Liquidity in Credit Networks: A Little Trust Goes a Long Way

Pranav Dandekar* Ashish Goel† Ramesh Govindan‡ Ian Post§

Abstract

Credit networks represent a way of modeling trust between entities in a network. Nodes in the network print their own currency and trust each other for a certain amount of each other's currency. This allows the network to serve as a decentralized payment infrastructure—arbitrary payments can be routed through the network by passing IOUs between trusting nodes in their respective currencies—and obviates the need for a common currency. Nodes can repeatedly transact with each other and pay for the transaction using trusted currency. A natural question to ask in this setting is: how long can the network sustain liquidity, i.e. how long can the network support the routing of payments before credit dries up? We answer this question in terms of the long term success probability of transactions for various network topologies and credit values.

We show that a number of well-known graph families have the remarkable property that repeated transactions do not result in a loss of liquidity. Further, we show using simulations that the success probability of transactions in Erdös-Rényi and power-law networks depends only on average node degree and credit capacity, not on network size. Finally, we compare liquidity in credit networks to that in a centralized payment infrastructure and show that credit networks are not significantly less liquid; thus we do not lose much liquidity in return for their robustness and decentralized properties.

1 Introduction

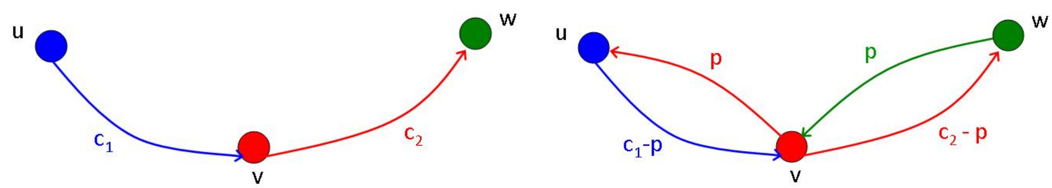
One of the primary functions of money is as a medium of exchange. This function is facilitated by governments and central banks that issue currency and declare it to be legal tender, i.e. the state promises to accept that currency toward settlement of debt. Thus currency represents an obligation issued by the state, and when used to make a payment, results in a transfer of obligation from the payer to the payee. A decision to accept payment in a currency is therefore a decision to trust the issuer of the currency to fulfill its obligations.

Modern banking system is a centralized currency infrastructure. The central bank sits at the root of the tree. It issues currency, government notes, etc. Retail banks are child nodes of the central bank and trust the central bank for an infinite amount of money. Individuals and businesses form the leaves of this tree. Since they cannot print their own currency, they are trusted by banks for a finite amount of money while they, in turn, trust the banks for an infinite amount of money (in theory, at least). We study an alternate model of credit, credit networks, introduced by DeFigueiredo and Barr [DB05] and by Ghosh et al. [GMR+07]. We generalize the model based on the insight that payments can be made in any currency as long as it is trusted by the payee. So nodes in our model act as banks and print their own currency and trust each other for certain amounts of each other's currency. This allows them to transact with each other without the need for a common currency: as long as there is a chain of sufficient credit from the payee to the payer, arbitrary payments can be made by passing obligations between trusting agents in their respective currencies. We study the question of long-term liquidity (i.e. capacity to route payments) in credit networks when nodes repeatedly transact with each other.

Ilustrative Example. Consider a credit network with three nodes u, v and w with edge (u, v) having credit limit c_1 (i.e. u trusts v for up to c_1 units of v's currency) and edge (v, w) having a credit limit c_2. If w wants to purchase a product or service from u worth p units of w's currency, where $p \leq \min\{c_1, c_2\}$, the transaction proceeds by w issuing an IOU to v worth p units of w's currency and v issuing an IOU to u worth p units of v's currency. However, if $p > \min\{c_1, c_2\}$ the transaction fails. As a result of a successful transaction, the edges (u, v) and (v, w) now have capacities $c_1 - p$ and $c_2 - p$ respectively (since v and w have depleted part of their credit line). But this transaction also results in the creation of edges (v, u) and (w, v) each having capacity p, since now v can trust u for up to p units of its own currency, and w can trust v for up to p units of its own currency. Thus routing payments in credit networks is identical to routing residual flows in general flow networks.

Credit networks have a number of interesting and useful properties. First, the loss incurred by nodes in the network is *bounded*. Consider a scenario where a potential attacker injects nodes into this network. The attacker's nodes can extend each

*Department of Management Science & Engineering, Stanford University. Email: ppd@stanford.edu.
†Departments of Management Science & Engineering and (by courtesy) Computer Science, Stanford University. Email: ashishg@stanford.edu.
‡Computer Science Department, University of Southern California. Email: ramesh@usc.edu.
§Computer Science Department, Stanford University. Email: itp@stanford.edu.



(a) Before the transaction

(b) After the transaction

Figure 1: Illustrative Example

other credit lines worth arbitrary amount of money, but in order to transact with one of the honest players the attacker must dupe one of them, say u, into extending him a line of credit. If u extends the attacker node a credit line for c units, it can transact with honest nodes in the network for up to c units by giving u IOUs worth c units in its currency. Once the transaction is complete, the attacker can then disappear leaving u with unredeemable IOUs worth c units, and u incurs a loss equal to c units in the attacker's currency. This demonstrates that the total loss incurred by the nodes in the network is bounded by the total credit extended by honest nodes to the attacker's nodes. In particular, the loss incurred by the nodes in the network does not depend on the number of nodes the attacker injects. Second, since a node in the network only accepts IOUs from other nodes to whom it has extended a line of credit, the loss incurred by honest players in the network is *localized*. In other words, the only nodes that incur a loss are the ones that extended credit to the attacker nodes. In this sense, the system is fair; it protects a node from losses resulting from another node's lack of judgement. Finally, routing payments in credit networks is not significantly less efficient compared to a centralized network since it only requires a max-flow computation [DB05, GMR+07].

These properties make this model useful not only for monetary transactions, but also in any setting where there is a need to model trust between nodes in a network. It is particularly well-suited for transactions in exchange economies such as P2P networks where it can be used to improve inefficiencies resulting from asynchronous demand and bilateral trading [LHL+10]. It has been used as a way of imposing group budget constraint on bidders in a mutli-unit auction [GMR+07]. It can also be used in settings such as packet routing in mobile ad-hoc networks and combating spam in viral marketing over social networks. It has applications in defense where it is important to know who to trust.

However, in order for the model to be of practical use it should be able to support repeated transactions between nodes over a long period of time. This motivates the following question, which we formulate and study in this paper: if the network is sufficiently well-connected and has sufficient credit to begin with, can we sustain transactions in perpetuity without additional injection of credit? How does liquidity depend upon network topology and transaction rates between nodes, and how does it compare with the centralized model described above?

We show positive results for all of the above questions. We show analytically and via simulations that under mild assumptions the success probability of transactions in a number of network topologies such as trees, cycles, complete graphs, Erdös-Rényi graphs and power-law graphs reaches a non-zero steady-state value. Further, for trees, complete graphs and Erdös-Rényi graphs this probability is not significantly lower than that in a centralized payment model with equivalent credit capacity. So in return for all the benefits resulting from the decentralized nature of the system, we do not give up a lot of liquidity compared to a centralized model with a common currency.

We would like to qualify our results by stating that nodes in our model are not endowed with any rationality. We treat credit values on edges and transaction rates between nodes as being exogenously defined. We also ignore currency exchange ratios, i.e. we assume them to be unity for clarity of exposition; this does not qualitatively affect our results. These questions represent promising directions for further study of this model and we discuss them at the end of the paper along with other open problems.

The rest of the paper is organized as follows: Section 2 describes previous work broadly related to trust and reputation in networks, Section 3 defines our model and describes our main results, Section 4 contains steady-state analysis of the transaction success probability in our model and its comparison with a centralized model of credit, Section 5 discusses results of simulating repeated transactions on two random graph families, Erdös-Rényi graphs and power-law graphs constructed using the Barabasi-Albert preferential attachment model [BA99], and finally Section 6 has some concluding remarks along with a discussion of a number of interesting directions for further study of this model.

2 Related Work

The credit network model derives its lineage from two rich bodies of work, the first on studying trust in social networks, and the second on link-based voting/reputation models used most prominently in web search. The study of trust in social networks was first undertaken in the social psychology community in the 1950s (e.g. [CH56]) to understand the implications of Heider's theory of structural balance for networks. More recently, a number of online social services such as Epinions, Essembly, CouchSurfing, etc. have started allowing users to express trust/distrust in other users. This led to the natural question of predicting the polarity between a given pair of users using information about their neighborhood. The question has been studied using a combination of algorithmic techniques and ideas from sociology (e.g. [GKRT04], [LHK10]). The use of link-based voting models was first proposed by Eugene Garfield [Gar55] for measuring the impact of scientific publications based on which other publications cite a given article. More recently, this model has been used extensively in ranking web pages beginning with PageRank [PBMW99] and in subsequent work on similar models, as well as in combating link spam on the web (e.g. [GGMP04]). In all these settings links between nodes are interpreted as a vote of trust and are used to compute the reputation/quality of nodes in the network.

DeFigueiredo and Barr [DB05] and Ghosh et al. [GMR$^+$07] are the first papers that model trust using a credit network. They used the term trust network for the model; we prefer the term "credit network" to distinguish this model from all the previous ones that also model trust in networks. DeFigueiredo and Barr describe strategies to buy and sell credit lines such that losses due to bad transactions are bounded. Ghosh et al. study the problem of multi-unit auction over credit networks. Both these papers assume the existence of a common currency that we do away with. While this prior work has focused on individual transactions, we study the long-term evolution of these networks and their ability to route payments due to repeated transactions.

3 Our Model & Results

A credit network $G = (V, E)$ is a directed graph with n nodes and m edges. Nodes represent entities or agents. Edges represent pairwise credit limits between agents. An edge $(u, v) \in E$ has capacity $c_{uv} > 0$, which implies that u has extended a credit line of c_{uv} units to v in v's currency. If a node s needs to pay p units in her currency to node t (for example, to buy a good that t is selling), the payment can go through if the max credit flow from t to s is at least p units. The payment will get routed through a chain of trusted nodes from s to t. Note that payment flows in the opposite direction of credit, so a payment merely results in a redistribution of credit; the total credit in the network remains unchanged.

We study liquidity under the following simple model of repeated transactions: we assume that each edge in G has integer credit capacity c. The transaction rate between nodes is given by the $n \times n$ matrix $\Lambda = \{\lambda_{uv} : u, v \in V; \lambda_{uu} = 0\}$. At each time step, we pick a pair of nodes (s, t) with probability λ_{st} and try to route a unit payment along the shortest feasible path from s to t. If such a path exists, we route the flow along that path and modify edge capacities along the path as described previously (i.e. for each edge (u, v) on the path, decrement c_{uv} by 1 and increment c_{vu} by 1). The transaction fails if there is no path between s and t. It might seem that the model is too stylized, but in fact routing any integer flow in a network where edges have integer capacities can be performed by successively routing unit flows along paths from the source to destination. So this setting is fairly general and captures a number of intended applications of this model.

We begin by noting that at any time step an edge between nodes u and v can be in one of $(c + 1)$ states, having capacity c from u to v and 0 from v to u, having capacity $c - 1$ from u to v and 1 from v to u and so on down to having 0 capacity from u to v and c from v to u. Each successful transaction results in a change of state for all edges along the shortest path from the source to the destination. Therefore, these repeated transactions between nodes in G induce the following Markov chain $\mathcal{M}(G, \Lambda)$: each state of the Markov chain encapsulates the state of all the edges in the network. Since there are m edges, each of which can be in one of $(c + 1)$ states, the Markov chain has $(c + 1)^m$ states. A successful transaction results in a transition between two states. Let P be the transition matrix. The transition probability, $P(\mathcal{S}_1, \mathcal{S}_2)$ from state $\mathcal{S}_1$ to $\mathcal{S}_2$ is equal to λ_{st} where routing a unit payment from s to t in state $\mathcal{S}_1$ along the shortest feasible path results in the network being in state $\mathcal{S}_2$. The probability $P(\mathcal{S}, \mathcal{S})$ is the probability that the transaction is infeasible when the network is in state $\mathcal{S}$. This is equal to the sum of all probabilities λ_{st} such that the network in state $\mathcal{S}$ has no path from s to t.

A note on terminology: we will sometimes refer to an edge (u, v) as being "bidirectional" if both c_{uv} and c_{vu} are non-zero, and as "unidirectional" if exactly one of c_{uv} and c_{vu} is zero.

We describe our main results below. Our first result obviates the need to route payments along the shortest feasible path.

Theorem. *Let $(s_1, t_1), (s_2, t_2), \ldots, (s_T, t_T)$ be the set of transactions of value $v_1, v_2, \ldots, v_T$ respectively that succeed when the payment from s_i to t_i is routed along a path $\mathcal{P}_i$. Then the same set of transactions succeed when the payment from s_i to t_i is routed along any other feasible path $\mathcal{P}'_i$.*

In order to prove this, we observe that routing flow along a directed cycle changes the state of the network but does not affect the total credit available to any node. This motivates the following definition:

Definition 1. *Given a credit network G, let $\mathcal{S}$ and $\mathcal{S}'$ be two states of G. We say that $\mathcal{S}'$ is* cycle-reachable *from $\mathcal{S}$ if the network can be transformed from state $\mathcal{S}$ to state $\mathcal{S}'$ by routing a sequence of payments along feasible cycles (i.e. from a node to itself along a feasible path).*

Cycle-reachability defines a partition $\mathcal{C}$ over the set of possible states of the network; each equivalence class in $\mathcal{C}$ is the set of states that are cycle-reachable from each other. We show the following result about the steady-state distribution of $\mathcal{M}$ over equivalence classes in $\mathcal{C}$:

Theorem. *Consider a Markov chain $\mathcal{M}_{\mathcal{S}_0}(G, \Lambda)$ starting in state $\mathcal{S}_0$ induced by a symmetric transaction rate matrix Λ over nodes in G. Let $\mathcal{C}_{\mathcal{S}_0} \subseteq \mathcal{C}$ be the set of equivalence classes accessible from $\mathcal{S}_0$ under the regime defined by Λ. Then $\mathcal{M}_{\mathcal{S}_0}$ has a uniform steady-state distribution over $\mathcal{C}_{\mathcal{S}_0}$.*

We use this result to show that if $\mathcal{M}(G, \Lambda)$ is ergodic and Λ is symmetric, then trees, cycles and complete graphs have a non-zero steady-state success probability. Further, we characterize a centralized banking system as a special type of a tree network (one where edges to the root have infinite credit capacity, and transactions occur only between leaf nodes) and compute its steady-state success probability. We show how to construct a centralized system that is equivalent to a given credit network and that, remarkably, the steady-state success probability in trees and complete graphs is not significantly worse than equivalent centralized payment systems.

In addition to these analytical results, we simulated repeated transactions on two random graph families, Erdös-Rényi graphs graphs and power-law graphs generated using the Barabasi-Albert preferential attachment (PA) model [BA99]. The simulations showed that for both topologies the size of the network (i.e. number of nodes) had no effect on the steady-state success probability, only network density (i.e. average node degree) and the credit capacity do. Further, relatively small values of density and credit capacity were sufficient to sustain a high transaction success probability.

4 Analysis

In this section we analyze the steady-state behavior of the Markov chain induced by repeated transactions on the credit network and compare the steady-state success probability in credit networks with various topologies to that in an equivalent centralized payment infrastructure.

4.1 Steady-state Analysis

A state of the network can be associated with a score vector, which characterizes the credit available to each node in that state.

Definition 2 (adapted from [KW81])**.** *Given a directed graph D over n nodes, a vector $V = \langle v_1, \ldots, v_n \rangle$ is the score vector of D if the total capacity of outgoing edges from i in D is v_i for $i = 1, \ldots, n$.*

Note that two cycle-reachable states have the same score vector, and in fact, score vectors characterize classes of cycle-reachable states:

Lemma 1 (from [Gio07])**.** *Given a credit network G, two states $\mathcal{S}$ and $\mathcal{S}'$ of G are cycle-reachable if and only if they have the same score vector.*

Moreover, two cycle-reachable states have exactly the same set of feasible transactions:

Lemma 2. *For any equivalence class $C \in \mathcal{C}$ of a given network, if a transaction (s, t) is feasible in some state $S \in C$, it is feasible in all states $S' \in C$.*

Note that the above observations do not require the edge capacities or payment flows to be integral. However, when restricted to a setting where edge capacities are integers and we route unit flows between nodes, the states of the credit network become equivalent to orientations of a multigraph (since an edge of capacity c can be viewed as c edges of unit capacity each). There is a rich literature (e.g. [BO92, KW81, Gio07] and references therein) on the connections between the number of orientations of a graph, score vectors of a graph, forests of a graph and colorings of a graph. These and other graph invariants can be calculated as an evaluation of its Tutte polynomial. We use some of these results in this paper, particularly the ones on complete graphs.

Using the definition of cycle-reachability and the above observations, we show the following:

Theorem 3. *Let $(s_1, t_1), (s_2, t_2), \ldots, (s_T, t_T)$ be the set of transactions of value $v_1, v_2, \ldots, v_T$ respectively that succeed when the payment from s_i to t_i is routed along a path $\mathcal{P}_i$. Then the same set of transactions succeed when the payment from s_i to t_i is routed along any other feasible path $\mathcal{P}'_i$.*

Proof. Note that a failed transaction does not change the state of the network, so without loss of generality, we assume that the set of successful transactions (s_i, t_i) occurred in successive timesteps $i = 1$ to T. We prove the result by induction on T. The statement clearly holds for $T = 1$.

Assume that the statement holds for $T = k$. Let the initial state of the network be $\mathcal{S}_0$. Let $\mathcal{S}_k$ be the state of the network when, starting from $\mathcal{S}_0$, transactions (s_i, t_i) for $i = 1, \ldots, k$ were routed along path $\mathcal{P}_i$. Let $\mathcal{S}'_k$ be the corresponding state of the network after, again starting from $\mathcal{S}_0$, the same set of transactions were routed along paths $\mathcal{P}'_i, i = 1, \ldots, k$. We will show that $\mathcal{S}_k$ is cycle-reachable from $\mathcal{S}'_k$.

A successful transaction (s, t) only changes the credit extended to s and t; the credit extended to any of the intermediate nodes remains unchanged. Since $\mathcal{S}_k$ and $\mathcal{S}'_k$ are the resulting states of the network after the same sequence of payments were successfully routed starting at the same state, the credit distribution (or equivalently, the score vectors) in $\mathcal{S}_k$ and $\mathcal{S}'_k$ must be identical. Therefore, $\mathcal{S}_k$ and $\mathcal{S}'_k$ are cycle-reachable. Thus, from Lemma 2 if the transaction (s_{k+1}, t_{k+1}) is feasible in $\mathcal{S}_k$ then it is also feasible in $\mathcal{S}'_k$. This proves the result. □

This result shows that routing in credit networks has a *path-independence property*: the choice of paths along which a sequence of payments are routed does not affect their feasibility. Lemma 1 and Lemma 2 immediately implies the following corollary:

Corollary 4. *If a transaction (s, t) in some state in the equivalence class C_i results in a transition to a state in equivalence class C_j, then the reverse transaction (t, s) from any state in C_j will result in a transition to a state in C_i.*

Using these facts we next prove a result about the steady-state distribution over $\mathcal{C}$.

Theorem 5. *Consider a Markov chain $\mathcal{M}_{\mathcal{S}_0}(G, \Lambda)$ starting in state $\mathcal{S}_0$ induced by a symmetric transaction rate matrix Λ over nodes in G. Let $\mathcal{C}_{\mathcal{S}_0} \subseteq \mathcal{C}$ be the set of equivalence classes accessible from $\mathcal{S}_0$ under the regime defined by Λ. Then $\mathcal{M}_{\mathcal{S}_0}$ has a uniform steady-state distribution over $\mathcal{C}_{\mathcal{S}_0}$.*

Proof. As a consequence of the above facts, we can represent transactions as resulting in transitions between equivalence classes in $\mathcal{C}$ instead of transitions between states of $\mathcal{M}(G, \Lambda)$. Let $\mathcal{T}_{ij}$ be the set of transactions (s, t) that result in a transition from some state in C_i to some state in C_j. Then, (overloading the symbol P) we define the transition matrix over equivalence classes in $\mathcal{C}_{\mathcal{S}_0}$ as

$$P(C_i, C_j) = \sum_{(s,t) \in \mathcal{T}_{ij}} \lambda_{st}$$

Note that Corollary 4 implies that $(s, t) \in \mathcal{T}_{ij}$ if and only if $(t, s) \in \mathcal{T}_{ji}$. Further since Λ is symmetric, for any equivalence classes $C_i, C_j \in \mathcal{C}_{\mathcal{S}_0}, P(C_i, C_j) = P(C_j, C_i)$. Since P is a symmetric stochastic matrix, the uniform distribution is stationary with respect to P. □

This immediately gives the following corollary:

Corollary 6. *If $\mathcal{M}(G, \Lambda)$ is an ergodic Markov chain induced by a symmetric transaction rate matrix Λ, it has a uniform steady state distribution over $\mathcal{C}$.*

Next we show a sufficient condition for $\mathcal{M}$ to be ergodic.

Lemma 7. *If for all $u \neq v, \lambda_{uv} > 0$, then $\mathcal{M}$ is ergodic.*

Proof. $\mathcal{M}$ has a finite number of states. Since $\forall u, v, u \neq v, \lambda_{uv} > 0$, for all states $\mathcal{S}$ of $\mathcal{M}$, $P(\mathcal{S}, \mathcal{S}) < 1$. Therefore, $\mathcal{M}$ does not have a sink state. It is irreducible since for any states $\mathcal{S}$ and $\mathcal{S}'$ such that the "edit distance" between $\mathcal{S}$ and $\mathcal{S}'$ is k, the k-step transition probability from $\mathcal{S}$ to $\mathcal{S}'$, $P^k(\mathcal{S}, \mathcal{S}') > 0$. Finally, to see that $\mathcal{M}$ is aperiodic, consider a state $\mathcal{S}$ such that $P(\mathcal{S}, \mathcal{S}) > 0$. For this state, both $P^2(\mathcal{S}, \mathcal{S})$ and $P^3(\mathcal{S}, \mathcal{S})$ are non-zero. Therefore, state i is aperiodic. Since $\mathcal{M}$ is finite-state, irreducible and aperiodic, it is ergodic. □

Theorem 5 and Corollary 6 constitute the central result of this part of the paper. We instantiate this theorem for various network topologies to characterize their steady-state distribution and infer from it the steady-state success probability in those topologies.

4.1.1 Trees

The above results allow us to make the following observations about the steady-state distribution over trees. We denote by $T_{m;c}$ a tree with m edges whose each edge has credit capacity c.

Theorem 8. *If $\mathcal{M}(T_{m;c}, \Lambda)$ is an ergodic Markov chain induced by a symmetric transaction rate matrix Λ on a tree $T_{m;c}$, then $\mathcal{M}(T_{m;c}, \Lambda)$ has a uniform steady-state distribution over its states.*

Proof. The result follows directly from Corollary 6 and the fact that, since trees have no cycles, each equivalence class in $\mathcal{C}$ is a singleton. □

Since $\mathcal{M}(T_{m;c}, \Lambda)$ has a uniform steady-state distribution over its states, the steady-state probability that a transaction over a path of length l will succeed is $[c/(c+1)]^l$; this follows from the fact that each edge along the path should have non-zero capacity in the direction of the transaction. Therefore, the steady-state success probability, $\mathbb{P}_s$, for $T_{m;c}$ is given by

$$\mathbb{P}_s(T_{m;c}) = \mathbb{E}_l \left(\frac{c}{c+1}\right)^l$$

Using this expression for the steady-state success probability, we can show that if Λ is uniform (i.e. $\lambda_{st} = 1/n(n-1)$ for all s, t), the success probability for tree networks is bounded as:

$$\frac{2c}{m(m+1)} \left[1 - \left(\frac{c}{c+1}\right)^m\right] \le \mathbb{P}_s(T_{m;c}) \le \frac{2 + c(m+1)}{m+1} \cdot \frac{c}{(c+1)^2}$$

The lower bound is attained for line networks and the upper bound for star networks (see Appendix).

4.1.2 Cycles

Next we derive the steady-state success probability for cycle graphs. Let $G^\circ_{m;c}$ be a cycle graph of $m > 2$ edges each having a credit capacity of c. Let $\mathcal{M}(G^\circ_{m;c}, \Lambda)$ be an ergodic Markov chain induced by a symmetric transaction matrix Λ. Then $\mathcal{M}(G^\circ_{m;c}, \Lambda)$ will have a uniform steady-state distribution over the cycle-reachable equivalence classes of $G^\circ_{m;c}$. We use this fact to show the following result:

Theorem 9. *If $\mathcal{M}(G^\circ_{m;c}, \Lambda)$ is an ergodic Markov chain induced by a symmetric transaction matrix Λ over nodes in $G^\circ_{m;c}$, then the steady-state transaction success probability, $\mathbb{P}_s(G^\circ_{m;c})$, is given by*

$$\mathbb{P}_s(G^\circ_{m;c}) = \mathbb{E}_l \frac{r^l + r^{m-l} - 2r^m}{1 - r^m}$$

where $r = c/(c+1)$ and l is the distance between a pair of transacting nodes.

Proof. Since $\mathcal{M}$ has a uniform steady-state distribution over the cycle-reachable equivalence classes, in order to compute the steady-state success probability between a pair of nodes separated by l edges in $G^\circ_{m;c}$, we need to compute the total number of equivalence classes and the number of equivalence classes which allow a transaction between that pair of nodes.

Note that each equivalence class can be characterized by a state in which at least one edge in $G^\circ_{c,m}$ has zero capacity in the counterclockwise direction (any state where all edges are bidirectional is cycle-reachable from one where at least one edge is unidirectional). Assume that the nodes are numbered $1, \ldots, m$ and the edge between nodes j and $j+1$ is numbered j. Let the jth edge be the first edge that has zero capacity in the counterclockwise direction. Then the first $j-1$ edges can each be in one of c states, and the remaining $m-j$ edges can each be in one of $c+1$ states. Thus, the total number of states where one or more edges have zero capacity in the counterclockwise direction is given by:

$$\sum_{j=1}^{m} c^{j-1}(c+1)^{m-j} = (c+1)^{m-1} \sum_{j=1}^{m} \left(\frac{c}{c+1}\right)^{j-1} = (c+1)^{m-1} \frac{1-r^m}{1-r} \text{ (where } r = c/(c+1))$$

Now fix nodes s and t that are a distance l apart in the clockwise direction from s to t. Lets call the path of length l between s and t, $\mathcal{P}_1$, and that of length $m-l$, $\mathcal{P}_2$. A successful transaction between s and t along $\mathcal{P}_1$ requires edges in $\mathcal{P}_1$ to have capacity in the counterclockwise direction, whereas that along $\mathcal{P}_2$ requires edges in $\mathcal{P}_2$ to have capacity in the clockwise direction. If the jth edge lies in $\mathcal{P}_1$ then there is no path of length l between s and t. On the other hand, a path of length $m-l$ between s

and t requires each of the $m-l$ edges to be in one of c states. Thus, the number of equivalence classes that only have a path of length $m-l$ between s and t is

$$\sum_{j=1}^{l} c^{j-1}(c+1)^{l-j}c^{m-l} = (c+1)^{l-1}c^{m-l}\sum_{j=1}^{l}\left(\frac{c}{c+1}\right)^{j-1} = c^{m-l}(c+1)^{l-1}\frac{1-r^l}{1-r} \text{ (where } r = c/(c+1)) \quad (1)$$

However, if the jth edge is in $\mathcal{P}_2$, it implies that all edges in $\mathcal{P}_1$ have positive capacity in the counterclockwise direction. Thus, the number of equivalence classes that have a path of length l between s and t is

$$\sum_{j=l+1}^{m} c^{j-1}(c+1)^{m-j} = c^l(c+1)^{m-l-1}\sum_{j=1}^{m-l}\left(\frac{c}{c+1}\right)^{j-1} = c^l(c+1)^{m-l-1}\frac{1-r^{m-l}}{1-r} \text{ (where } r = c/(c+1)) \quad (2)$$

Note that the above equation includes equivalence classes that have capacity along both $\mathcal{P}_1$ and $\mathcal{P}_2$ as well as those that have capacity along only $\mathcal{P}_1$.

Thus, the probability that a transaction between nodes separated by l edges will succeed is given by

$$\frac{c^l(c+1)^{m-l-1}(1-r^{m-l}) + c^{m-l}(c+1)^{l-1}(1-r^l)}{(c+1)^{m-1}(1-r^m)} = \frac{r^l(1-r^{m-l}) + r^{m-l}(1-r^l)}{1-r^m} = \frac{r^l + r^{m-l} - 2r^m}{1-r^m}$$

Taking expectation over the length l of the transaction path gives the result. □

4.1.3 Complete Graphs

Using a similar analysis as we did for trees and cycles, we compute the steady-state probability that a given node in a complete graph will go bankrupt, i.e. will have zero credit available to it. Note that a node being bankrupt is a sufficient but not necessary condition for a transaction with that node as the payer to fail; the necessary and sufficient condition for a transaction to fail is that the network has a directed cut between the payer and the payee. Let $K_{n;c}$ be a complete directed graph over n nodes such that each edge has capacity c (i.e. for all (u,v), $c_{uv} = c_{vu} = c$). Since c is an integer and we route unit flows at each time step, each state of $K_{n;c}$ corresponds to an orientation of a $2c$-*fold* complete graph, where c-fold graph is defined as:

Definition 3 (from [KW81]). *A graph G is c-fold if there are exactly zero or exactly c edges between every pair of nodes.*

Theorem 10. *Let $\mathcal{M}(K_{n;c}, \Lambda)$ be an ergodic Markov chain induced by a symmetric transaction matrix Λ over nodes in $K_{n;c}$. Then the steady-state probability that a given node in $K_{n;c}$ is bankrupt is $\Theta(1/nce)$.*

Proof. Note that $\mathcal{M}(K_{n;c}, \Lambda)$ has a uniform steady-state distribution over the cycle-reachable equivalence classes of $K_{n;c}$. In order to compute the number of equivalence classes in $K_{n;c}$, we first note that the equivalence classes of $K_{n;c}$ are in one-to-one correspondence with score vectors of a $2c$-fold complete graph. Let $V_{n;c}$ be the number of valid score vectors of a c-fold complete graph over n nodes. Kleitman and Winston [KW81] show that

$$\lim_{n\to\infty} V_{n;c} = e^{1/2c}c^{n-1}n^{n-2}$$

The number of equivalence classes in which node u is bankrupt is simply the total number of equivalence classes on the remaining $n-1$ nodes (since the orientation of edges between u and the rest of the nodes is fixed). Therefore, the steady-state probability that a given node in $K_{n;c}$ is bankrupt in the limit as $n \to \infty$ is given by:

$$\lim_{n\to\infty} \frac{V_{n-1;2c}}{V_{n;2c}} = \frac{e^{1/4c}(2c)^{n-2}(n-1)^{n-3}}{e^{1/4c}(2c)^{n-1}n^{n-2}} = \frac{1}{2cn}\left(1-\frac{1}{n}\right)^{n-3} = \Theta\left(\frac{1}{nce}\right)$$

□

We will compare this probability with that in an equivalent centralized model as a way to show that, at least in terms of steady-state bankruptcy probability, credit networks with this topology are comparable to the centralized model. Additionally, we conjecture that in the steady-state transactions primarily fail because of either the payer is bankrupt or the payee does not trust anyone (i.e. the directed-cut that causes the transaction to fail has a single node on one side). More formally,

Conjecture 1. *Let $\mathcal{M}(K_{n;c}, \Lambda)$ be an ergodic Markov chain induced by a symmetric transaction matrix Λ over nodes in $K_{n;c}$. Then the steady-state transaction failure probability in $K_{n;c}$ is $\Theta(1/nce)$.*

4.1.4 Erdös-Rényi Networks

Here we state a conjecture regarding the steady-state failure probability of transactions between nodes in a $G(n,p)$ network under a symmetric transaction regime. The conjecture is based upon a heuristic calculation that the bankruptcy probability of a given node in $G_c(n,p)$ is $\Theta(1/npc)$ (see Appendix) as well as on simulation results described in Section 5. We denote by $G_c(n,p)$ a $G(n,p)$ credit network where each edge has capacity c. The conjecture is as follows:

Conjecture 2. *Let $G_c(n,p)$ be a connected credit network (i.e. $np > \ln n$) and let $\mathcal{M}(G_c(n,p),\Lambda)$ be an ergodic Markov chain induced by a symmetric transaction matrix Λ over nodes in $G_c(n,p)$. Then, the steady-state transaction failure probability in $G_c(n,p)$ is $\Theta(1/npc)$.*

As with complete graphs, we will also compute these probabilities in an equivalent centralized model and argue that this conjecture, if true, would imply that $G(n,p)$ credit networks are not significantly less liquid than a centralized model.

4.2 Centralized Currency Infrastructure

Since we introduced credit networks as an alternative to a centralized currency infrastructure, a natural question that arises is how the steady-state success probability in a credit network compares with that in a centralized infrastructure. Next we address this question.

As described previously, a centralized currency infrastructure can be modeled as a tree with a bank at its root and individuals as leaves. However, this setup is different from a star network in that the root does not participate in any transactions; all transactions occur between leaf nodes, i.e. along paths of length two. Let the centralized infrastructure be represented by a network $\mathcal{G}^*$ with a root node r representing the bank, and n leaf nodes representing individuals. Let the credit capacity extended by the bank to each leaf node be c. Note that since the bank issues a common currency a leaf node trusts the bank for an infinite sum of money. The total credit extended to leaf nodes in the system stays fixed at nc; a successful transaction leads to an increase by one of the payee's credit and a decrease by one of the payer's credit. This induces a Markov chain with $\binom{nc+n-1}{n-1}$ states. Since the centralized system is a tree, from Theorem 8 we know that if this Markov chain is ergodic, it has a uniform steady-state distribution over its states under a symmetric transaction regime. This directly gives the following result:

Theorem 11. *Let $\mathcal{M}(\mathcal{G},\Lambda)$ be an ergodic Markov chain induced by a symmetric transaction matrix Λ over a centralized payment network $\mathcal{G}^*$. Then the steady-state success probability in $\mathcal{M}$ is $c/(c+1)$.*

Proof. Again, the steady-state success probability in $\mathcal{G}^*$ is given by:

$$\mathbb{P}_s(\mathcal{G}^*) = \mathbb{E}_l \left(\frac{c}{c+1}\right)^l$$

where l is the length of the shortest path between the payer and the payee. Since any transaction between leaf nodes in $\mathcal{G}^*$ is constrained only by the credit capacity on the edge from the root to the payer (the edge from the payee to the root has infinite credit capacity), l is always equal to 1. The result follows. $\square$

Thus the liquidity in the banking infrastructure is not significantly better than that in a star-shaped credit network.

4.2.1 Equivalence between Credit Networks and Centralized Currency Infrastructure

In order to be able to compare liquidity in the two models, we need to define the notion of equivalence between the two models. Given a credit network $G = (V,E)$ with n nodes, we construct an equivalent centralized currency infrastructure $\mathcal{G}(G)$ as follows: $\mathcal{G}$ has $\{V\} \cup \{r\}$ nodes (where r is the root node) and a bidirectional edge between each node u and the root r. For each node u in G, the total credit extended to it by the bank in the centralized model, c_{ru}, is given by $c_{ru} = \sum_{v\in V} c_{vu}$ where c_{vu} is the initial credit extended by v to u before the nodes start transacting. And finally, for all nodes u, we set $c_{ur} = \infty$. This ensures that each node has access to the same amount of credit in both models. We then analyze the steady-state success probability of transactions in $\mathcal{G}$ under the same transaction regime.

4.2.2 Cycles

Consider a cycle graph G° with n nodes and edges having capacity c each. The equivalent centralized network $\mathcal{G}(G^\circ)$ will have n leaf nodes, and for every leaf node u, $c_{ru} = c$. Thus, from Theorem 11, the steady-state success probability for $\mathcal{G}(G^\circ)$ under a symmetric transaction regime is $c/(c+1)$. On the other hand, the corresponding probability for the cycle graph is given by

$$\mathbb{P}_s(G^\circ) = \mathbb{E}_l \frac{r^l + r^{n-l} - 2r^n}{1-r^n} \text{ (where } r = c/c+1)$$

which implies that under a uniform transaction regime, the steady-state success probability in cycle graphs as $n \to \infty$ is:

$$\lim_{n\to\infty} \mathbb{P}_s(G^\circ) = \Theta(1/n(1-r))$$

Thus, under a uniform transaction regime cycle graphs have a significantly worse steady-state success probability compared to an equivalent centralized network.

4.2.3 Complete Graphs

Let $K_{n;c}$ be a complete graph on n nodes with edges having capacity c each. We showed that under a symmetric transaction regime, the steady-state probability that a given node will go bankrupt is $\Theta(1/nce)$. The equivalent centralized network $\mathcal{G}(K_{n;c})$ will have n leaf nodes, and for every leaf node $u, c_{ru} = (n-1)c$. From Theorem 11, we know that the steady-state success probability for $\mathcal{G}(K_{n;c})$ under a symmetric transaction regime is $\Theta(nc/(nc+1))$. We prove the following about the steady-state probability that a given node will go bankrupt:

Theorem 12. *Let $\mathcal{M}(\mathcal{G}(K_{n;c}), \Lambda)$ be an ergodic Markov chain induced by a symmetric transaction matrix Λ over nodes in $\mathcal{G}(K_{n;c})$. Then the steady-state probability that a given node in $\mathcal{G}(K_{n;c})$ will go bankrupt is $\Theta(1/nc)$.*

Proof. Note that Markov chain $\mathcal{M}(\mathcal{G}(K_{n;c}), \Lambda)$ has a uniform steady-state distribution over its $\binom{Nc+n-1}{n-1}$ states (where $N = n(n-1)$). Of that, the number of states where a given node u has $c_{ru} = 0$ is given by $\binom{Nc+n-2}{n-2}$. Thus the steady-state probability that a given node is bankrupt is:

$$\frac{\binom{Nc+n-2}{n-2}}{\binom{Nc+n-1}{n-1}} = \frac{n-1}{Nc+n-1} = \Theta\left(\frac{1}{nc}\right)$$

□

Thus, the steady-state bankruptcy probability in $K_{n;c}$ is only a constant factor away from that in an equivalent centralized model, and if our conjecture about the steady-state success probability in $K_{n;c}$ holds, then the success probability too will be only a constant factor worse than that in an equivalent centralized model.

4.2.4 Erdös-Rényi Networks

Let $G_c(n,p)$ be an Erdös-Rényi network with edges having capacity c each. We conjectured that under a symmetric transaction regime, the steady-state probability that a given node in $G_c(n,p)$ with $c = 1$ will go bankrupt is $\Theta(1/np)$. The equivalent centralized network $\mathcal{G}(G_c(n,p))$ will have n leaf nodes, and for every leaf node $u, \mathbb{E}[c_{ru}] = (n-1)pc$. Again, from Theorem 11, we know that the steady-state success probability for $\mathcal{G}(G_c(n,p))$ under a symmetric transaction regime is $\Theta(npc/(npc+1))$. Along the lines of complete graphs, we prove the following about the steady-state probability that a given node in $\mathcal{G}(G_c(n,p))$ will go bankrupt:

Theorem 13. *Let $\mathcal{M}(\mathcal{G}(G_c(n,p), \Lambda)$ be an ergodic Markov chain induced by a symmetric transaction matrix Λ over nodes in $\mathcal{G}(G_c(n,p))$. Then the steady-state probability that a given node in $\mathcal{G}(G_c(n,p))$ will go bankrupt is $\Theta(1/npc)$.*

Proof. The argument is analogous to Theorem 12. The expected number of states in $\mathcal{M}(\mathcal{G}(G_c(n,p), \Lambda)$ is $\binom{Nc+n-1}{n-1}$ states (where $N = n(n-1)p$). Of that, the expected number of states where a given node u has $c_{ru} = 0$ is given by $\binom{Nc+n-2}{n-2}$. Thus the expected steady-state probability that a given node is bankrupt is:

$$\frac{\binom{Nc+n-2}{n-2}}{\binom{Nc+n-1}{n-1}} = \frac{n-1}{Nc+n-1} = \Theta\left(\frac{1}{npc}\right)$$

□

Thus, if our conjecture about the bankruptcy probability and the success probability for $G_c(n,p)$ holds, then the $G_c(n,p)$ credit network will also be at most a constant factor less liquid than an equivalent centralized model.

5 Simulations

Next we present the results of simulating repeated transactions on credit networks from two well-studied families of random graphs: $G_c(n, p)$ (this is a Erdös-Rényi random graph with edge orientation chosen randomly and each edge assigned capacity c) and power-law graphs generated using the Barabasi-Albert preferential attachment (PA) model [BA99]. The Barabasi-Albert model is a evolving random graph model where each new node creates d edges to existing nodes with probability proportional to their degrees. After constructing the graph, we assigned edge orientations randomly and set the capacity of each edge to c. The goal of the simulations, as of the analysis, was to understand how the network evolves as a result of repeated transactions, to determine the steady-state success probability in these networks, and to understand how the network evolution and success probability depend on key network parameters. For $G_c(n, p)$ the parameters of interest are n, p and c. However, since varying either n or p changes the density of the graph (i.e. average node degree), we also ran simulations where we changed both n and p such that np was held constant. For PA graphs, the parameters of interest are the number of nodes (n), the number of edges each node creates (d), and credit capacity (c).

A simulation run consisted of constructing a network with a given set of parameters and doing repeated transactions on it. At each time step, we chose a node pair (s, t) with uniform probability and tried to route a unit payment from s to t via the shortest feasible path. If there was a path from s to t, we routed the payment and modifed edge capacities along the path. Otherwise, we counted the transaction as a failure. We repeated this until the success-rate (i.e fraction of transactions that succeed) in two consecutive time windows was within ε of each other. We used a window size of 1000 time steps and $\varepsilon = 0.002$. When the process converged, we measured the success rate at convergence, i.e. the total number of successful transactions divided by the total number of time steps. We called this the steady-state success probability for that set of network parameters. For each combination of network parameter values, we did a simulation run on 100 graphs constructed from the same distribution and computed the average and the standard deviation in success-rate over 100 runs. In addition to the success-rate, we also record the following metrics for each run:

1. Number of (weakly) connected components: For some values of network parameters, the constructed network had more than one connected component. While that number does not change during the course of a simulation run, it does affect the overall success-rate for the run since transactions between nodes that do not belong in the same connected component fail. So we recorded the number of connected components in the constructed network for each run.

2. Average path-length of successful transactions: For each successful transaction, we recorded the length of the path along which the payment was routed. We averaged it over all the successful transactions in a run. This metric is useful in evaluating the choice of network parameters for problems where transactions over short paths might be preferable either for latency or for reliability reasons.

3. Number of "sinks"/"sources": A sink node is a node that has only incoming edges, no outgoing edges. Analogously, a source node is one that has only outgoing edges, no incoming ones. A payment that has a sink node as its destination or a source node as its source will always fail. We kept a count of the number of source and sink nodes in the network at each time step and averaged it over the duration of the run. Due to the symmetry of transactions, these numbers were approximately equal and exhibit similar variation due to changes in network parameters.

In order to understand how network evolution and success probability depend on network parameters, we repeated the set of 100 runs for different values of one of the parameters, keeping the rest constant.

5.1 Effect of Variation in Network Density

In order to study the effect of density, we fixed $n = 200$ and $c = 1$ for both graph families. For $G_c(n, p)$ we varied p from 0.02 to 0.45 and for PA graphs we varied d from 2 to 45. The average node degree of PA graphs is equal to $2d$ whereas for $G_c(n, p)$ the expected node degree is given by $(n - 1)p$. We picked d corresponding to each p such that $2d \approx (n - 1)p$ so that we can compare the plots for the two graph families. For $p = 0.02$, the constructed $G_c(n, p)$ graphs had 4.52 connected components on average. That number went down to 1 at $p = 0.07$ and stayed there for higher values of p. Figure 2 shows the mean and standard deviation in the steady-state success-rate, the pathlength of successful transactions and the number of sink nodes for both graph families as we vary network density.

Figure 2(a) shows the success probability for $G_c(n, p)$ and PA graphs when np is sufficiently larger than $\ln n$ so that the $G_c(n, p)$ graph is connected. The success rate for both graph families is concave, non-decreasing in p. Further, the actual values of the success probability are also similar for both graph families. Also, the standard deviation in success-rate for both graphs went down with increase in density.

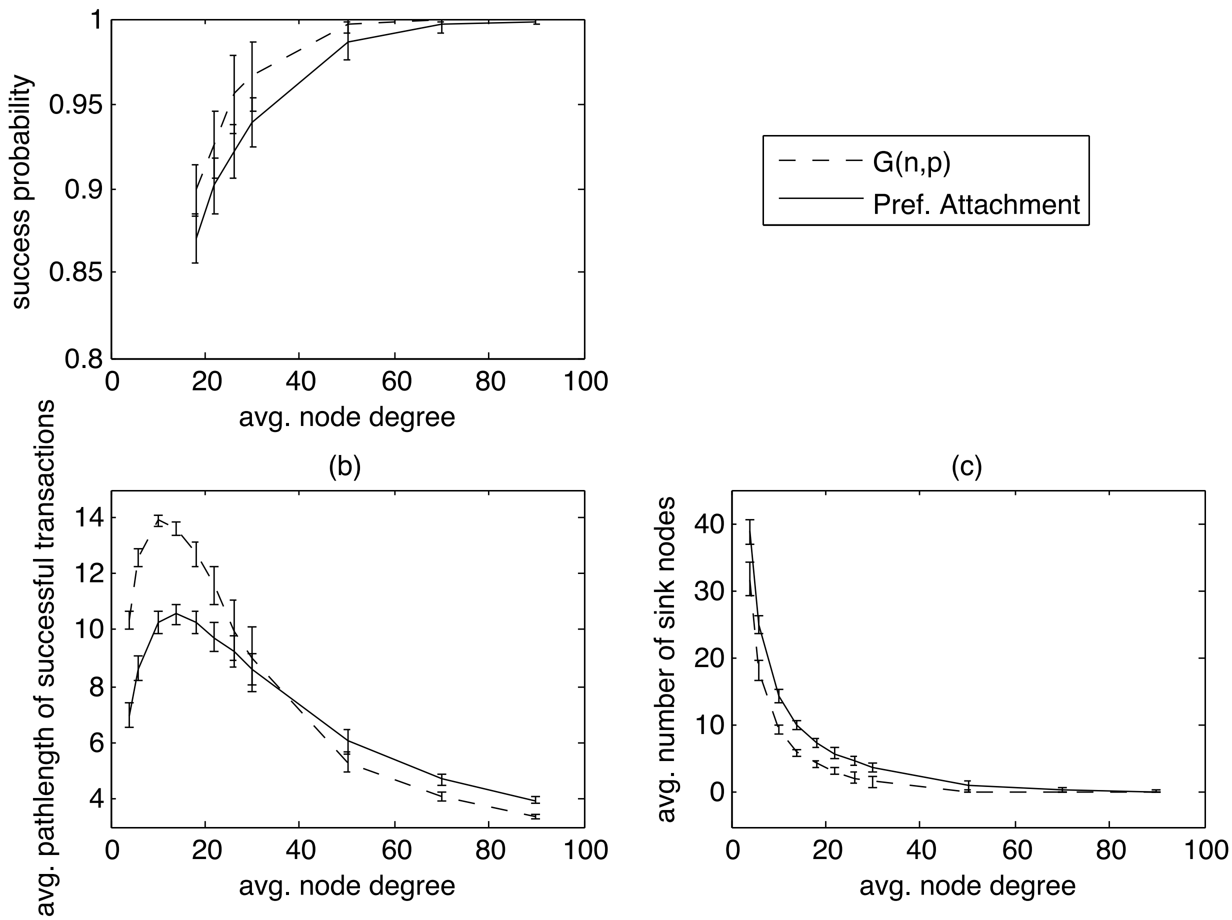


Figure 2: Effect of varying graph density (average node degree) in $G_c(n, p)$ and PA graphs. (a) Variation in average steady-state success probability. (b) Variation in the average pathlength of successful transactions. (c) Variation in the average number of sink nodes.

Figure 2(b) shows the average pathlength of successful transactions for both graph families. The average pathlength for both $G_c(n, p)$ and PA graphs peaks for $p = 0.07$ and then goes down. The increase in the average pathlength for $G_c(n, p)$ graphs for low values of p can be attributed to the fact that the graph is not fully connected; so increasing p enables transactions between distant nodes that were heretofore disconnected. As the graphs become denser, the values for $G_c(n, p)$ come down rapidly and go slightly below those for PA graphs. Thus, if the graph is dense enough, the topology does not significantly affect the average pathlength.

Figure 2(c) shows the average number of sink nodes for both graph families. Because of a heavy-tailed degree distribution, a large fraction of nodes in PA graphs have a degree exactly d. Therefore the number of sink nodes in PA graphs is consistently higher than corresponding $G_c(n, p)$ graphs. However, as the graphs become denser, the difference between the two values becomes insignificant.

Thus, increasing network density increases the steady state success probability and lowers both the average pathlength of transactions as well as the number of sink/source nodes. Moreover, all the metrics quickly reach a point of diminishing returns where making the network denser does not help significantly. This is useful since it means that, in practice, the network does not need to be very dense in order to route payments with a sufficiently high probability.

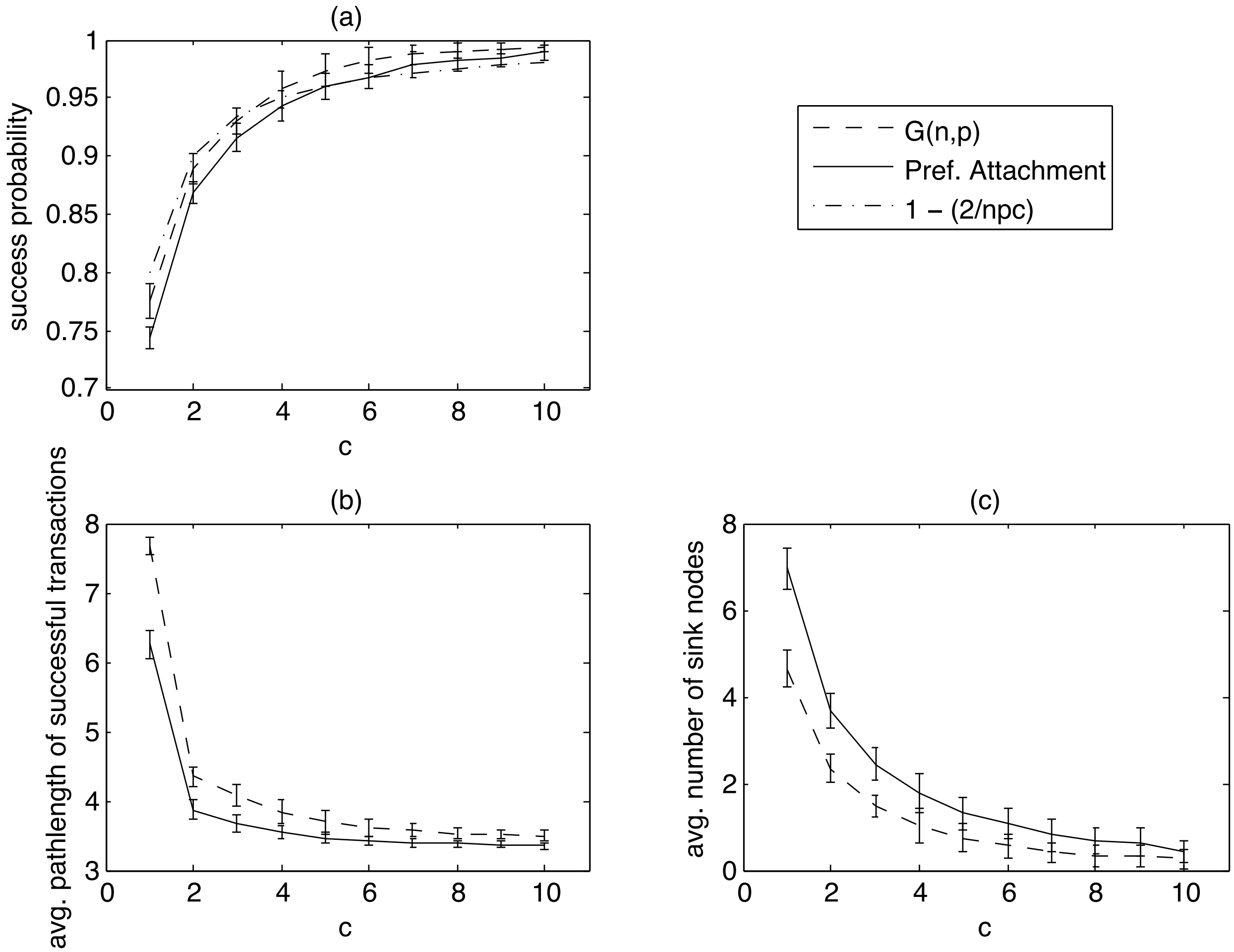


Figure 3: Effect of varying credit capacity in $G_c(n,p)$ and PA graphs. (a) Variation in average steady-state success probability. (b) Variation in the average pathlength of successful transactions. (c) Variation in the average number of sink nodes.

5.2 Effect of Variation in Credit Capacity

In order to study the effect of varying credit capacity in the network, we fixed $n = 100$ for both graphs. We set $p = 0.10$ for $G_c(n,p)$ and $d = 5$ for PA graphs, thereby ensuring that their densities were equal. This choice of p and d also ensured that all the graphs we constructed were connected. We varied c from 1 to 10. Figure 3 shows the mean and standard deviation in the steady-state success-rate, the pathlength of successful transactions and the number of sink nodes for both graph families as we vary c.

Figure 3(a) shows that the success probability in both graph families is similar and is concave non-decreasing in c. In addition to the two plots, we also plot $1 - (2/npc)$ as a function of c. That plot tracks the $G_c(n,p)$ plot fairly closely and is further evidence in support of our conjecture that the steady-state failure probability in $G_c(n,p)$ graphs is $\Theta(1/npc)$.

Figure 3(b) shows the average pathlengths of successful transactions for both graph families. The average pathlength for both $G_c(n,p)$ and PA graphs drops dramatically as c goes from 1 to 2, and then continues decreasing at a much slower rate. The average pathlength of successful transaction in $G_c(n,p)$ graphs remains higher than that for PA graphs throughout, which is consistent with the fact that on average the paths between nodes in $G(n,p)$ are longer than those in PA graphs.

Figure 3(c) shows that the average number of sink nodes in both graph families goes down from greater than 4 at $c = 1$ to less than 1 at $c = 7$. As expected, the number is higher for PA graphs than for corresponding $G_c(n,p)$ graphs due to higher variation in node degrees. Also, the number does not go down as quickly as it does on increasing network density. The variation in the number of source nodes with c looks very similar to these plots.

Thus, increasing credit capacity in the network improves all the metrics we record: success-rate, average pathlengths and number of source/sink nodes. As with density, the metrics quickly reach a point of diminishing return where added capacity does not significantly improve any of the metrics. This is also good news: a credit network does not need a lot of initial credit to bootstrap itself.

5.3 Effect of Variation in Network Size

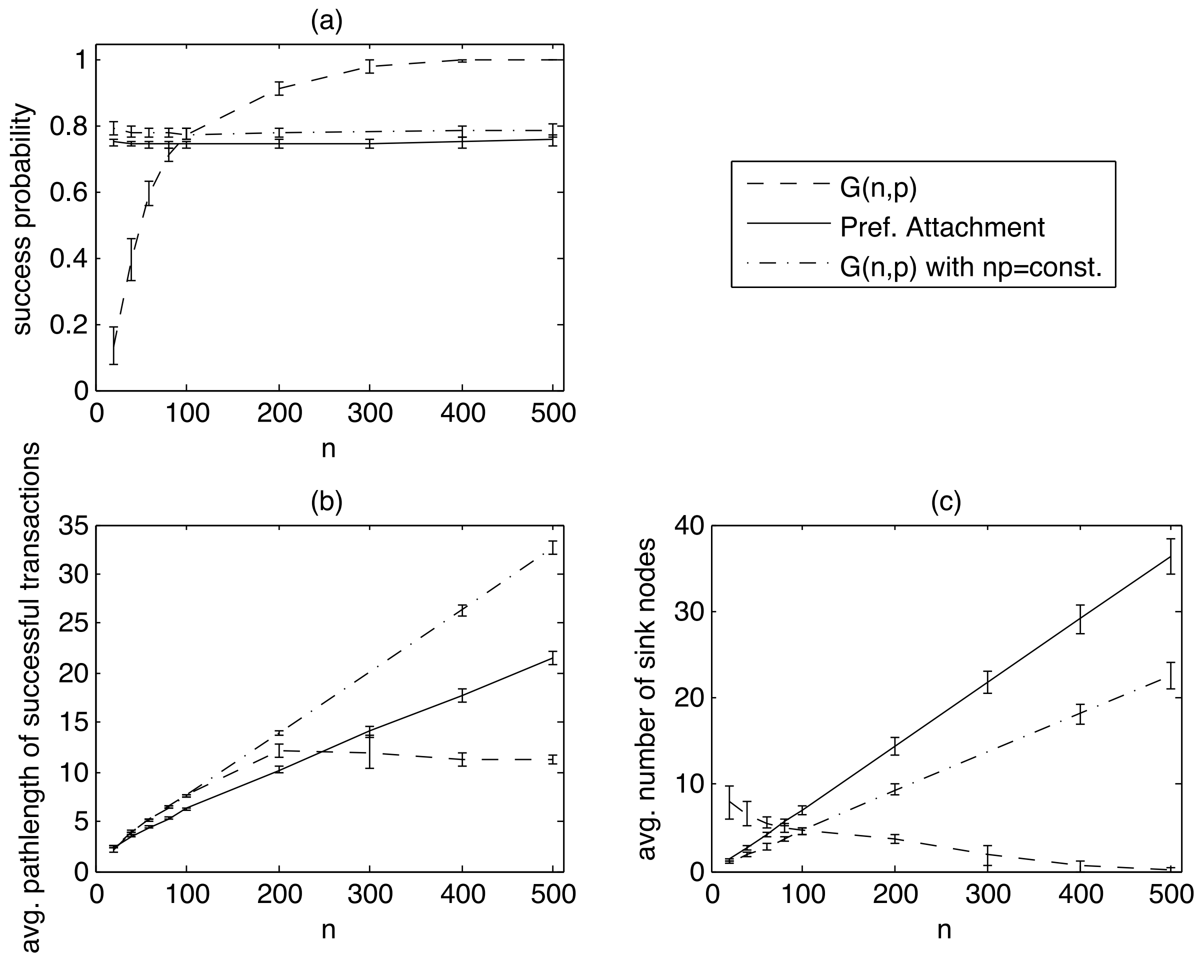


Figure 4: Effect of varying network size (number of nodes) in $G_c(n,p)$ and PA graphs. (a) Variation in average steady-state success probability. (b) Variation in the average pathlength of successful transactions. (c) Variation in the average number of sink nodes.

In order to study the effect of varying network size for PA graphs, we fixed $d = 5$ and $c = 1$ and varied n from 20 to 500. However, the same approach does not work for $G_c(n,p)$ graphs since if we fix p and c and varied n, it would also vary the average node degree. Therefore, we ran two sets of simulations for $G_c(n,p)$ graphs: one where we varied n keeping $p = 0.10$ and $c = 1$ and the other where we fixed $c = 1$ and varied both n and p such that np was kept constant at 10.

Figure 4(a) shows that effect of varying network size on the success probability of both graph families. The remarkable observation is that if the density of the network is kept constant, then network size has no effect on the success rate for either graph family; the success rate remains nearly constant at about 0.75 for PA graphs and about 0.78 for $G_c(n,p)$ graphs. The same pattern is also observed if np is held constant at 20 instead of 10 (so it is not an artifact of that specific value of np). This is another observation in support of our conjecture about the steady-state failure probability in $G_c(n,p)$ graphs. If, however, np is not held constant, then success probability is again a concave non-decreasing function of n (much the same way as it is of p).

Figure 4(b) shows the average pathlength of successful transactions for both graph families as n goes from 20 to 500. If the density of the graph is kept constant, the average pathlength increases almost linearly for both $G_c(n,p)$ graphs as well as PA graphs, although the rate of growth for the latter is slower than the former. This is again consistent if the fact that on average the paths in PA graphs are shorter than those in $G(n, p)$ graphs. If, however, we increase n without keeping np constant, the average pathlength increases from 2.15 to 12.1 as n goes from 20 to 200 and then gradually decreases.

Figure 4(c) shows that the average number of sink nodes in both graph families grows almost linearly with n when we keep the average node degree constant. The number of sink nodes grows faster in PA graphs since they have a larger number of nodes with degree $d = 5$. If we do not keep the density constant, then the number of sink nodes for $G_c(n, p)$ monotonically decreases with n.

Thus, we showed that both $G(n, p)$ and PA graphs have the remarkable property that steady state success probability is independent of network size if the density and edge capacity are kept constant. This is also useful from a practical standpoint, as it means that even small networks can be equally effective at routing payments as long as they are sufficiently well-connected. However, applications that prefer shorter paths will need to increase network density as they grow in order to counteract the adverse effect of network size on pathlengths of transactions.

6 Conclusion & Future Work

In this paper we generalized the credit network model by allowing every node to print its own currency, and formulated and studied the question of long-term liquidity in this network under a simple model of repeated transactions. Using the notion of cycle-reachability, we showed that routing payments in credit networks has a *path-independence* property. We also showed that the Markov chain induced by a symmetric transaction regime has a uniform steady-state distribution over the equivalence classes defined by the cycle-reachability relation. Using this fact, we derived the steady-state success probability for trees and cycles and bankruptcy probability in complete graphs and showed that, except in cycles, these probabilities are not significantly worse than the corresponding values for an equivalent centralized payment infrastructure. Our results imply that for a number of well-known graph families, the steady-state success probability under reasonable transaction regimes is bounded away from zero regardless of the size and credit capacity of the network, and in return for the robustness and decentralized properties this model does not lose much liquidity compared to a centralized model. We also showed through simulations that the steady-state success probability in $G(n, p)$ and power-law graphs under a uniform transaction regime increases with network density and credit capacity, but remarkably, is independent of network size.

6.1 Open Problems related to Liquidity

Our results on liquidity can be extended in a number of ways. One way is to express the steady state success probability and rate of convergence in general networks in terms of some key network properties such as density, size of the min-cut, etc.—in particular to analytically show that success probability in $G(n, p)$ and power-law graphs is independent of network size. Another direction is to understand how success-rate and rate of convergence vary under various transaction regimes. A natural regime to study would be where the probability that a node is a payer (payee) is proportional to the node's in-degree (out-degree). We also leave unanswered the question of mixing times of the Markov chain: our simulations on $G_c(n, p)$ and PA graphs converged relatively quickly but we do not have analytical results on the mixing time of the Markov chain and how it depends on credit capacity and network topology. One can also ask how fault-tolerant credit networks are under various models of node failures. For example, if we remove k nodes (and all their edges) from the network, how would liquidity change in the worst-case, and what network topologies are most robust against such failures? If node failures cascade through the network, how quickly would the network lose liquidity?

6.2 Further Understanding the Model

This model presents a number of interesting directions for further inquiry. The first is the question of pricing of currencies. Let r_{uv} be the exchange ratio for converting u's currency into v's (i.e. 1 unit of u's currency is r_{uv} units of v's currency). The following two properties of these conversion ratios are necessary for the generalized credit network model to be well-defined:

- *Conservation:* For any $S \subseteq V$, where $S = \{u_1, u_2, \ldots, u_k\}, \prod r_{u_1u_2}r_{u_2u_3}\ldots, r_{u_{k-1}u_k}r_{u_ku_1} = 1$. In words, this property means that currency is conserved along any cycle of payments in the network.

- *Common Knowledge:* The conversion ratios are common knowledge for all nodes in V. This means that nodes along a path through which payment is being routed can compute the outflow given the inflow.

However, these properties do not guarantee that it is incentive-compatible for nodes to route payments. For example, it is natural to believe that the currency of a node that lies on the path between a number of (source, destination) pairs should be more expensive compared to one that has, say, only one neighbor. More generally, can we endow the nodes in the network with a model of rationality that will govern how much credit nodes extend each other, how that changes over time, and how nodes negotiate currency exchange ratios as a function of transaction rates, success probabilities, network topology, etc.?

The second broad question is how to apply this model to various practical applications, some of which we mention in the introduction. Of particular interest are analytical results as well as empirical or simulation-based studies that evaluate the effectiveness of this model in settings such as IP routing, peer-to-peer systems, viral marketing, etc. compared to existing models.

Another potential direction is to consider credit networks as enabling a trust-based market where nodes buy and sell goods using trusted currency instead of a common currency. Then we can ask the same questions that have been asked of various traditional market models: Does the market clear? At what prices? Can we efficiently find them? How do equilibrium prices in this model compare with those in models that assume a common currency? Answers to these questions will provide insights into whether and when can we trade off the robustness properties and the decentralized nature of this model with the well-understood characteristics of traditional markets.

7 Acknowledgements

This research was supported in part by NSF award IIS-0904325. Part of the research was also sponsored by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-09-2-0053. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation here on.

References

[BA99] A. L. Barabasi and R. Albert. Emergence of scaling in random networks. *Science*, 286(5439):509–512, 1999. doi:10.1126/science.286.5439.509.

[BO92] T. Brylawski and J. Oxley. The Tutte polynomial and its applications. In N. White, editor, *Matroid Applications*, number 40 in Encyclopedia of Mathematics and its Applications, chapter 6, pages 123–225. Cambridge University Press, 1992.

[CH56] D. Cartwright and F. Harary. Structural balance: a generalization of Heider's theory. *Psychological Review*, 63:277–293, 1956. doi:10.1037/h0046049.

[DB05] Dimitri B. DeFigueiredo and Earl T. Barr. Trustdavis: A non-exploitable online reputation system. In *CEC '05: Proceedings of the Seventh IEEE International Conference on E-Commerce Technology*, pages 274–283, Washington, DC, USA, 2005. doi:10.1109/ICECT.2005.98.

[Gar55] Eugene Garfield. Citation indexes for science: A new dimension in documentation through association of ideas. *Science*, 122(3159):108–111, 1955. doi:10.1126/science.122.3159.108.

[GGMP04] Zoltán Gyöngyi, Hector Garcia-Molina, and Jan Pedersen. Combating web spam with trustrank. In *VLDB '04: Proceedings of the Thirtieth international conference on Very large data bases*, pages 576–587, 2004. Available from: http://portal.acm.org/citation.cfm?doid=1316689.1316740.

[Gio07] Emeric Gioan. Enumerating degree sequences in digraphs and a cycle-cocycle reversing system. *European Journal of Combinatorics*, 28(4):1351 – 1366, 2007. doi:10.1016/j.ejc.2005.11.006.

[GKRT04] R. Guha, Ravi Kumar, Prabhakar Raghavan, and Andrew Tomkins. Propagation of trust and distrust. In *WWW '04: Proceedings of the 13th international conference on World Wide Web*, pages 403–412, New York, NY, USA, 2004. doi:10.1145/988672.988727.

[GMR$^+$07] Arpita Ghosh, Mohammad Mahdian, Daniel M. Reeves, David M. Pennock, and Ryan Fugger. Mechanism design on trust networks. In *WINE '07: Proceedings of the 3rd international workshop on Internet and Network Economics*, pages 257–268, 2007. doi:10.1007/978-3-540-77105-0_25.

[KW81] Daniel J. Kleitman and Kenneth J. Winston. Forests and score vectors. *Combinatorica*, 1(1):49–54, 1981. doi:10.1007/BF02579176.

[LHK10] J. Leskovec, D. Huttenlocher, and J. Kleinberg. Signed networks in social media. In *CHI '10: ACM SIGCHI 28th ACM Conference on Human Factors in Computing Systems*, 2010. arXiv:1003.2424v1.

[LHL+10] Zhengye Liu, Hao Hu, Yong Liu, Keith W. Ross, Yao Wang, and Markus Mobius. P2P trading in social networks: The value of staying connected. In *Proceedings of IEEE INFOCOM 2010*, pages 1–9, March 2010. doi:10.1109/INFCOM.2010.5462077.

[PBMW99] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab, 1999. Available from: http://ilpubs.stanford.edu:8090/422/.

A Estimate of Failure Probability in $G_c(n, p)$

Consider a node u in $G_c(n, p)$ with $c = 1$. If u has degree k, then repeated transactions under Λ induce a one-dimensional random walk of length $k+1$ for u; each of the $(k+1)$ states represents the total credit extended by u's neighbors to u. If u is at state zero, a transaction that has u as a payer will fail, whereas if u is at state k, a transaction involving u as a payee will fail. We estimate (but cannot prove) that the steady-state probability of u being at state zero or state k is $1/(k+1)$ each. Using this, we estimate the steady-state probability of failed transaction as

$$S(n, p) = \sum_{k=0}^{n-1} \frac{2}{k+1} \binom{n-1}{k} p^k (1-p)^{n-1-k}.$$

To compute this quantity, define

$$f(n, x, y) := \sum_{k=0}^{n-1} \frac{2}{k+1} \binom{n-1}{k} x^k y^{n-1-k}.$$

Now,

$$f(n, x, y) = (2/x) \sum_{k=0}^{n-1} \binom{n-1}{k} y^{n-1-k} \int_0^x z^k dz.$$

Exchanging sums and integrations, we get

$$f(n, x, y) = (2/x) \int_0^x \left(\sum_{k=0}^{n-1} \binom{n-1}{k} y^{n-1-k} z^k \right) dz.$$

This gives

$$f(n, x, y) = (2/x) \int_0^x (z+y)^{n-1} dz = \frac{1}{nx(x+y)^n}.$$

Since $S(n, p) = f(n, p, 1-p)$, we obtain

$$S(n, p) = \frac{2}{np}.$$

For a general c, a similar analysis gives the following heuristic bounds on the steady-state transaction failure probability in $G_c(n, p)$:

$$\frac{4}{npc} \geq S(n, p, c) \geq \frac{2}{npc}$$

B Steady-state Success Probability in Trees

We show lower and upper bounds on the steady-state success probability for trees under a uniform transaction regime along with the topologies that achieve those bounds.

Lemma 14. *Let $G^*_{m;c}$ be a star network with m edges each having capacity c and let $\mathcal{M}(G^*_{m;c}, \Lambda)$ be an ergodic Markov chain induced by a uniform transaction rate matrix Λ over nodes in $G^*_{m;c}$. Then the steady-state transaction success probability, $\mathbb{P}_s(G^*_{m;c})$, is given by*

$$\frac{2 + c(m+1)}{m+1} \cdot \frac{c}{(c+1)^2}$$

Proof. Since $\mathcal{M}$ has a uniform steady-state distribution over the states of the network, therefore

$$\mathbb{P}_s(G^*_{m;c}) = \mathbb{E}_l \left(\frac{c}{c+1}\right)^l$$

For a star network of m edges, the probability that $l = 1$ is given by $2m/m(m+1) = 2/(m+1)$. The probability that $l = 2$ is $1 - 2/(m+1) = (m-1)/(m+1)$.Therefore, the steady-state success probability, $\mathbb{P}_s(G^*_{m;c})$, is given by

$$\begin{aligned} \mathbb{P}_s(G^*_{m;c}) &= \frac{2}{m+1}\frac{c}{c+1} + \frac{m-1}{m+1}\left(\frac{c}{c+1}\right)^2 \\ &= \frac{c}{(c+1)(m+1)}\left(2 + \frac{c(m-1)}{c+1}\right) \\ &= \frac{2+c(m+1)}{m+1} \cdot \frac{c}{(c+1)^2} \end{aligned}$$

□

It is easy to see from the above expression that $[c/(c+1)]^2 \leq \mathbb{P}_s(G^*_{m;c}) \leq c/(c+1)$.

Theorem 15. *Let $T_{m;c}$ be a tree with m edges each having capacity c and let $\mathcal{M}(T_{m;c}, \Lambda)$ be an ergodic Markov chain induced by a uniform transaction rate matrix Λ over nodes in $T_{m;c}$. Then the steady-state transaction success probability, $\mathbb{P}_s(T_{m;c})$, is bounded above by*

$$\mathbb{P}_s(T_{m;c}) \leq \frac{2+c(m+1)}{m+1} \cdot \frac{c}{(c+1)^2}$$

Proof. Consider a tree T having $m > 1$ edges. Let $l_{\max}$ be the maximum distance between any two nodes in the tree. Then $l_{max} \geq 2$. We have that for any $l, 1 \leq l \leq l_{max}$, the probability of having a transaction of length at most l in a star network of m edges is greater than the probability of having a transaction of length at most l in T (in other words, the number of (source, destination) pairs having a distance at most l is greater in a star network than in T). Also, the probability of a successful transaction of length l, $[c/(c+1)]^l$, is monotone decreasing in l. Therefore, the success probability in T is bounded above by the success probability in a star network. □

This shows that the star-network yields the best steady-state success probability among all tree networks. Next we prove a lower bound on the steady-state success probability of trees.

Lemma 16. *Let $L_{m;c}$ be a line graph with m edges each having capacity c and let $\mathcal{M}(L_{m;c}, \Lambda)$ be an ergodic Markov chain induced by a uniform transaction rate matrix Λ over nodes in $L_{m;c}$. Then the steady-state transaction success probability, $\mathbb{P}_s(L_{m;c})$, is at least*

$$\frac{2c}{m(m+1)}\left[1 - \left(\frac{c}{c+1}\right)^m\right]$$

Proof. Since $\mathcal{M}$ has a uniform steady-state distribution over the states of the network, therefore

$$\mathbb{P}_s(L_{m;c}) = \mathbb{E}_l \left(\frac{c}{c+1}\right)^l$$

There are $2(m+1-l)$ (source, destination) pairs of length l in a line graph of m edges. Therefore, the success probability of a line network is given by

$$
\begin{aligned}
\mathbb{P}_s(L_{m;c}) &= \sum_{l=1}^{m} \frac{2(m+1-l)}{m(m+1)} \left(\frac{c}{c+1}\right)^l \\
&\geq \sum_{l=1}^{m} \frac{2}{m(m+1)} \left(\frac{c}{c+1}\right)^l \\
&= \frac{2}{m(m+1)} \frac{\frac{c}{c+1}\left(1-\left(\frac{c}{c+1}\right)^m\right)}{1-\frac{c}{c+1}} \\
&= \frac{2c}{m(m+1)} \left[1-\left(\frac{c}{c+1}\right)^m\right]
\end{aligned}
$$

□

Theorem 17. *Let $T_{m;c}$ be a tree with m edges each having capacity c and let $\mathcal{M}(T_{m;c}, \Lambda)$ be an ergodic Markov chain induced by a uniform transaction rate matrix Λ over nodes in $T_{m;c}$. Then the steady-state transaction success probability, $\mathbb{P}_s(T_{m;c})$, is bounded below by*

$$\mathbb{P}_s(T_{m;c}) \geq \frac{2c}{m(m+1)} \left[1-\left(\frac{c}{c+1}\right)^m\right]$$

Proof. Using an argument similar to Theorem 15, we can show that the probability of a transaction of length at most l in a tree T of m edges is higher than the corresponding probability in a line network of m edges for $1 \leq l \leq m$. Therefore, the steady-state success probability in T is bounded below by the success probability in a line network. □