself to m (in this instance two) of the network tips (middle panel). If any of the selected tips was inconsistent with the previous transactions, or with each other, the selection would be rejected and the process would be performed again, until two consistent sites are found. Notice that at this stage, the newly arrived transactions are carrying out the required PoW,

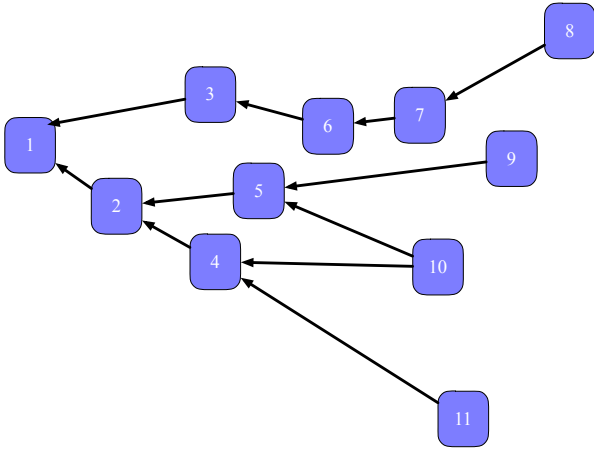


Fig. 1. Example of a DAG with 11 vertices and 10 edges. All the vertices are directed and it is impossible to find a path that connects any vertex with itself.

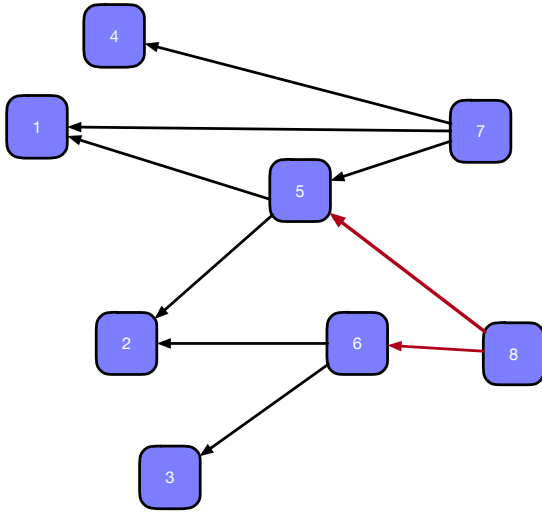


Fig. 2. Transaction 8 directly approves 5 and 6. It indirectly approves 1, 2 and 3. It does not approve 4 and 7.

and that the tips remain unconfirmed (dashed lines) until this process is over. Once the PoW is finished, the selected tips become confirmed sites and the grey blocks are added to the tips set (lower panel).

A. Double Spending Attack and Tips Selection Algorithms

A *double spending* attack is an attempt to exploit the structure of the DLT in order to spend the same digital token multiple times.

To see how a double spending attack might be carried out on the Tangle, let us consider a specific example to explain with more detail the process of approval. Figure 5 shows an

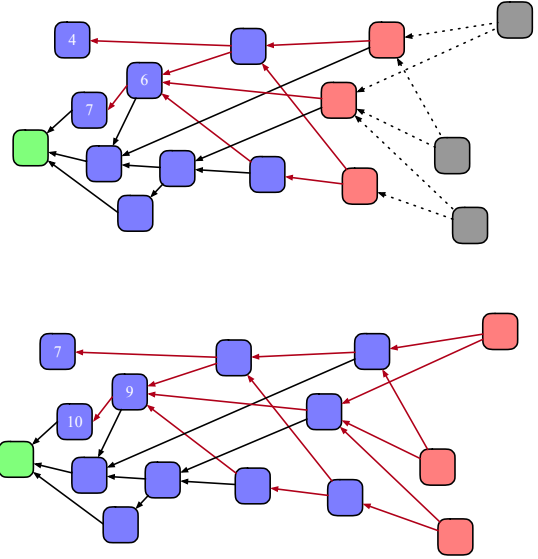


Fig. 3. Representation of the evolution of the Cumulative weight of three sites as three new transactions enter the Tangle

instance of the Tangle. A malicious user sent a certain amount of currency to a merchant. The corresponding transaction is the yellow site. The same user, afterwards, produces other transactions, where she spends again the same tokens that were sent to the merchant. These correspond to the green blocks. It is worth mentioning, at this point, that there is no mechanism to force a user to select certain sites for approval: any transaction can be considered for selection. Nevertheless it is reasonable to assume that the vast majority of users would have little interest in targeting specific sites for approval and would instead follow the tips selection algorithm that the main protocol proposes. In this scenario, all the transactions that approve the original yellow site (the blue blocks) are incompatible with the green ones, therefore any new transactions can either approve the green/black sites or the blue/black ones. The green/blue combination would be considered invalid (as there is an inconsistency in the ledger) and a new selection would be made. The objective of a hypothetical attacker would be then to wait for the merchant to accept their payment, receive their goods, then create one or more double spending transactions that get approved by other honest sites. It is not possible for two conflicting branches of the Tangle to both continue growing indefinitely (for a formal proof on the subject see [8]). Therefore it follows that, if the attacker succeeds in their attempt, the main Tangle will continue to grow from the illegitimate double spending transaction, and the legitimate branch with the original payment to the merchant will be *orphaned*, meaning that its sites will not be approved by new transactions. The success probability of such an attack depends on the particular selection algorithm employed and, as mentioned in the previous Section, two algorithms have

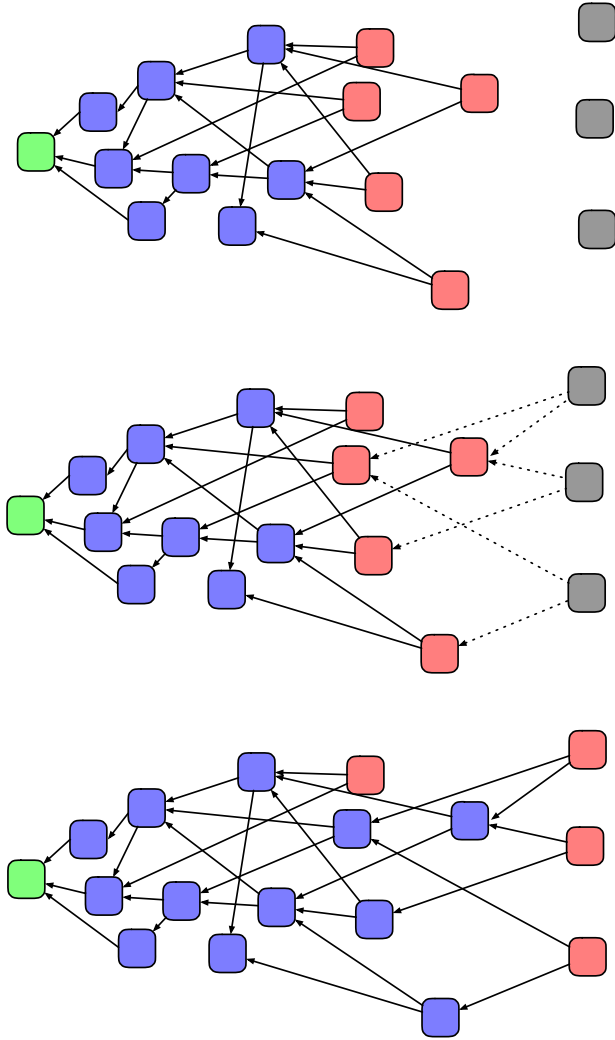


Fig. 4. Sequence to issue a new transaction. The green site represents the genesis block, the blue sites represent the approved transactions and the red ones represent the tips. The black edges represent approvals, whereas the dashed ones represent transactions that are performing the PoW in order to approve two tips.

been proposed so far:

- *Random Selection Algorithm:* The Random Selection (RS) algorithm selects m (generally two) tips randomly from the pool of all possible tips. The upper panel of Figure 6 shows an illustrative example of this procedure. This algorithm, due to its simplicity, makes the Tangle vulnerable to double spending attacks. The interested reader can refer to [9] for a detailed discussion on this topic.
- *Markov Chain Monte Carlo Algorithm:* The Markov Chain Monte Carlo Selection Algorithm (MCMC)

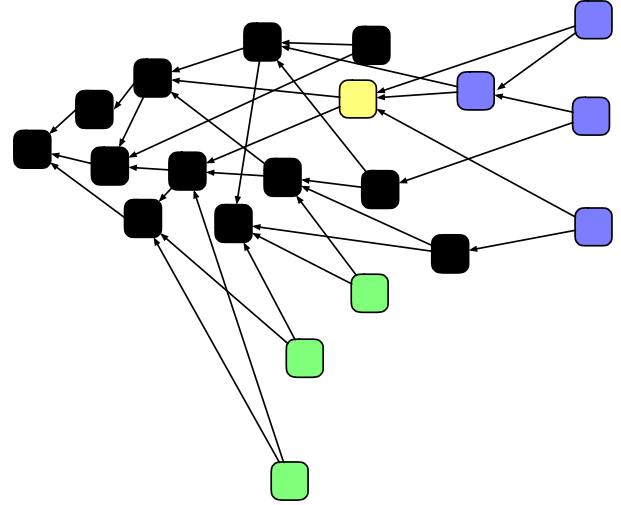


Fig. 5. The blue and the green transactions are incompatible with each other.

works in a slightly more elaborate way than its random counterpart. In the MCMC algorithm m (generally two) independent random walks are created on the tangle; the walks start at the genesis transaction and move along edges of the tangle. The jumping probability from transaction j to transaction k is proportional to $f(-\alpha(\vartheta_j - \vartheta_k))$, where $f(\cdot)$ is a monotonic increasing function (generally an exponential), α is a positive constant and ϑ_i represents the Cumulative Weight of transaction i . The jumping process stops when the particle reaches a tip, which is then selected for approval. The lower panel of Figure 6 shows an example of the paths of two walks in this selection procedure. The main difference with the RS algorithm lies in the use of the graph structure: an attacker would need to create enough transactions, with cumulative weights equal to or larger than the cumulative weight of the main DAG, in order to make the double spending successful. This, in turn, would require the malicious user to possess an amount of computational power comparable to the network of honest users. The interested reader can find more details on this topic in [9] [16].

The goal of the MCMC algorithm is to increase (and ultimately ensure) the security of the Tangle against double spending attacks performed by malicious users. The algorithm and its security from attacks can be tuned using the parameter α : high values of α increase the probability that the particle will jump to the transactions with the largest available cumulative weight (as α approaches ∞ the particles will move in a deterministic way), whereas lower values of α make the algorithm and its output more unpredictable (as the cumulative weights matter less and the jumping probability tends to become uniform as α approaches zero). A good way of picturing the effects of this parameter is by comparing it

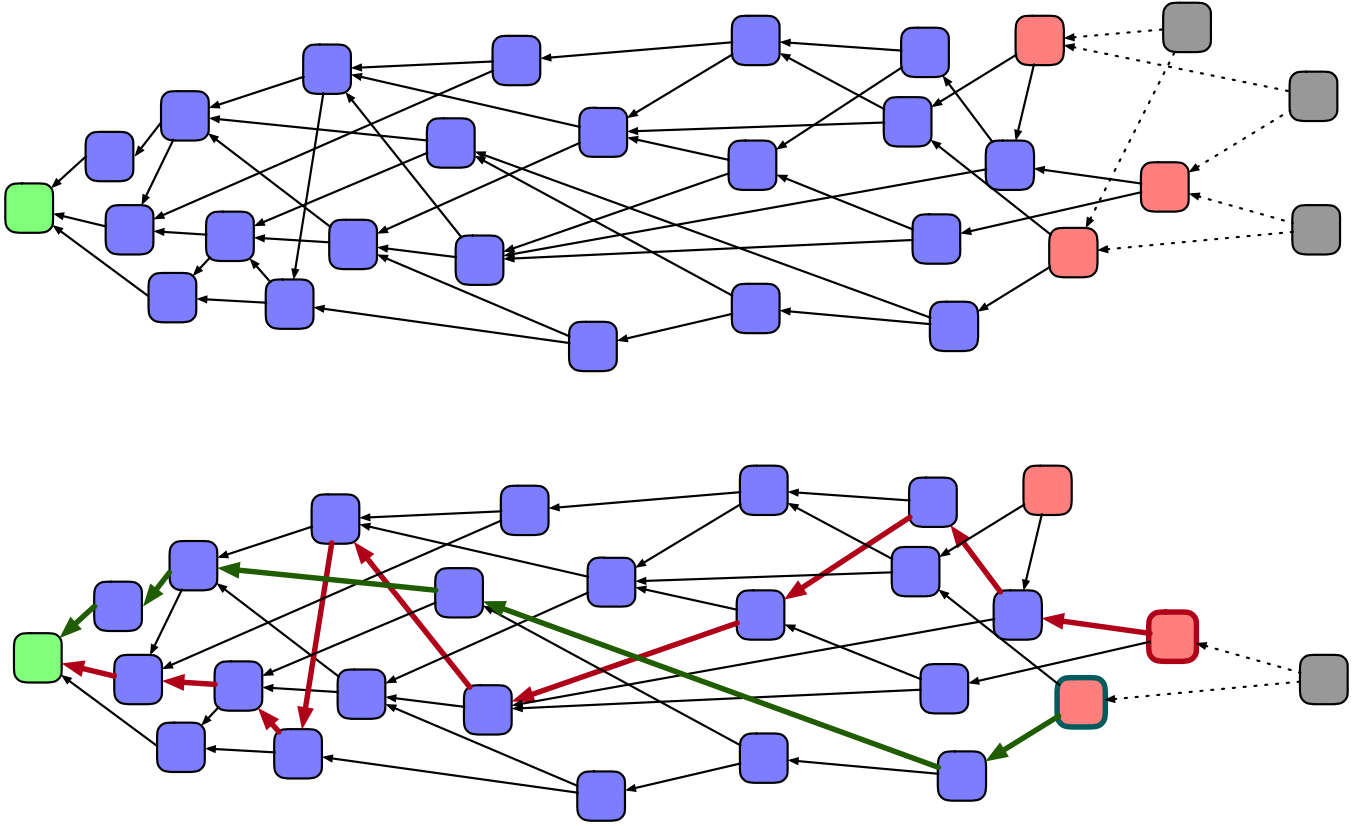


Fig. 6. Representation of the two main tips selection algorithms: the upper panel shows an instance of the Random Selection algorithm whereas the lower one presents a possible example of the Monte Carlo Markov Chain algorithm

to the inverse temperature of a gas: the smaller the value of α the warmer the gas and vice-versa.

It would seem intuitive that in order to ensure the security of the Tangle the best strategy would be to use the MCMC algorithm with a very large value for α , leading to approval for only the most reliable tips (since the ones at the end of the most likely paths would be chosen by the MCMC walk). Unfortunately, this approach would also mean that a very large number of honest tips would never be approved: indeed large α would imply that newly arrived transactions would select with higher probability a small group of tips (the ones which can be reached along paths where the cumulative weight decreases slowly between successive sites on the path). As time passes an unselected tip would be even less likely to be selected since its distance from the genesis would not increase, while the cumulative weight of sites along a path connecting it to the genesis would increase. This outcome is undesirable because it implies that the majority of the transactions would never get approved. However if small values of α are used, so that most honest tips get eventually approved, then the security against

double spending attacks would be compromised. To the best of the authors' knowledge, the only solution proposed so far to this issue is to find a value of α that represents a compromise between the system throughput and its security [16]. To verify that these behaviours do in fact occur we use an agent based simulation of the Tangle: at each time step a random number of transactions arrive, and for each one of these transactions the tip selection algorithm (the MCMC in this scenario) is performed on the current tips set in order to generate graph structures equivalent to the ones presented in detail in Section II. In other words, this agent based model simulates the behaviour of each transaction, therefore providing an accurate replica of the mechanism described in Section II. We suppose that at every time step the number of new transactions is generated according to a Poisson process with parameter λ and that the PoW for the approval procedure of a tip takes h time steps to complete. The variable of interest is the number of tips $L(t)$, which is the number of transactions present in the tips set at the end of each time step. Due to the stochastic nature of the Tangle, 50 Monte Carlo simulations are performed in order to obtain meaningful results.

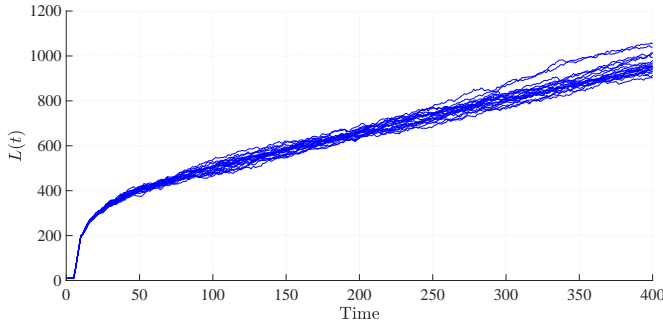


Fig. 7. 50 realizations of the Tangle with the MCMC selection algorithm. The system's parameter are $\lambda = 30$, $h = 5$, $\alpha = 0.1$

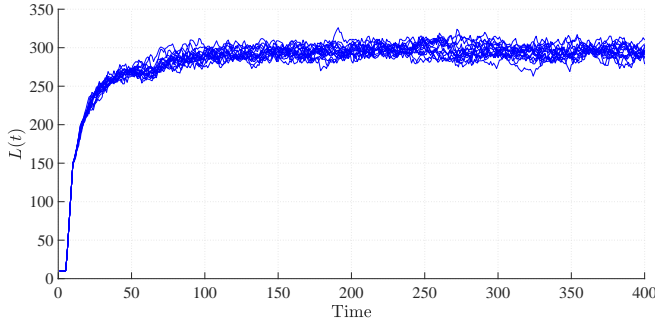


Fig. 8. 50 realizations of the Tangle with the MCMC selection algorithm. The system's parameter are $\lambda = 30$, $h = 5$, $\alpha = 0.001$

Accordingly, Figures 7 and 8 show that for $\alpha = 0.01$ the number of tips diverges with time, whereas for $\alpha = 0.0001$ it fluctuates around a constant value. Clearly, these are only two examples of the possible instances of the Tangle as α varies. From a general point of view, it is very hard to find, in advance, a good trade-off value to make the Tangle simultaneously stable and secure. The Tangle's behaviour depends heavily on parameters such as the rate of arrivals of new transactions and the time needed to complete the PoW.

III. MCMC-LIKE TIP SELECTION WITHOUT ORPHANS

As these simulations of the MCMC algorithm illustrate, there is a tension between the need to protect the Tangle against double spending attacks, and the need to ensure that all transactions eventually get approved. For high values of α the MCMC algorithm favours longer paths from the starting site to the terminal tip, and so the probability of selecting older tips decreases with time, as the paths to these older tips do not grow. Hence the number of unapproved tips continues to grow, and with time the probability of selection for older tips decreases to the point that it becomes impossible for old tips to be selected. By contrast small values of α allow older tips to be selected, however at the same time leaving the Tangle vulnerable to double spending attacks. This motivates the search for new tip selection algorithms that are not easily

exploitable and for which all tips eventually get approved. In this section we propose one such new algorithm, which we call the *hybrid selection algorithm*.

In order to combine the best properties of the two scenarios (large and small α), we propose the following hybrid selection algorithm, which can be divided conceptually into two steps:

- *Security Step*: the first set of selections is made using the MCMC algorithm with a high value of α . This is done to protect security by ensuring that honest tips get selected preferentially;
- *Swipe Step*: the set of second selections is performed using a different algorithm: it can be a RS or a MCMC with a low value of α . This step serves the purpose of taking care of the older transactions that are not likely to be selected by the first step.

Roughly speaking, the Security Step takes care of the security of the system, whereas the purpose of the Swipe Step is to ensure that no tips get left behind by the first, more accurate, selection.

Figure 9 shows the dynamics of the number of tips $L(t)$ under the hybrid selection algorithm. Note that the values for α and λ in this simulation are higher than the unstable simulations performed in Section II. The selection procedure, while being as safe from double spending attacks as any Tangle with high values of α , also ensures that eventually all tips get approved, thus confirming the theoretical result presented in this section.

In the next section we present a mathematical model for the dynamics of the Tangle under various tip selection algorithms, in the regime of high arrival rate (also known as the fluid limit). Within this model we formulate and prove a result that demonstrates stability of the size of the tip set when the hybrid algorithm is used.

IV. MATHEMATICS OF THE DAG: A FLUID LIMIT

We now present, on the basis of some of the results obtained in [8], a mathematical model that approximates the Tangle behaviour with large arrival rate, taking into account the mechanics discussed in the previous Section.

The Tangle can be described as an increasing family of DAG's, $\{G(t), t > 0\}$ where each site of $G(t)$ contains the record of a transaction which arrived at or before time t . We are interested in describing the growth of the Tangle $G(t)$ in a situation where one central server keeps the record of all the transactions. In a real network there would be multiple local copies of the Tangle, and each user would independently update its own copy. However due to synchronization issues, the presence of multiple independent servers complicates the analysis; therefore, we assume the simpler scenario with one main server, and multiple users accessing it whenever they

want to issue a transaction.

We assume that each newly created transaction selects $m \geq 2$ tips for approval, and attempts to validate them (note that $m = 2$ in the standard Tangle). If validation fails the choices are discarded and another m tips are selected for validation. This continues until the process is successful, and we assume that this whole validation effort is essentially instantaneous. However after the validation there is a fixed waiting period h during which the PoW is carried out and the transaction is communicated to the server where the Tangle is stored. During this time the approvals of the selected tips are pending, so the tips may still be available for selection by other new transactions. After the waiting time h , new directed edges are added to the graph, directed from the new site (corresponding to the m tips which were successfully validated). After this point the m parent sites are no longer tips, and so are no longer available for selection by other new transactions. However it may happen that some of these sites had already ceased to be tips at an earlier time, due to their being validated by some other new transaction.

To facilitate the discussion in the sequel, we consider a general algorithm for selecting the tips for validation. Letting $\mathcal{L}(t)$ denote the set of tips at time t , we assume that there are m probability distributions $Q^{(j)}(t) = \{Q_b^{(j)}(t) : b \in \mathcal{L}(t)\}$ ($j = 1, \dots, m$) defined on the set of tips at time t . The newly created transaction selects the first tip for approval using $Q^{(1)}(t)$, selects the second tip using $Q^{(2)}(t)$, and so on to the m^{th} tip. These probability distributions are continually updated as the tangle grows.

The model involves these variables:

- 1) $\mathcal{L}(t)$ is the set of tips at time t
- 2) $\mathcal{W}(t)$ is the set of ‘pending’ tips at time t which are being considered for approval by some new transaction
- 3) $\mathcal{X}(t) = \mathcal{L}(t) \setminus \mathcal{W}(t)$ is the set of ‘free’ tips at time t
- 4) $L(t) = |\mathcal{L}(t)|$, $W(t) = |\mathcal{W}(t)|$, $X(t) = |\mathcal{X}(t)|$
- 5) T_a is the time when transaction a is created, and the PoW starts
- 6) $T_a + h$ is the time when the PoW ends, and the transaction a is added to the tangle
- 7) $N(t)$ is the number of transactions created up to time t
- 8) $|G(t)| = N(t - h)$ is the size of the DAG at time t
- 9) $U(T_a) \in \{0, 1, \dots, m\}$ is the number of free tips selected for approval by transaction a at time T_a

We have the relations

$$N(t) = \sum_{a: T_a \leq t} 1 \quad (1)$$

$$W(t) = \sum_{a: t-h < T_a \leq t} U(T_a) \quad (2)$$

$$X(t) = N(t - h) - \sum_{a: T_a \leq t} U(T_a) \quad (3)$$

$$L(t) = N(t - h) - \sum_{a: T_a + h \leq t} U(T_a) \quad (4)$$

$U(T_a)$ is a random variable whose distribution depends on the sets $\mathcal{X}(T_a)$, $\mathcal{W}(T_a)$ and $\mathcal{L}(T_a)$, as well as the distributions $\{Q^{(j)}(T_a)\}$. (Note: we assume that variables are updated at the times $\{T_a\}$ and $\{T_a + h\}$ so $U(T_a)$ really depends on the values just before time T_a . For convenience we will ignore this distinction in our notation). Let $R_b^{(j)}(T_a)$ be the indicator variable of the event that b is the j^{th} tip selected by the new transaction a . Since tips are selected independently, and the same tip may be selected multiple times, we have

$$U(T_a) = \sum_{b \in \mathcal{X}(t)} \left[1 - \prod_{j=1}^m \left(1 - R_b^{(j)}(T_a) \right) \right] \quad (5)$$

It follows that the expected value is

$$\mathbb{E}[U(T_a)] = \sum_{b \in \mathcal{X}(t)} \left[1 - \prod_{j=1}^m \left(1 - Q_b^{(j)}(T_a) \right) \right] \quad (6)$$

A. Tip selection algorithms

We will consider a class of selection distributions which are modeled on two well-studied examples from the literature. The first example is the random tip selection algorithm, where the distribution $Q^{(\text{ran})}$ is uniform over the tip set $\mathcal{L}(t)$. In this case

$$Q_b^{(\text{ran})}(t) = \frac{1}{L(t)} \quad (7)$$

and it follows that

$$\mathbb{P}(\text{select a free tip at time } t \text{ with } Q^{(\text{ran})}) = \frac{X(t)}{L(t)} \quad (8)$$

The second example is the MCMC attachment algorithm which selects tips by generating a random walk along paths in the Tangle from the genesis to the tips. The algorithm is designed to favor paths where the change in accumulated weight at each step is small. This feature tends to favor paths where each site was approved relatively soon after being added to the Tangle: if a tip had to wait a long time before it was approved, then its parent sites in the Tangle would have been more likely to accumulate a higher weight while the tip was waiting, which would have led to a larger change in accumulated weight at the step from parent to child. Thus the MCMC algorithm will tend to favor tips which are relatively newly added to the Tangle.

Because the MCMC algorithm generates a complicated probability distribution on the tips which depends on the detailed structure of the DAG, we will model it with a simpler class of distribution which we believe captures important features of the MCMC distribution. We define the *age of a tip* to be the time since the tip was added to the Tangle. As argued above, the MCMC algorithm will generate a distribution which is biased toward tips with smaller age, and thus a suitably age-biased distribution should provide a reasonable model for the behavior of the MCMC algorithm. So we assume that there is a positive function $g(\cdot)$ such that the MCMC distribution $Q^{(\text{MC})}(t)$ has the form

$$\begin{aligned} Q_b^{(\text{MC})}(t) &= \mathbb{P}(\text{select tip } b \text{ at time } t \text{ with MCMC}) \\ &= Z(t)^{-1} g(t - T_b - h) \end{aligned} \quad (9)$$

where $T_a + h$ was the time when the tip was first added to the Tangle, so $t - T_a - h$ is the tip's current age, and where $Z(t)$ is the normalization factor

$$Z(t) = \sum_{b \in \mathcal{L}(t)} g(t - T_b - h) \quad (10)$$

We then have

$$\begin{aligned} \mathbb{P}(\text{select a free tip at time } t \text{ with } Q^{(\text{MC})}) &= \sum_{b \in \mathcal{X}(t)} Z(t)^{-1} g(t - T_b - h) \\ &= \frac{\sum_{b \in \mathcal{X}(t)} g(t - T_b - h)}{\sum_{b \in \mathcal{L}(t)} g(t - T_b - h)} \end{aligned} \quad (11)$$

We will consider tip selection algorithms of the form (9) where g is a general positive function. So we consider functions g_1, \dots, g_m , and following (9) define for $j = 1, \dots, m$

$$Q_b^{(j)}(t) = Z_j(t)^{-1} g_j(t - T_b - h), \quad (12)$$

$$Z_j(t) = \sum_{b \in \mathcal{L}(t)} g_j(t - T_b - h) \quad (13)$$

This class encompasses the random tip selection algorithm (in which case g is constant) as well as the proxy model for the MCMC algorithm. In the case of the MCMC algorithm the weighting provided by the age-biased factor $g(\cdot)$ will concentrate the distribution on recent arrivals, and thus we will assume that, for MCMC-like distributions, for at least for some values of the parameter α , the function g is integrable, that is

$$\int_0^\infty g(s) ds < \infty \quad (14)$$

Note that the condition (14) will not hold true for the random tip selection algorithm.

B. The fluid model

We consider the regime of high arrival rate, where the growth of the Tangle is described by a fluid model. We will present some plausible arguments to describe the dynamics of the tip set at high arrival rate, and use these to derive the fluid limit. We assume a Poisson arrival process with rate λ , so that the fluid limit corresponds to $\lambda \rightarrow \infty$. We rescale variables by λ^{-1} and define the rescaled numbers of tips to be

$$l(t) = \lim_{\lambda \rightarrow \infty} \lambda^{-1} L(t), \quad (15)$$

$$x(t) = \lim_{\lambda \rightarrow \infty} \lambda^{-1} X(t), \quad (16)$$

$$w(t) = \lim_{\lambda \rightarrow \infty} \lambda^{-1} W(t) \quad (17)$$

For a fixed age $s > 0$, the number of tips at time t with ages between s and $s + \delta$ is

$$\#\{a \in \mathcal{L}(t) : s \leq t - T_a - h \leq s + \delta\} \quad (18)$$

In the fluid limit we assume that this quantity grows proportionally to $\lambda \delta$, so we define the density $l(t, s)$ as

$$l(t, s) = \lim_{\substack{\lambda \rightarrow \infty \\ \delta \rightarrow 0}} (\lambda \delta)^{-1} \#\{a \in \mathcal{L}(t) : s \leq t - T_a - h \leq s + \delta\} \quad (19)$$

Then the total number of tips at time t (after re-scaling by λ^{-1}) is

$$l(t) = \int_0^{t-h} l(t, s) ds \quad (20)$$

(note that $s \leq t - h$ since we assume that no tips arrived before time h). We make similar assumptions about the numbers of pending tips and free tips, and define age-dependent densities $w(t, s)$ and $x(t, s)$ in a similar way.

Note that each new transaction selects m tips for approval, and some of these may not be free (due to having been selected by earlier transactions). Therefore at time t the number of pending tips $W(t)$ is at most m times the number of new transactions created in the interval $[t - h, t]$. In the fluid limit this implies the bound

$$w(t) \leq m h \quad (21)$$

Over a small time interval $(t, t + \epsilon)$ the number of arrivals will be $\lambda \epsilon$, and these arrivals will select tips for approval according to the distributions $\{Q^{(j)}(t)\}$ defined in (12). Using (6), it follows that the expected number of tips selected by this group of arrivals which lie in the age range s to $s + \epsilon$ is approximately

$$\lambda \epsilon \sum_{\substack{b \in \mathcal{L}(t): \\ s \leq t - T_b - h \leq s + \epsilon}} \left[1 - \prod_{j=1}^m (1 - Q_b^{(j)}(t)) \right] \quad (22)$$

Similarly the expected number of free tips in the age range s to $s + \epsilon$ selected by this group of arrivals is

$$\lambda \epsilon \sum_{\substack{b \in \mathcal{X}(t): \\ s \leq t - T_b - h \leq s + \epsilon}} \left[1 - \prod_{j=1}^m (1 - Q_b^{(j)}(t)) \right] \quad (23)$$

Now we consider the evolution of the density of free tips in the age range $(s, s + \epsilon)$ between times t and $t + \epsilon$. After rescaling by λ^{-1} , the number of these tips at time t is (approximately) $x(t, s)\epsilon$, and we will compare this with the number at time $t + \epsilon$. During this time increment ϵ the subset of tips of age $(s, s + \epsilon)$ inherits the tips from the age range $(s - \epsilon, s)$ while it also loses some tips due to the selection by new arrivals, as detailed above in (23). Thus the evolution is

$$\begin{aligned} x(t + \epsilon, s) \epsilon &= x(t, s - \epsilon) \epsilon \\ &- \sum_{\substack{b \in \mathcal{X}(t): \\ s \leq t - T_b - h \leq s + \epsilon}} \left[1 - \prod_{j=1}^m (1 - Q_b^{(j)}(t)) \right] \end{aligned} \quad (24)$$

Furthermore for the j^{th} tip selection we have

$$\begin{aligned}
Z_j(t) &= \sum_{b \in \mathcal{L}(t)} g_j(t - T_b - h) \\
&= \sum_{k \geq 1} \sum_{\substack{b \in \mathcal{L}(t): \\ (k-1)\epsilon \leq t - T_b - h < k\epsilon}} g_j(t - T_b - h) \\
&\simeq \sum_{k \geq 1} g_j(k\epsilon) \lambda l(t, k\epsilon) \epsilon \\
&\simeq \lambda \int_0^{t-h} g_j(s) l(t, s) ds
\end{aligned} \tag{25}$$

We define

$$\zeta_j(t) = \int_0^{t-h} g_j(s) l(t, s) ds \tag{26}$$

then we also have

$$\begin{aligned}
&\sum_{\substack{b \in \mathcal{X}(t): \\ s \leq t - T_b - h \leq s + \epsilon}} \left[1 - \prod_{j=1}^m \left(1 - Q_b^{(j)}(t) \right) \right] \\
&\simeq \sum_{\substack{b \in \mathcal{X}(t): \\ s \leq t - T_b - h \leq s + \epsilon}} \left[1 - \prod_{j=1}^m \left(1 - \lambda^{-1} \frac{g_j(s)}{\zeta_j(t)} \right) \right] \\
&\simeq \lambda x(t, s) \epsilon \left[1 - \prod_{j=1}^m \left(1 - \lambda^{-1} \frac{g_j(s)}{\zeta_j(t)} \right) \right] \\
&= \epsilon x(t, s) \sum_{j=1}^m \frac{g_j(s)}{\zeta_j(t)} + O(\lambda^{-1})
\end{aligned} \tag{27}$$

Thus in the limit $\lambda \rightarrow \infty$ the evolution equation (24) becomes

$$x(t + \epsilon, s) = x(t, s - \epsilon) - \epsilon x(t, s) \sum_{j=1}^m \frac{g_j(s)}{\zeta_j(t)} \tag{28}$$

and hence in the limit $\epsilon \rightarrow 0$ we get

$$\frac{\partial x}{\partial t}(t, s) + \frac{\partial x}{\partial s}(t, s) = -x(t, s) \sum_{j=1}^m \frac{g_j(s)}{\zeta_j(t)} \tag{29}$$

This delay equation holds for $s > 0$. At $s = 0$ the number of free tips is continually replenished at the rate λ , as transactions complete the approval process. Hence (29) must be supplemented with the boundary condition

$$x(t, 0) = 1. \tag{30}$$

The dynamical equation for the pending tips can be derived in a similar way. The density $w(t, s)$ receives the outflow from $x(t, s)$, so for $0 \leq s \leq h$ we have

$$\frac{\partial w}{\partial t}(t, s) + \frac{\partial w}{\partial s}(t, s) = x(t, s) \sum_{j=1}^m \frac{g_j(s)}{\zeta_j(t)} \tag{31}$$

For $s > h$ there is also an outflow from w due to approval by new transactions which were created at time $t - h$ and which

selected some of these tips for validation, so for $s > h$ we have

$$\begin{aligned}
\frac{\partial w}{\partial t}(t, s) + \frac{\partial w}{\partial s}(t, s) &= x(t, s) \sum_{j=1}^m \frac{g_j(s)}{\zeta_j(t)} \\
&\quad - x(t - h, s - h) \sum_{j=1}^m \frac{g_j(s - h)}{\zeta_j(t - h)}
\end{aligned} \tag{32}$$

Define

$$\theta(u) = \begin{cases} 0 & \text{for } u < 0 \\ 1 & \text{for } u \geq 0 \end{cases} \tag{33}$$

Noting that $l(t, s) = x(t, s) + w(t, s)$ we get

$$\begin{aligned}
\frac{\partial l}{\partial t}(t, s) + \frac{\partial l}{\partial s}(t, s) &= \\
&\quad - x(t - h, s - h) \sum_{j=1}^m \frac{g_j(s - h)}{\zeta_j(t - h)} \theta(s - h)
\end{aligned} \tag{34}$$

and again this is supplemented by the initial condition

$$l(t, 0) = 1. \tag{35}$$

In summary: we have the following system of dynamical equations for the tip densities in the fluid limit:

$$\begin{aligned}
\frac{\partial x}{\partial t}(t, s) + \frac{\partial x}{\partial s}(t, s) &= -x(t, s) \sum_{j=1}^m \frac{g_j(s)}{\zeta_j(t)} \\
\frac{\partial l}{\partial t}(t, s) + \frac{\partial l}{\partial s}(t, s) &= -x(t - h, s - h) \sum_{j=1}^m \frac{g_j(s - h)}{\zeta_j(t - h)} \theta(s - h)
\end{aligned} \tag{36}$$

with boundary conditions

$$x(t, 0) = l(t, 0) = 1 \tag{37}$$

and where

$$\zeta_j(t) = \int_0^{t-h} g_j(s) l(t, s) ds \tag{38}$$

These equations are deterministic and describe the mean-field behavior of the system. Therefore they provide information about the average behavior of the Tangle in the regime of large arrival rate. Note that these are delay equations and so must be supplemented with additional initial conditions, namely in order to determine the solution the functions $\{x(t, s), l(t, s)\}$ must be provided for all $0 \leq s \leq t \leq h$. Also note that, since we are interested in physically meaningful solutions of (36) (e.g., solutions that satisfy (21)), we assume that (36) admits, at least, one solution $\{l(t, s), x(t, s)\}$, such that $l, x \in C^1(\mathbb{R} \times \mathbb{R}_+)$.

C. Integrable weights: persistence of orphans

In this section we consider the case where all the weight functions are integrable, that is

$$\int_0^\infty g_j(s) ds < \infty, \quad j = 1, \dots, m \tag{39}$$

For example this situation is expected to hold when tips are selected using the MCMC algorithm with a sufficiently large value of α , as our simulations in a previous section show. We will show that orphans

must persist in this situation, meaning that the set of unapproved tips must grow without bound.

Theorem 1. Suppose (39) holds, and assume also that the solution $\{l(t, s), x(t, s)\}$ of the system (36) converges as $t \rightarrow \infty$ (so that the Tangle converges to an equilibrium average behavior, such that $\frac{\partial x}{\partial t}(t, s) = \frac{\partial l}{\partial t}(t, s) = 0$). Let $l(s) = \lim_{t \rightarrow \infty} l(t, s)$, $x(s) = \lim_{t \rightarrow \infty} x(t, s)$, then

$$\int_0^\infty l(s) ds = \int_0^\infty x(s) ds = \infty \quad (40)$$

Proof: Define

$$\zeta_j = \int_0^\infty g_j(s) l(s) ds \quad (41)$$

Time independence of $x(s)$ and $l(s)$ implies from (36) that

$$\frac{dx}{ds}(s) = -x(s) \sum_{j=1}^m \frac{g_j(s)}{\zeta_j}$$

$$\frac{dl}{ds}(s) = -x(s-h) \sum_{j=1}^m \frac{g_j(s-h)}{\zeta_j} \theta(s-h)$$

$$x(0) = 1 \quad (42)$$

$$l(0) = 1 \quad (43)$$

It follows immediately that $l(s) = l(0) = 1$ for $0 \leq s \leq h$, and that

$$l(s) = x(s-h) \quad \text{for } s \geq h \quad (44)$$

Furthermore $l(s)$ is decreasing and so

$$\zeta_j \leq \int_0^\infty g_j(s) l(0) ds < \infty \quad j = 1, \dots, m \quad (45)$$

The solution of (42) is

$$x(s) = \exp \left[- \sum_{j=1}^m \zeta_j^{-1} \int_0^s g_j(u) du \right] \quad (46)$$

and thus it follows that

$$\lim_{s \rightarrow \infty} x(s) = \exp \left[- \sum_{j=1}^m \zeta_j^{-1} \int_0^\infty g_j(u) du \right] > 0 \quad (47)$$

Therefore

$$\lim_{s \rightarrow \infty} \int_0^s x(u) du = \infty \quad (48)$$

Since $l(s) \geq x(s)$ this also implies that $l(s)$ is not integrable.

Remark: as noted before, Theorem 1 implies that when all tip selection algorithms are 'short-range', meaning that all selections favor more recently arrived transactions, the number of orphan transactions must grow without bound.

D. Random tip selection: eventual approval of all transactions

We now consider the situation where at least two of the tips are selected randomly, meaning that the corresponding weight functions are constant. Without loss of generality these can be the first two tip selections, so we assume

$$g_1(s) = g_2(s) = 1 \quad (49)$$

We state and prove a Theorem which shows that in this situation the number of unapproved tips remains bounded as the Tangle grows.

This is in contrast to the situation with integrable weight functions, where the pool of unapproved tips continues to grow without bound.

Theorem 2. Let $\{x(t, s), l(t, s)\}$ be a solution of the system (36), and assume that (21) and (49) hold. Then $l(t)$ is uniformly bounded for all $t \geq 0$.

Proof: applying (49) to (36) we get for $t \geq h$

$$\frac{\partial x}{\partial t}(t, s) + \frac{\partial x}{\partial s}(t, s) \leq -\frac{2x(t, s)}{l(t)} \quad (50)$$

Recall that

$$x(t) = \int_0^{t-h} x(t, s) ds \quad (51)$$

and therefore

$$\frac{dx}{dt} = x(t, t-h) + \int_0^{t-h} \frac{\partial x}{\partial t}(t, s) ds \quad (52)$$

Also we have

$$\int_0^{t-h} \frac{\partial x}{\partial s}(t, s) ds = x(t, t-h) - x(t, 0) = x(t, t-h) - 1 \quad (53)$$

Therefore integrating over s in the inequality (50) we get

$$\frac{dx}{dt} \leq 1 - 2 \frac{x(t)}{l(t)} \quad (54)$$

Furthermore we have $l(t) = x(t) + w(t)$ and thus from (21) we derive

$$l(t) \leq x(t) + mh \quad (55)$$

Substituting into (54), it gives

$$\frac{dx}{dt} \leq 1 - 2 \frac{x(t)}{x(t) + mh} \quad (56)$$

It follows from (56) that

$$\frac{dx}{dt}(t) < 0 \quad \text{for } x(t) > mh \quad (57)$$

Recall that the values $\{x(t)\}$ for $0 \leq t \leq h$ are set by the initial conditions. Therefore from (57) we deduce

$$\sup_{t \geq 0} x(t) \leq \max\{mh, \sup_{0 \leq u \leq h} x(u)\} \quad (58)$$

Since $l(t) \leq x(t) + mh$ this proves that $l(t)$ is uniformly bounded.

Remark: assuming that the distribution of tips approaches an equilibrium as $t \rightarrow \infty$, the result that $l(t)$ is uniformly bounded means that the density of tips at age s converges to zero as $s \rightarrow \infty$. This means that, except possibly for a transient set of tips created by the initial conditions, all tips will be eventually approved under the assumptions of Theorem 2. We conjecture that the same result holds under the weaker assumption that at least one of the tips is randomly selected. This conjecture would imply that $l(t)$ is uniformly bounded for our hybrid algorithm where one tip is selected randomly, and the other tip is selected by the MCMC algorithm. Our simulations of the Tangle presented in Section II support this conjecture. Notice that in these simulations the number of tips $L(t)$ diverges or converges depending on the size of the parameter α (in the proposed model changing α is equivalent to changing the function $g(\cdot)$: larger α is closer to MCMC, smaller α is closer to the random tip selection model). This supports our contention that the PDE model is consistent and matches the behavior of the agent based model of the Tangle.

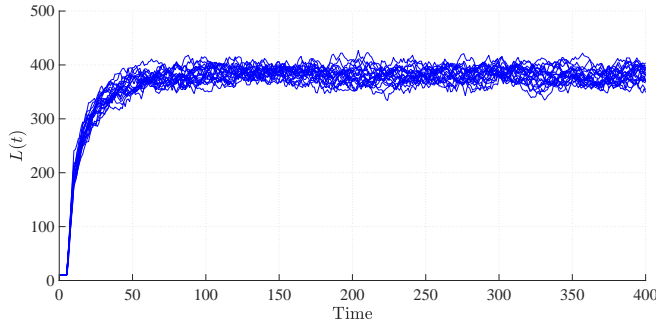


Fig. 9. 50 realizations of the Tangle with the hybrid selection algorithm. The system's parameter are $\lambda = 40$, $h = 5$, $\alpha = 1$

Remark: As a final note, even if the hybrid selection algorithm described so far protects the Tangle from orphaned transactions, it might still leave it vulnerable to a Denial of Service (DoS) attack. As dishonest transactions get issued and not selected, they will accumulate: if their quantity increases to the point that they outnumber honest sites then due to the RS algorithm the selection tip algorithm might need to be repeated multiple times (due to the incompatibility between the swipe selection and the security selection), thus slowing the whole network, or even bringing it to a standstill.

V. CONCLUSIONS

In this paper we develop a new tip selection algorithm that ensure that all honest transactions get eventually approved, while at the same time preserves the basic properties of the MCMC tip-selection algorithm. A fluid approximation of the new IOTA Dag is derived, and the mathematical predictions of this model, under reasonable assumptions, are found to be consistent with the desired properties of the new tip-selection procedure. Future work will be focused on modifications of the hybrid selection algorithm in order to remove the threat denial of service attacks.

VI. ACKNOWLEDGEMENTS

The work is partly supported by the Danish ForskEL programme (now EUDP) through the Energy Collective project (grant no. 2016-1-12530) and by by SFI grant 16/IA/4610.

REFERENCES

- [1] Zheng, Z., Xie, S., Dai, H., Chen, X. and Wang, H., "An overview of Blockchain technology: Architecture, consensus, and future trends" *IEEE International Congress on Big Data*, pp. 557-564, 2017
- [2] Nakamoto, S., "Bitcoin: A peer-to-peer electronic cash system", available at <https://bitcoin.org/bitcoin.pdf>, 2008
- [3] Walport, M. G. C. S. A., "Distributed ledger technology: Beyond blockchain", *UK Government Office for Science, 1 Victoria Street London SW1H 0ET*, No. GS/16/1, 2016
- [4] Puthal, D., Malik, N., Mohanty, S. P., Kougianos, E., and Das, G., "Everything You Wanted to Know About the Blockchain: Its Promise, Components, Processes, and Problems" *IEEE Consumer Electronics Magazine*, Vol. 7, No.4, pp. 6-14, July 2018
- [5] Feng, T., "An agri-food supply chain traceability system for China based on RFID & blockchain technology, *Service Systems and Service Management (ICSSSM)*, 2016 13th International Conference on. *IEEE*, p. 1-6, 2016
- [6] Guo, Y. and Liang, C., "Blockchain application and outlook in the banking industry", *Financial Innovation*, Vol. 2, No. 1, pp. 24, 2016

- [7] Olmes, S., Ubacht, J. and Janssen, M., "Blockchain in government: Benefits and implications of distributed ledger technology for information sharing", *Government Information Quarterly*, Vol. 34, No. 3, pp. 355-364, 2017
- [8] Ferraro, P., King, C. And Shorten, B., "Distributed Ledger Technology for Smart Cities, the Sharing Economy, and Social Compliance, *IEEE Access*, Vol. 6, pp. 62728 - 62746, 2018
- [9] Popov, S., "The Tangle-Version 1.4.2", available at https://iota.org/IOTA_Whitepaper.pdf, 2017
- [10] Fioravanti, A., Marecek, J., Shorten, R. Souza, M., Wirth, F., "On classical control and smart cities", *arXiv preprint arXiv:1703.07308*, 2017.
- [11] Ramakrishna, M. V., Fu, E. and Bahcekapili, E., "Efficient hardware hashing functions for high performance computers" *IEEE Transactions on Computers*, Vol. 46, No. 12, pp. 1378-1381, 1997
- [12] Karame, G. O. and Androulaki, E., "Bitcoin and blockchain security", *Artech House*, 2016
- [13] Karame, G., "On the security and scalability of bitcoin's blockchain" *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pp. 1861-1862, 2016
- [14] Li, X., Jiang, P., Chen, T., Luo, X. and Wen, Q., "A survey on the security of Blockchain systems", *Elsevier, Future Generation Computer Systems*, available at <https://doi.org/10.1016/j.future.2017.08.020>, 2017
- [15] Gervais, A., Karame, G. O., Wst, K., Glykantzis, V., Ritzdorf, H. and Capkun, S., "On the security and performance of proof of work blockchains", *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pp. 3-16, 2016
- [16] Popov, S., Saa, O. and Finardi, P., "Equilibria in the Tangle" *arXiv preprint arXiv:1712.05385*, 2017