LIQUIDITY IN CREDIT NETWORKS

A Little Trust Goes a Long Way

Pranav Dandekar
Stanford University

(joint work with Ashish Goel, Ramesh Govindan, Ian Post)

NetEcon '10
Vancouver, BC, Canada

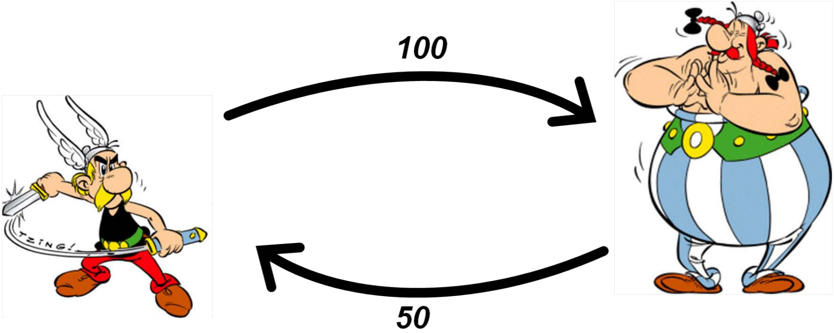
1 INTRODUCTION
- Illustrative Example
- What is a Credit Network?
- Applications

2 LIQUIDITY MODEL & ANALYSIS
- Liquidity Model
- Main Results
- Analysis

3 SIMULATIONS
- Setup
- Results

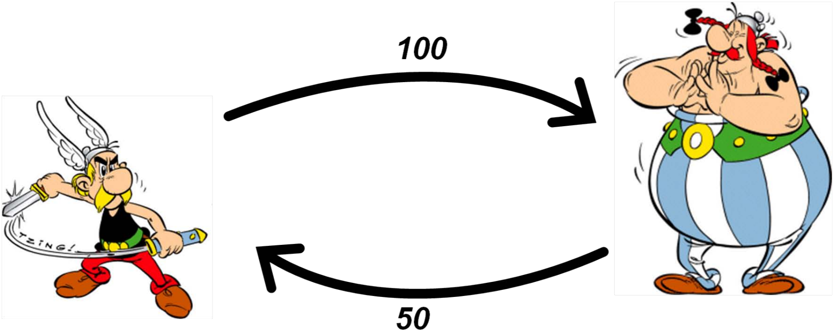
4 SUMMARY

Barter Economy in *Armorica*



Asterix willing to accept up to 100 IOUs from Obelix
Obelix willing to accept up to 50 IOUs from Asterix

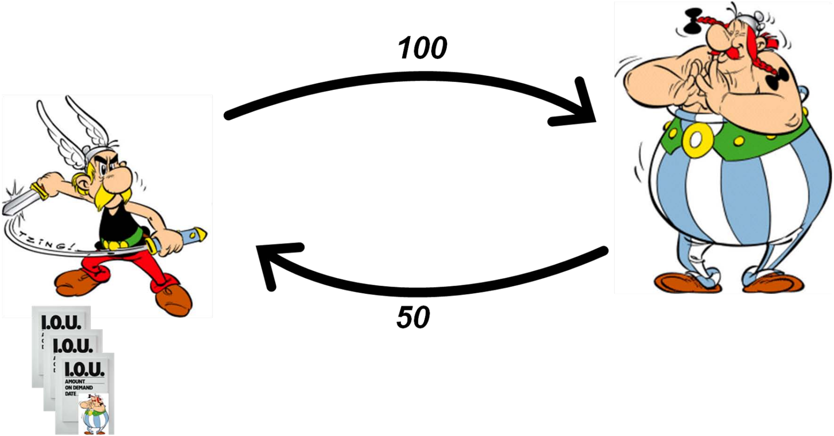
Barter Economy in *Armorica*



Asterix willing to accept up to 100 IOUs from Obelix
Obelix willing to accept up to 50 IOUs from Asterix

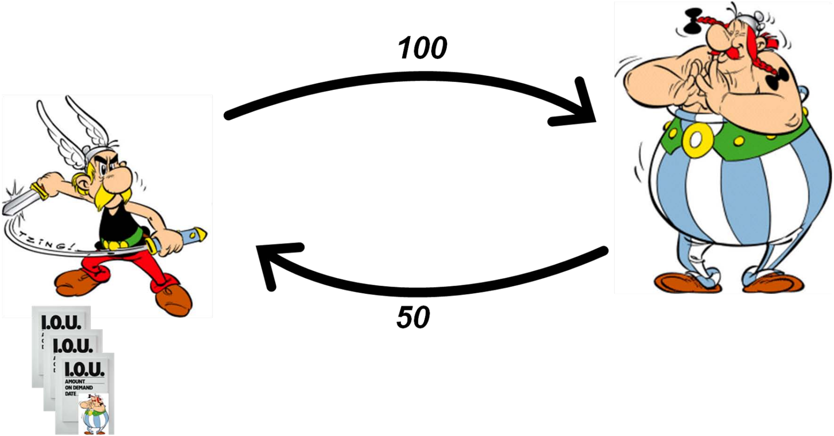
Barter Economy in *Armorica*

Obelix needs a favor worth 10 IOUs.



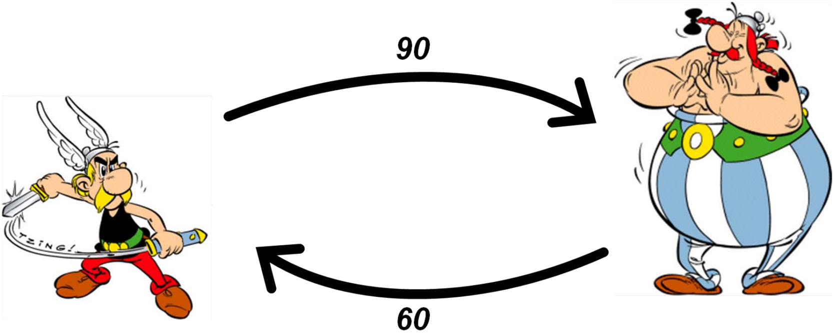
Gives Asterix 10 IOUs as payment.

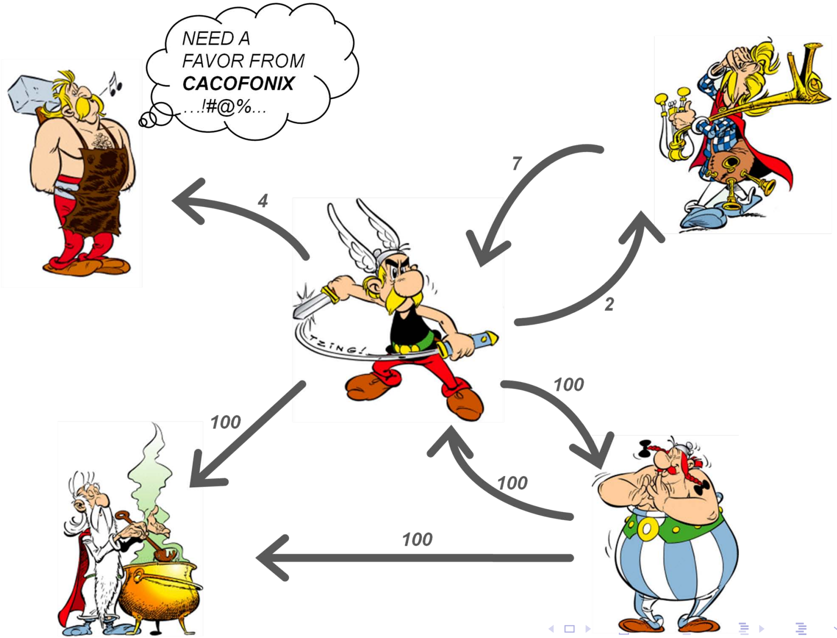
Obelix needs a favor worth 10 IOUs.



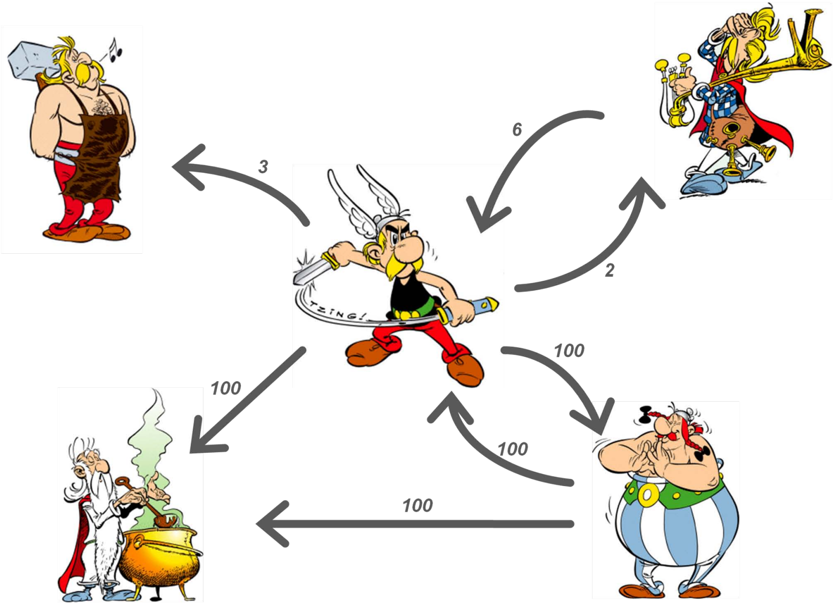
Gives Asterix 10 IOUs as payment.

New trust values....

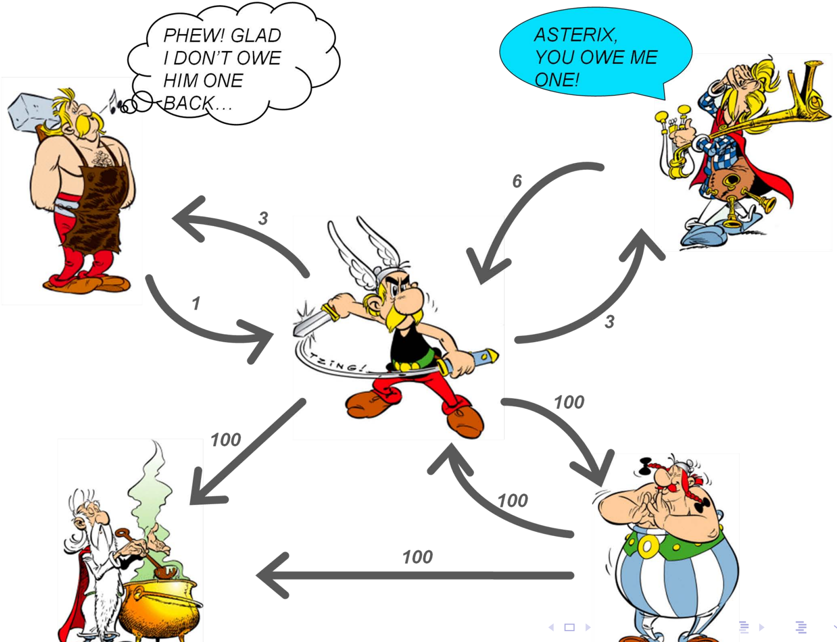



NEED A
FAVOR FROM
CACOFONIX
....!#@%...
4
7
2
TZING!
100
100
100
100

Barter Economy in *Armorica*



Barter Economy in *Armorica*



What is a Credit Network?

- Decentralized payment infrastructure introduced by [DeFigueiredo, Barr, 2005] and [Ghosh et. al., 2007]
- Do not need banks, common currency
- Models trust in networked interactions

What is a Credit Network?

- Graph $G(V, E)$ represents a network (social network, p2p network, etc.)
- **Nodes:** (non-rational) agents/players; print their own currency
- **Edges:** credit limits $c_{uv} > 0$ extended by nodes to each other[1]
- Payments made by passing IOUs along a chain of trust
- Credit gets replenished when payments are made in the other direction

[1] assume all currency exchange ratios to be unity

What is a Credit Network?

- Graph $G(V, E)$ represents a network (social network, p2p network, etc.)
- **Nodes:** (non-rational) agents/players; print their own currency
- **Edges:** credit limits $c_{uv} > 0$ extended by nodes to each other[1]
- Payments made by passing IOUs along a chain of trust
- Credit gets replenished when payments are made in the other direction

[1] assume all currency exchange ratios to be unity

What is a Credit Network?

- Graph $G(V, E)$ represents a network (social network, p2p network, etc.)
- **Nodes:** (non-rational) agents/players; print their own currency
- **Edges:** credit limits $c_{uv} > 0$ extended by nodes to each other[1]
- Payments made by passing IOUs along a chain of trust
- Credit gets replenished when payments are made in the other direction

[1] assume all currency exchange ratios to be unity

What is a Credit Network?

- Graph $G(V, E)$ represents a network (social network, p2p network, etc.)
- **Nodes:** (non-rational) agents/players; print their own currency
- **Edges:** credit limits $c_{uv} > 0$ extended by nodes to each other[1]
- Payments made by passing IOUs along a chain of trust
- Credit gets replenished when payments are made in the other direction

[1] assume all currency exchange ratios to be unity

What is a Credit Network?

- Graph $G(V,E)$ represents a network (social network, p2p network, etc.)
- **Nodes:** (non-rational) agents/players; print their own currency
- **Edges:** credit limits $c_{uv} > 0$ extended by nodes to each other[1]
- Payments made by passing IOUs along a chain of trust
- Credit gets replenished when payments are made in the other direction

[1] assume all currency exchange ratios to be unity

- Barter/Exchange economies like P2P networks.
- Combating social spam (Facebook, LinkedIn)
- Distributing proxy addresses to circumvent censorship in repressive regimes

- Edges have integer capacity $c > 0$
- Transaction rate matrix $\Lambda = \{\lambda_{uv} : u, v \in V, \lambda_{uu} = 0\}$
- Repeated transactions; at each time step choose (s, t) with prob. λ_{st}
- Try to route a unit payment from s to t via the shortest feasible path; **update edge capacities** along the path
- Transaction fails if no path exists

Markov Chain

- Repeated transactions induce a Markov chain $\mathcal{M}$ with $(c+1)^m$ states
- State $\mathcal{S}$ of $\mathcal{M}$ captures the states of all edges in G
- Transition probability $P(\mathcal{S}, \mathcal{S}') = \lambda_{st}$, where $s \to t$ in $\mathcal{S}$ leads to $\mathcal{S}'$
- $P(\mathcal{S}, \mathcal{S})$:= failure prob. at state $\mathcal{S}$

Questions

- Steady-state distribution?
- Steady-state transaction success probability?
- Comparison with a centralized payment infrastructure

Markov Chain

- Repeated transactions induce a Markov chain $\mathcal{M}$ with $(c+1)^m$ states
- State $\mathcal{S}$ of $\mathcal{M}$ captures the states of all edges in G
- Transition probability $P(\mathcal{S}, \mathcal{S}') = \lambda_{st}$, where $s \rightarrow t$ in $\mathcal{S}$ leads to $\mathcal{S}'$
- $P(\mathcal{S}, \mathcal{S}) :=$ failure prob. at state $\mathcal{S}$

Questions

- Steady-state distribution?
- Steady-state transaction success probability?
- Comparison with a centralized payment infrastructure

- Success probability independent of path along which transactions are routed
- For symmetric transaction rates, the success probability for
  - **Complete Graphs:** Goes to one with increase in network size or credit capacity.
  - $G_c(n, p)$ **networks** ($p > \ln n/n$)**:** Goes to one with increase in one of n, p or c keeping the other two constant.
  - **PA networks:** Goes to one with increase in avg. node degree or credit capacity (indepedent of network size).
- Success probability in Complete graphs and Erdös-Rényi graphs only constant-factor worse than equiv. centralized payment system.

Main Results

- Success probability independent of path along which transactions are routed
- For symmetric transaction rates, the success probability for
  - **Complete Graphs:** Goes to one with increase in network size or credit capacity.
  - $G_c(n, p)$ **networks** ($p > \ln n/n$)**:** Goes to one with increase in one of n, p or c keeping the other two constant.
  - **PA networks:** Goes to one with increase in avg. node degree or credit capacity (indepedent of network size).
- Success probability in Complete graphs and Erdös-Rényi graphs only constant-factor worse than equiv. centralized payment system.

- Success probability independent of path along which transactions are routed
- For symmetric transaction rates, the success probability for
  - **Complete Graphs:** Goes to one with increase in network size or credit capacity.
  - $G_c(n, p)$ **networks** ($p > \ln n/n$)**:** Goes to one with increase in one of n, p or c keeping the other two constant.
  - **PA networks:** Goes to one with increase in avg. node degree or credit capacity (indepedent of network size).
- Success probability in Complete graphs and Erdös-Rényi graphs only constant-factor worse than equiv. centralized payment system.

Definition

Let $\mathcal{S}$ and $\mathcal{S}'$ be two states of the network. 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).*

Transactions along a feasible cycle are "free".

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 is routed along the shortest feasible path from s_i to t_i. Then the same set of transactions succeed when the payment is routed along any feasible path from s_i to t_i.

Proof Sketch.

Proof by induction on T.

$\mathcal{S}_k :=$ state of the network when transactions $(s_1, t_1), \ldots, (s_k, t_k)$ are routed along the shortest feasible path

$\mathcal{S}'_k :=$ state of the network when not all of the transactions $(s_1, t_1), \ldots, (s_k, t_k)$ are routed along the shortest feasible path

From $\mathcal{S}'_k$ undo transactions $(s_k, t_k), (s_{k-1}, t_{k-1}), \ldots, (s_1, t_1)$ and redo $(s_1, t_1), \ldots, (s_k t_k)$ along their shortest feasible paths. This results in state $\mathcal{S}$.

But undoing and redoing is equal to k transactions along cycles. Therefore, $\mathcal{S}_k$ and $\mathcal{S}'_k$ are cycle-reachable.

So if (s_{k+1}, t_{k+1}) is feasible in state $\mathcal{S}_k$, it is also feasible in state $\mathcal{S}'_k$. □

Proof Sketch.

Proof by induction on T.

$\mathcal{S}_k :=$ state of the network when transactions $(s_1, t_1), \ldots, (s_k, t_k)$ are routed along the shortest feasible path

$\mathcal{S}'_k :=$ state of the network when not all of the transactions $(s_1, t_1), \ldots, (s_k, t_k)$ are routed along the shortest feasible path

From $\mathcal{S}'_k$ undo transactions $(s_k, t_k), (s_{k-1}, t_{k-1}), \ldots, (s_1, t_1)$ and redo $(s_1, t_1), \ldots, (s_k t_k)$ along their shortest feasible paths. This results in state $\mathcal{S}$.

But undoing and redoing is equal to k transactions along cycles. Therefore, $\mathcal{S}_k$ and $\mathcal{S}'_k$ are cycle-reachable.

So if (s_{k+1}, t_{k+1}) is feasible in state $\mathcal{S}_k$, it is also feasible in state $\mathcal{S}'_k$. □

Proof Sketch.

Proof by induction on T.

$\mathcal{S}_k :=$ state of the network when transactions $(s_1, t_1), \ldots, (s_k, t_k)$ are routed along the shortest feasible path

$\mathcal{S}'_k :=$ state of the network when not all of the transactions $(s_1, t_1), \ldots, (s_k, t_k)$ are routed along the shortest feasible path

From $\mathcal{S}'_k$ undo transactions $(s_k, t_k), (s_{k-1}, t_{k-1}), \ldots, (s_1, t_1)$ and redo $(s_1, t_1), \ldots, (s_k t_k)$ along their shortest feasible paths. This results in state $\mathcal{S}$.

But undoing and redoing is equal to k transactions along cycles.

Therefore, $\mathcal{S}_k$ and $\mathcal{S}'_k$ are cycle-reachable.

So if (s_{k+1}, t_{k+1}) is feasible in state $\mathcal{S}_k$, it is also feasible in state $\mathcal{S}'_k$. □

Proof Sketch.

Proof by induction on T.

$\mathcal{S}_k :=$ state of the network when transactions $(s_1, t_1), \ldots, (s_k, t_k)$ are routed along the shortest feasible path

$\mathcal{S}'_k :=$ state of the network when not all of the transactions $(s_1, t_1), \ldots, (s_k, t_k)$ are routed along the shortest feasible path

From $\mathcal{S}'_k$ undo transactions $(s_k, t_k), (s_{k-1}, t_{k-1}), \ldots, (s_1, t_1)$ and redo $(s_1, t_1), \ldots, (s_k t_k)$ along their shortest feasible paths. This results in state $\mathcal{S}$.

But undoing and redoing is equal to k transactions along cycles.

Therefore, $\mathcal{S}_k$ and $\mathcal{S}'_k$ are cycle-reachable.

So if (s_{k+1}, t_{k+1}) is feasible in state $\mathcal{S}_k$, it is also feasible in state $\mathcal{S}'_k$. □

Proof Sketch.

Proof by induction on T.
$\mathcal{S}_k :=$ state of the network when transactions $(s_1, t_1), \ldots, (s_k, t_k)$ are routed along the shortest feasible path
$\mathcal{S}'_k :=$ state of the network when not all of the transactions $(s_1, t_1), \ldots, (s_k, t_k)$ are routed along the shortest feasible path
From $\mathcal{S}'_k$ undo transactions $(s_k, t_k), (s_{k-1}, t_{k-1}), \ldots, (s_1, t_1)$ and redo $(s_1, t_1), \ldots, (s_k t_k)$ along their shortest feasible paths. This results in state $\mathcal{S}$.
But undoing and redoing is equal to k transactions along cycles.
Therefore, $\mathcal{S}_k$ and $\mathcal{S}'_k$ are cycle-reachable.
So if (s_{k+1}, t_{k+1}) is feasible in state $\mathcal{S}_k$, it is also feasible in state $\mathcal{S}'_k$. □

Cycle-reachability induces a partition $\mathcal{C}$ on the set of states in $\mathcal{M}$.

FACT

For any equivalence class $C \in \mathcal{C}$, if a transaction (s, t) is feasible in some state $\mathcal{S} \in C$, it is feasible for all states $\mathcal{S}' \in C$ (since $\mathcal{S}$ is cycle-reachable from $\mathcal{S}'$).

FACT

If a transaction (s, t) is feasible in two states $\mathcal{S}_i, \mathcal{S}_j \in C$ and results in transitions to states $\mathcal{S}_i'$ and $\mathcal{S}_j'$ respectively, then $\mathcal{S}_i'$ and $\mathcal{S}_j'$ are cycle-reachable (in other words, belong to the same equivalence class).

COROLLARY

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.

Cycle-reachability induces a partition $\mathcal{C}$ on the set of states in $\mathcal{M}$.

Fact

For any equivalence class $C \in \mathcal{C}$, if a transaction (s, t) is feasible in some state $\mathcal{S} \in C$, it is feasible for all states $\mathcal{S}' \in C$ (since $\mathcal{S}$ is cycle-reachable from $\mathcal{S}'$).

Fact

If a transaction (s, t) is feasible in two states $\mathcal{S}_i, \mathcal{S}_j \in C$ and results in transitions to states $\mathcal{S}_i'$ and $\mathcal{S}_j'$ respectively, then $\mathcal{S}_i'$ and $\mathcal{S}_j'$ are cycle-reachable (in other words, belong to the same equivalence class).

Corollary

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.

Cycle-reachability induces a partition $\mathcal{C}$ on the set of states in $\mathcal{M}$.

Fact

For any equivalence class $C \in \mathcal{C}$, if a transaction (s, t) is feasible in some state $\mathcal{S} \in C$, it is feasible for all states $\mathcal{S}' \in C$ (since $\mathcal{S}$ is cycle-reachable from $\mathcal{S}'$).

Fact

If a transaction (s, t) is feasible in two states $\mathcal{S}_i, \mathcal{S}_j \in C$ and results in transitions to states $\mathcal{S}_i'$ and $\mathcal{S}_j'$ respectively, then $\mathcal{S}_i'$ and $\mathcal{S}_j'$ are cycle-reachable (in other words, belong to the same equivalence class).

Corollary

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.

Cycle-reachability induces a partition $\mathcal{C}$ on the set of states in $\mathcal{M}$.

FACT

For any equivalence class $C \in \mathcal{C}$, if a transaction (s, t) is feasible in some state $\mathcal{S} \in C$, it is feasible for all states $\mathcal{S}' \in C$ (since $\mathcal{S}$ is cycle-reachable from $\mathcal{S}'$).

FACT

If a transaction (s, t) is feasible in two states $\mathcal{S}_i, \mathcal{S}_j \in C$ and results in transitions to states $\mathcal{S}'_i$ and $\mathcal{S}'_j$ respectively, then $\mathcal{S}'_i$ and $\mathcal{S}'_j$ are cycle-reachable (in other words, belong to the same equivalence class).

COROLLARY

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.

Theorem

Consider a Markov chain $\mathcal{M}_{\mathcal{S}_0}$ starting in state $\mathcal{S}_0$ induced by a symmetric transaction rate matrix Λ. 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.

$\mathcal{T}_{ij} := \{(s,t) \mid s \to t \text{ in state } \mathcal{S} \in C_i \text{ leads to state } \mathcal{S}' \in C_j\}$

Define transition probability between $C_i, C_j \in \mathcal{C}_{\mathcal{S}_0}$ as

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

Since $(s,t) \in \mathcal{T}_{ij} \Leftrightarrow (t,s) \in \mathcal{T}_{ji}$ and Λ is symmetric, therefore P is a symmetric stochastic matrix.

$\Longrightarrow$ uniform distribution over $\mathcal{C}_{\mathcal{S}_0}$ is stationary w.r.t. P. □

Corollary

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

Proof.

$\mathcal{T}_{ij} := \{(s, t) \mid s \rightarrow t \text{ in state } \mathcal{S} \in C_i \text{ leads to state } \mathcal{S}' \in C_j\}$
Define transition probability between $C_i, C_j \in \mathcal{C}_{\mathcal{S}_0}$ as

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

Since $(s, t) \in \mathcal{T}_{ij} \Leftrightarrow (t, s) \in \mathcal{T}_{ji}$ and Λ is symmetric, therefore P is a symmetric stochastic matrix.
$\Longrightarrow$ uniform distribution over $\mathcal{C}_{\mathcal{S}_0}$ is stationary w.r.t. P. □

Corollary

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

Proof.

$\mathcal{T}_{ij} := \{(s,t) \mid s \to t \text{ in state } \mathcal{S} \in C_i \text{ leads to state } \mathcal{S}' \in C_j\}$

Define transition probability between $C_i, C_j \in \mathcal{C}_{\mathcal{S}_0}$ as

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

Since $(s,t) \in \mathcal{T}_{ij} \Leftrightarrow (t,s) \in \mathcal{T}_{ji}$ and Λ is symmetric, therefore P is a symmetric stochastic matrix.

$\implies$ uniform distribution over $\mathcal{C}_{\mathcal{S}_0}$ is stationary w.r.t. P. □

Corollary

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

Proof.

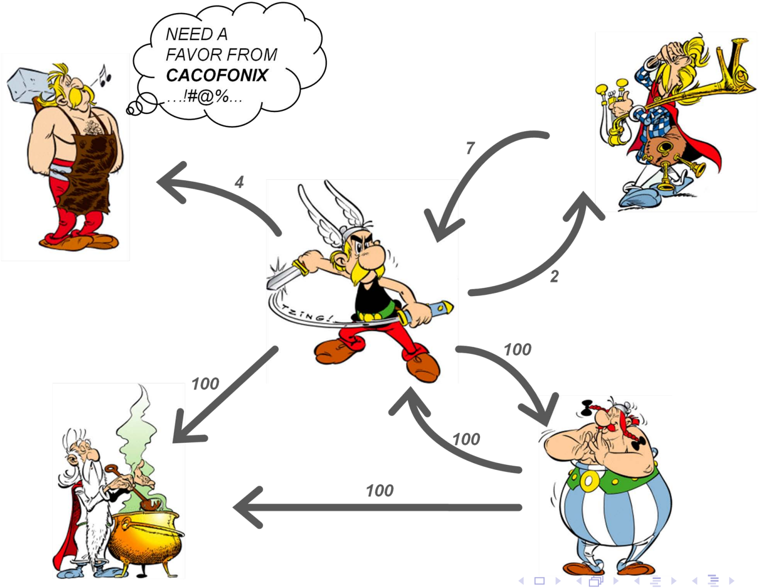
$\mathcal{T}_{ij} := \{(s,t) \mid s \to t$ in state $\mathcal{S} \in C_i$ leads to state $\mathcal{S}' \in C_j\}$
Define transition probability between $C_i, C_j \in \mathcal{C}_{\mathcal{S}_0}$ as

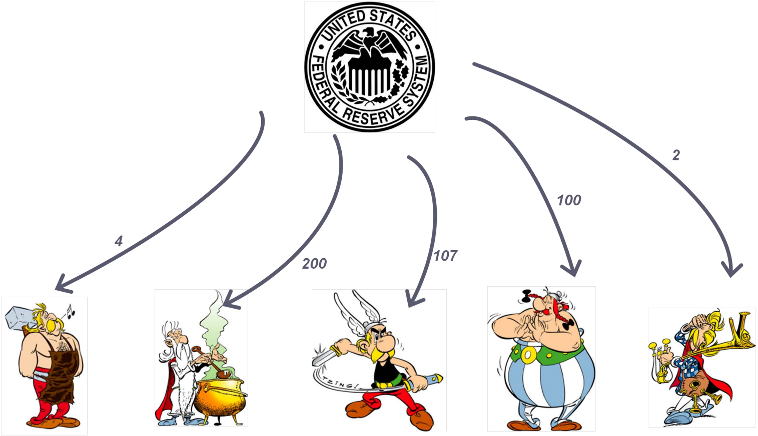
$$P(C_i, C_j) = \sum_{(s,t) \in \mathcal{T}_{ij}} \lambda_{st}$$

Since $(s,t) \in \mathcal{T}_{ij} \Leftrightarrow (t,s) \in \mathcal{T}_{ji}$ and Λ is symmetric, therefore P is a symmetric stochastic matrix.
$\implies$ uniform distribution over $\mathcal{C}_{\mathcal{S}_0}$ is stationary w.r.t. P. $\square$

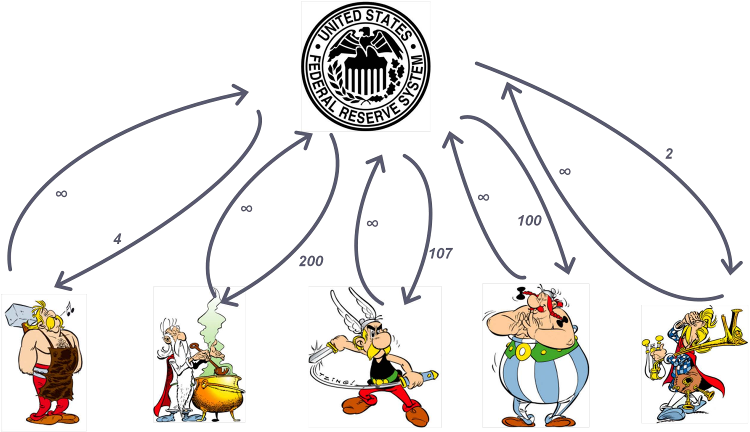
Corollary

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


NEED A
FAVOR FROM
CACOFONIX
...!#@%...
4
7
2
TZING!
100
100
100
100



UNITED STATES
FEDERAL RESERVE SYSTEM
4
200
107
100
2



UNITED STATES
FEDERAL RESERVE SYSTEM
∞
4
∞
200
∞
107
∞
100
∞
2
TZING!

Convert Credit Network → Centralized Model

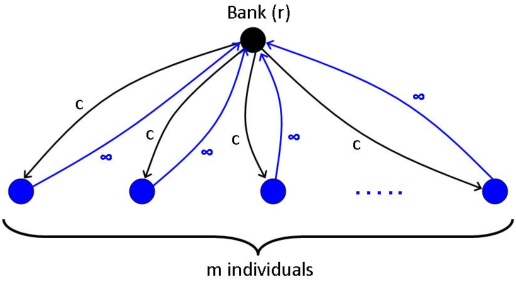
$$\forall u, c_{ru} = \sum_{v} c_{vu}$$

$\implies$ Total credit in the system is conserved during conversion

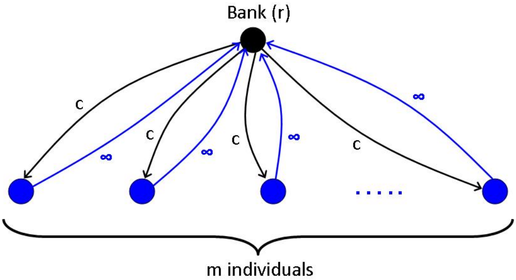
Convert Credit Network → Centralized Model

$$\forall u, c_{ru} = \sum_{v} c_{vu}$$

$\implies$ Total credit in the system is conserved during conversion



$\mathcal{M}$ has $\binom{mc+m-1}{m-1}$ states



$\mathcal{M}$ has $\binom{mc+m-1}{m-1}$ states

Theorem

If $\mathcal{M}$ is ergodic and Λ is symmetric, then $\mathcal{M}$ has a uniform steady-state distribution.

Corollary

If $\mathcal{M}$ is ergodic and Λ is symmetric, then the steady-state success probability is $c/(c+1)$.

Theorem

If $\mathcal{M}$ is ergodic and Λ is symmetric, then $\mathcal{M}$ has a uniform steady-state distribution.

Corollary

If $\mathcal{M}$ is ergodic and Λ is symmetric, then the steady-state success probability is $c/(c+1)$.

	Credit Network	Centralized System
Star-network	$\Theta(1/c)$	$\Theta(1/c)$
Complete Graph	$\Theta(1/nc)$	$\Theta(1/nc)$
$G_c(n, p)$[2]	$\Theta(1/npc)$	$\Theta(1/npc)$

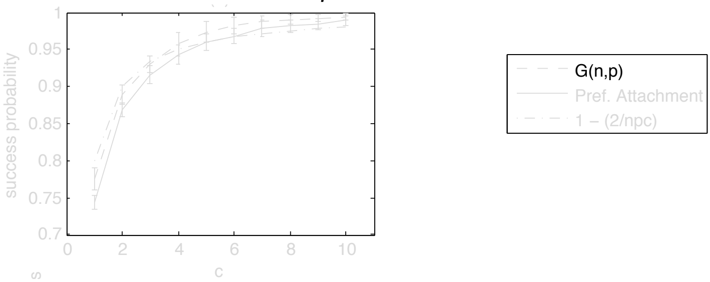
Table: Steady-state Failure Probability in Credit Network v/s Centralized System

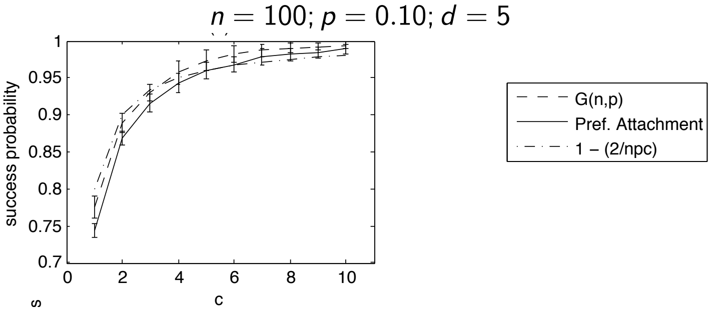
[2] bankruptcy probability

SETUP

- Repeated transactions on $G_c(n, p)$ and PA graphs.
- Stopping criterion: success-rate in consecutive time windows $\leq \epsilon$
- Studied effect of varying network size, network density, and credit capacity
- For each run, recorded following metrics:
  - Number of (weakly) connected components
  - Avg. path length of successful transactions
  - Number of "sink" / "source" nodes
- Averaged metrics over 100 runs

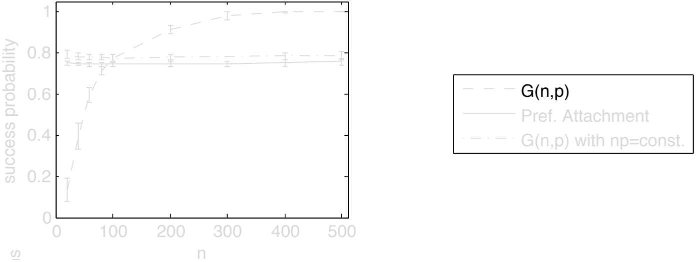
$n = 100; p = 0.10; d = 5$

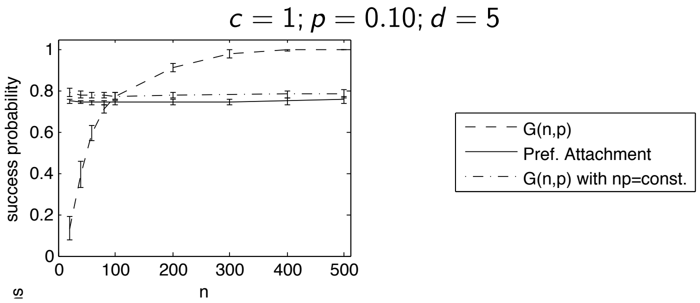

1
0.95
0.9
0.85
0.8
0.75
0.7
success probability
s
0
2
4
6
8
10
c
G(n,p)
Pref. Attachment
1 – (2/npc)



$n = 100; p = 0.10; d = 5$
success probability
1
0.95
0.9
0.85
0.8
0.75
0.7
0
2
4
6
8
10
c
G(n,p)
Pref. Attachment
1 – (2/npc)

$c = 1; p = 0.10; d = 5$


success probability
n
1
0.8
0.6
0.4
0.2
0
0
100
200
300
400
500
G(n,p)
Pref. Attachment
G(n,p) with np=const.


$c = 1; p = 0.10; d = 5$
success probability
n
0
0.2
0.4
0.6
0.8
1
0
100
200
300
400
500
G(n,p)
Pref. Attachment
G(n,p) with np=const.

- Effect of node failures on liquidity and how it varies with network topology
- Effect of non-zero payment routing fees on liquidity
- Endow nodes with rationality: how do nodes initialize and update trust values?

Questions?

- Effect of node failures on liquidity and how it varies with network topology
- Effect of non-zero payment routing fees on liquidity
- Endow nodes with rationality: how do nodes initialize and update trust values?

Questions?

Dimitri B. DeFigueiredo and Earl T. Barr
Trustdavis: A non-exploitable online reputation system, CEC 2005

Arpita Ghosh, Mohammad Mahdian, Daniel M. Reeves, David M. Pennock, and Ryan Fugger
Mechanism design on trust networks, WINE 2007.

Mohammad Mahdian
Fighting censorship with algorithms, FUN 2010.