Dynamic model of funding in interbank payment systems

Marco Galbiati
Bank of England

Kimmo Soramäki
Helsinki Univ. of Technology / ECB

BoF - Simulation seminar: August 28, 2007

Overview

- We develop a dynamic model of an RTGS interbank payment system with endogenous choices for funding by banks
- Banks have knowledge of settlement costs given their own liquidity and liquidity of other banks
- They learn about the behavior of other banks, and choose their own liquidity to minimize costs – given expected behavior of others.
- We look at both normal operating conditions and an operational failure

The work ties to two lines of research

- “Simulations”
  - Koponen-Soramaki (1998), Leinonen, ed. (2005).
  - Work at FRB, ECB, BoC, BoJ, BoE (FSR, June 2004), etc.
  - Use actual payment data and investigate alternative scenarios: effect on payment delays, liquidity needs, and risks
- “Game theoretic models”
  - Angelini (1998) and Kobayakawa (1997), Bech-Garrat (2003), Buckle and Campbell (2003), Willison (2004)
  - Investigate a "liquidity management game" to analyze intraday liquidity management behavior of banks in a RTGS (and DNS) environment

Previous works: problems and advantages

- Simulations have so far not endogenized bank behaviour
  - behaviour has been assumed to remain unchanged in spite of other changes in the system
  - or to change in a predetermined manner
  - due to the use of actual data, difficult to generalize

- Game theoretic models need to make many simplifying assumptions
  - on settlement process / time horizon
  - topology of interactions
  - do not give quantitative answers

- We model endogenous bank behaviour with a more realistic topology and settlement process.

Settlement algorithm

payment order to pay *j* arrives to *i*

payment order to pay *j* arrives to *i*

if *i* has funds the order is settled : j receives funds

if *j* has queued payments, the first one (say to *k*) is settled

if *k* has queued payments, the first one (to ...) is settled

... cascade ends when the recipient of the payment has no queued payments

else, the order is queued

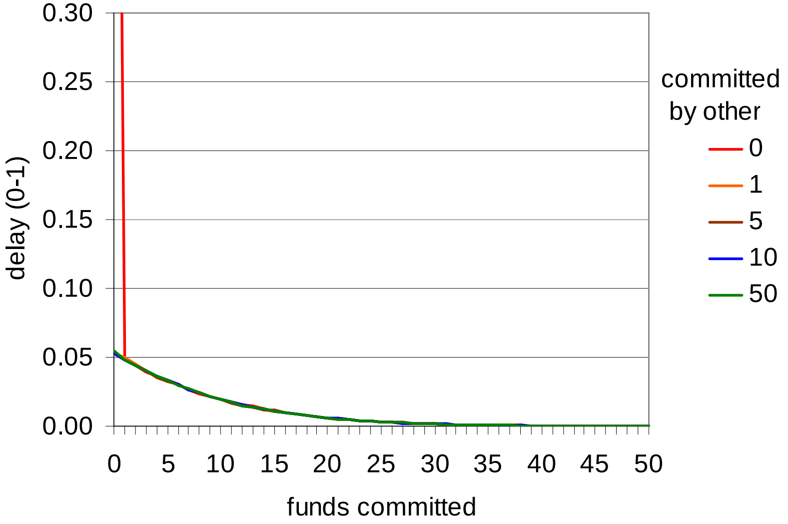
...

the algorithm is run 30 million times to numerically explore it

In the model payment order arrive according to a Poisson process. Each bank is equally likely as recipient -> **homogenous banks, complete network topology**

Settlement algorithm (cont'd)

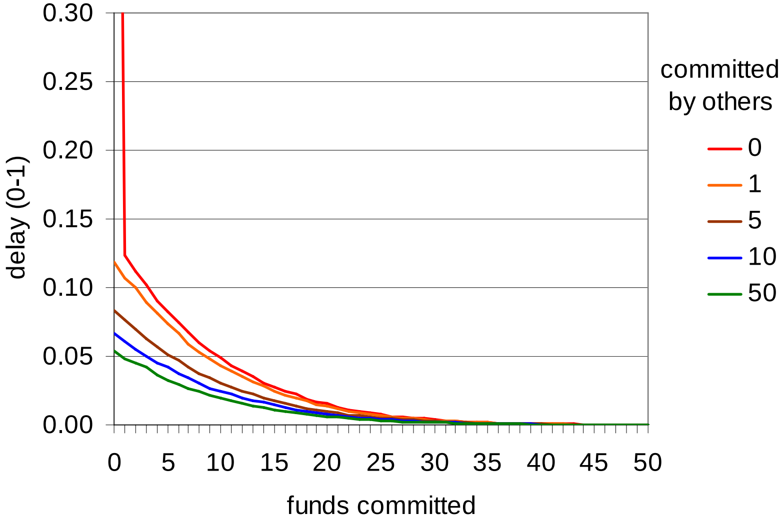
- With only two banks, liquidity of other bank does not matter
  - any liquidity sent out is quickly returned to sending bank
  - other bank has liquidity immediately when you send a payment
- Jump at liquidity choices (0,0)
  - If no liquidity – no settlement
  - already one unit by either bank allows good performance



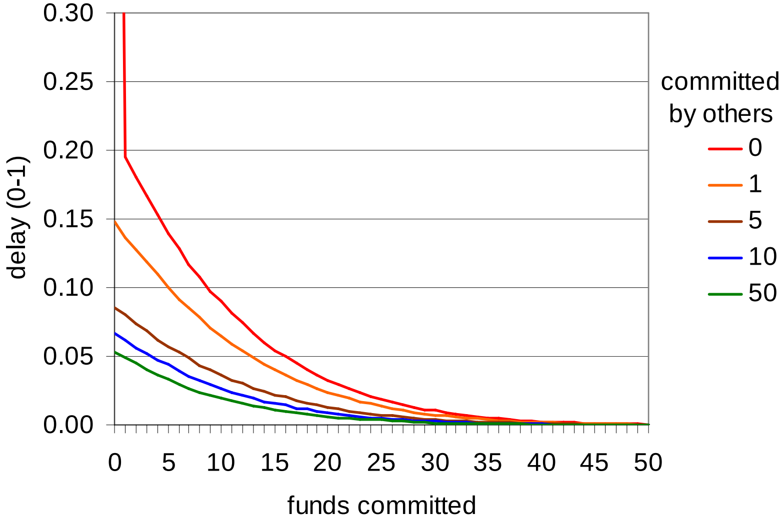
n=2, each bank sends 200 payments

Settlement algorithm (cont'd)

- With more banks, others' liquidity is important
  - liquidity sent has a higher probability of going to (and staying at) other banks
  - more dependence on liquidity of other banks
- Distribution of others' liquidity does not matter much, only total level



n=5, each bank sends 200 payments



n=15, each bank sends 200 payments

The liquidity game

- <u>Actions</u>: banks choose liquidity at the beginning of the day

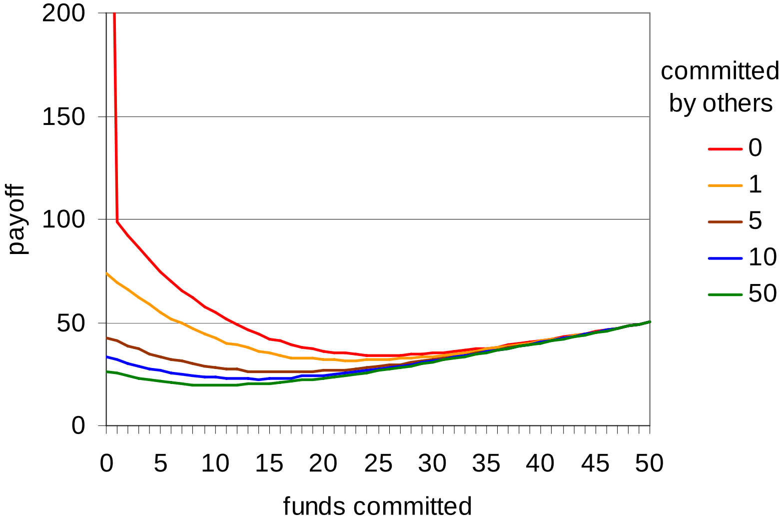
- <u>Outcome</u>: delays, determined by the settlement algorithm

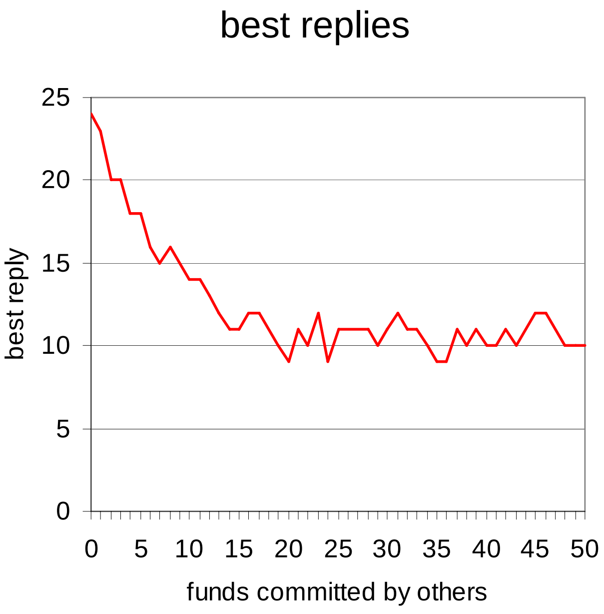
- <u>Costs</u>: banks incur costs that depend on
  - a) chosen liquidity
  - b) delays
  - $Cost = \lambda ai + \kappa \Sigma_r [t'(x_r) - t(x_r)]$

r indicizes payments

Illustration of costs and best replies

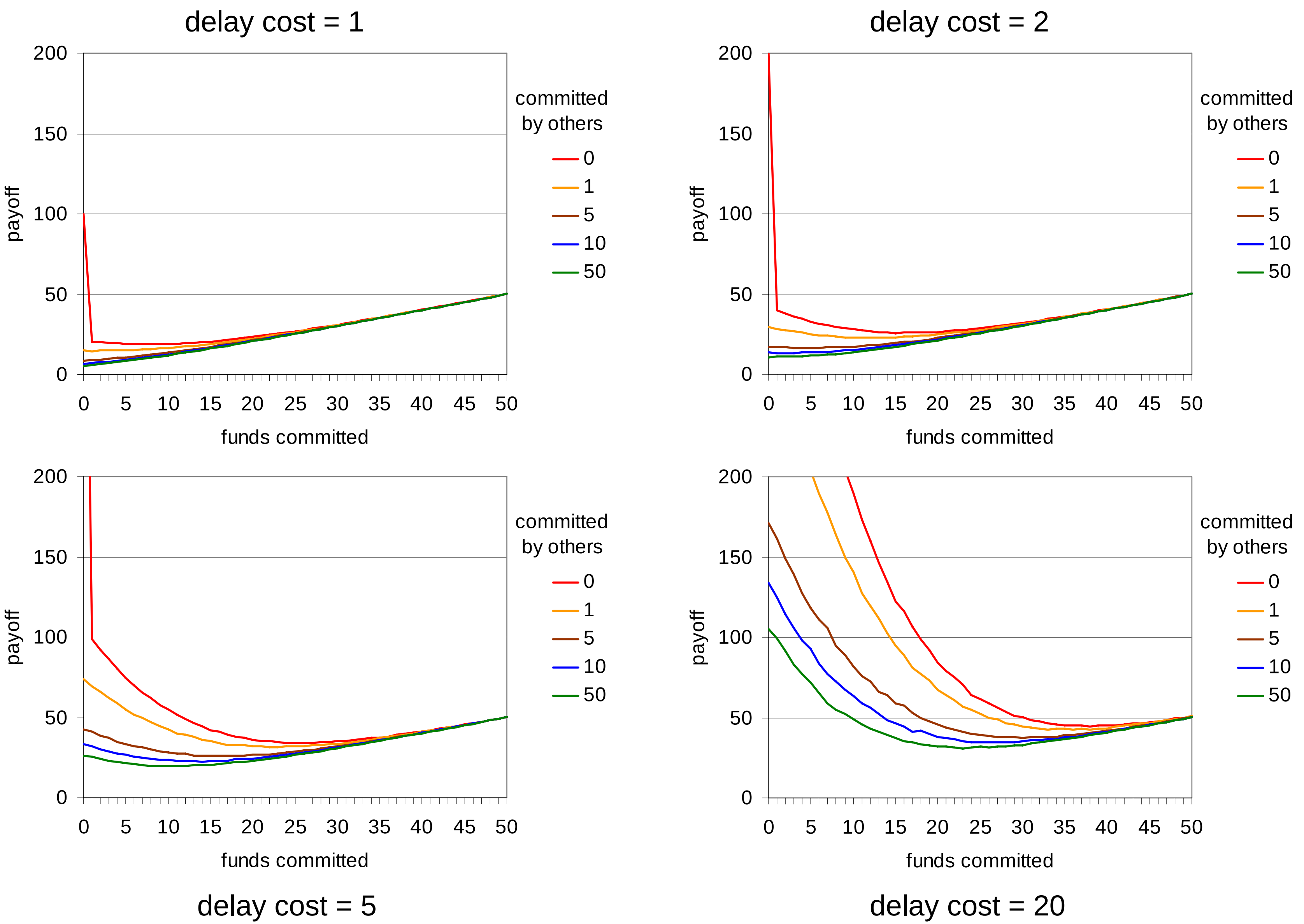
- Costs are minimized at different liquidity levels, depending on liquidity posted by other banks, e.g. (for n=15, delay cost=5)
  - if others post 1, I should post 24
  - if others post 5, I should post 15
  - if others post 50, I should post 10





for n=15

Total costs with alternative delay costs



Learning in the model

- In the model
  - Banks face uncertainty about the actions of other banks
  - Banks adapt their actions over time, depending on observed actions by others
  - This is modeled as fictitious play with given payoff functions
  - The game is played until convergence of beliefs takes place

- Properties of Fictitious play
  - If beliefs converge to 1 for some action, that action is a pure Nash-equilibrium
  - If beliefs converge to a distribution, then that distribution is the mixed Nash equilibrium of the game

- Our results
  - Beliefs converge mostly to a distribution, sometimes to a pure equilibrium
  - Results report weighted average in case of mixed equilibria

Fictitious play - example

	0	1
0	-10 ; -10	0 ; -3
1	-3 ; 0	-2 ; -2

“grab the dollar” or
“hawk-dove” game

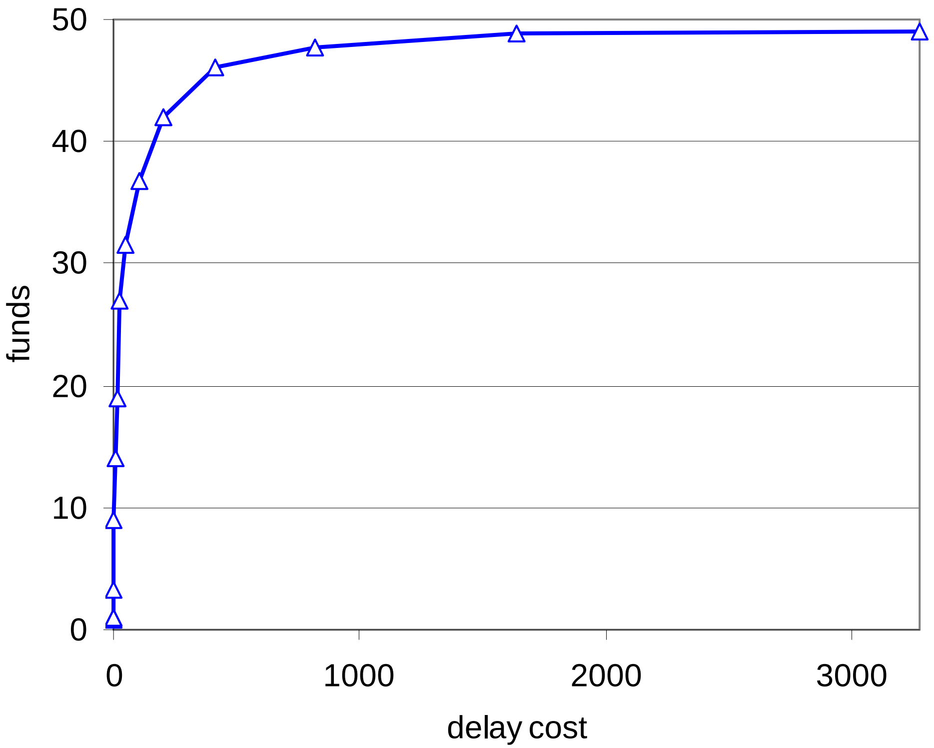
- Payoffs reflect situation with 15 banks and low liquidity costs – it turns out only choices of 0 and 1 are relevant

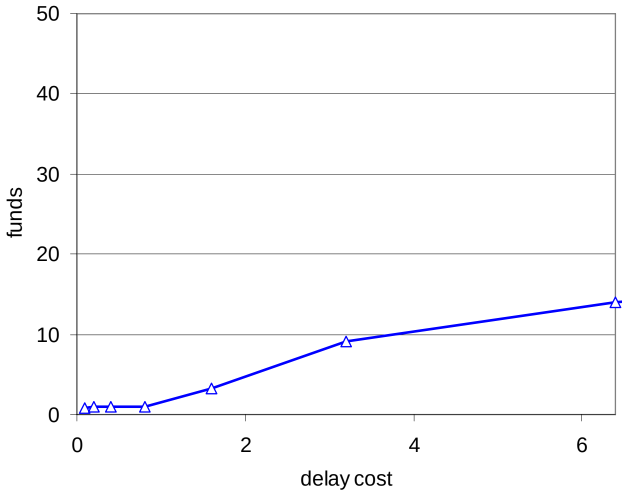
- The game begins with banks assuming equal probabilities for others’ action
  - expected payoff from 0: 0.5*(-10) + 0.5*(0) = -5
  - expected payoff from 1: 0.5*(-3) + 0.5*(-2) = -2.5 (chosen)

- Beliefs are updated on the basis of observed choices
  - e.g. with Bayes rule for updating beliefs, after 100 rounds, 0 is chosen 22 times and 1 is chosen 78 times
  - this is a mixed Nash-equilibrium
  - can be interpreted as the equilibrium probabilities for choices in an infinite game (22% vs 78%)

Results 1 – base case

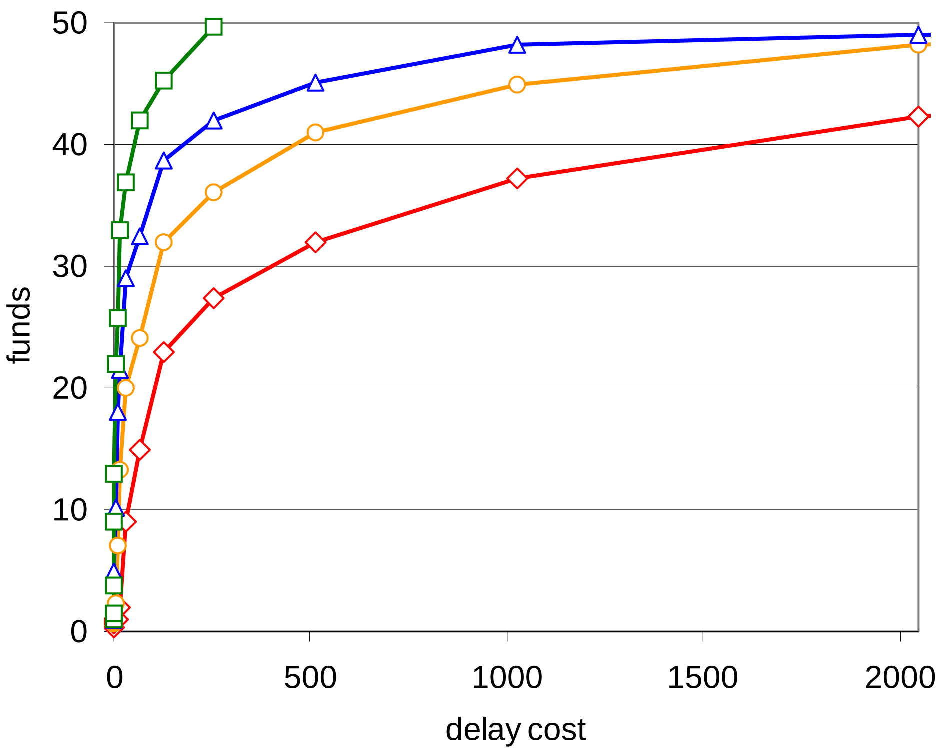
- Banks (naturally) use more liquidity when delay costs are high
- The amount used increases rapidly as delay cost is increased from 0
- At cost parity, banks post exactly 1 unit
- Banks will not post over 49 units, irrespective of delay cost

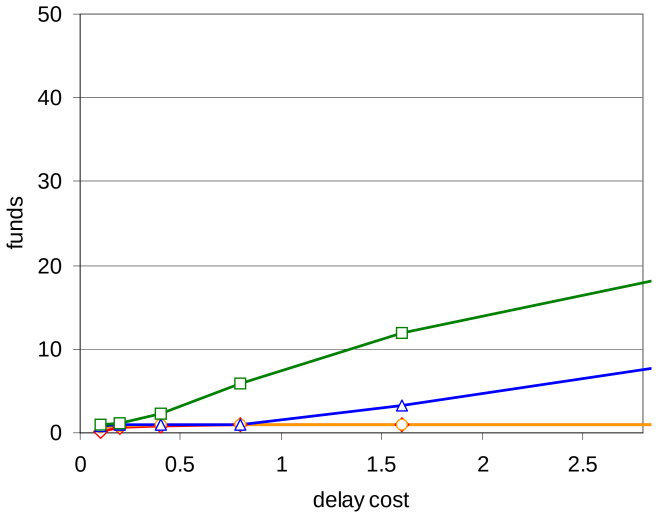




Results 2 - system size

- Banks post more liquidity for a given payment volume, the more other banks there are in the network (less reciprocity)

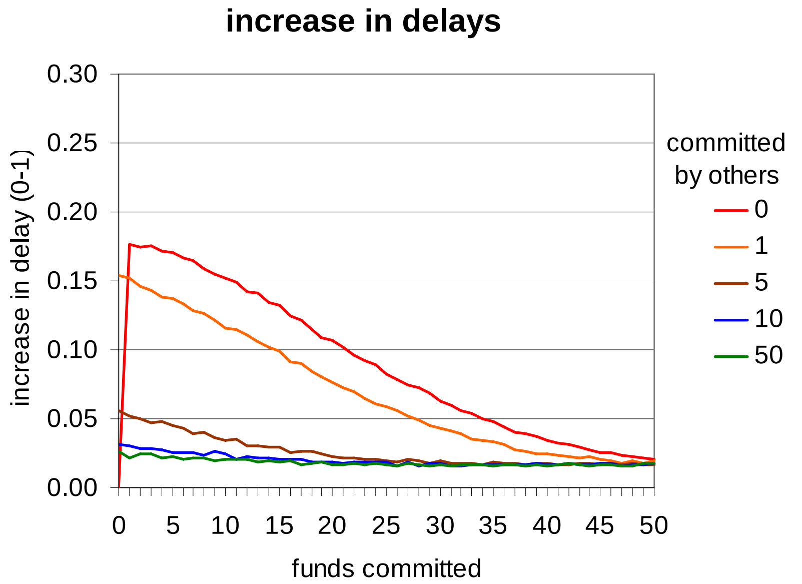




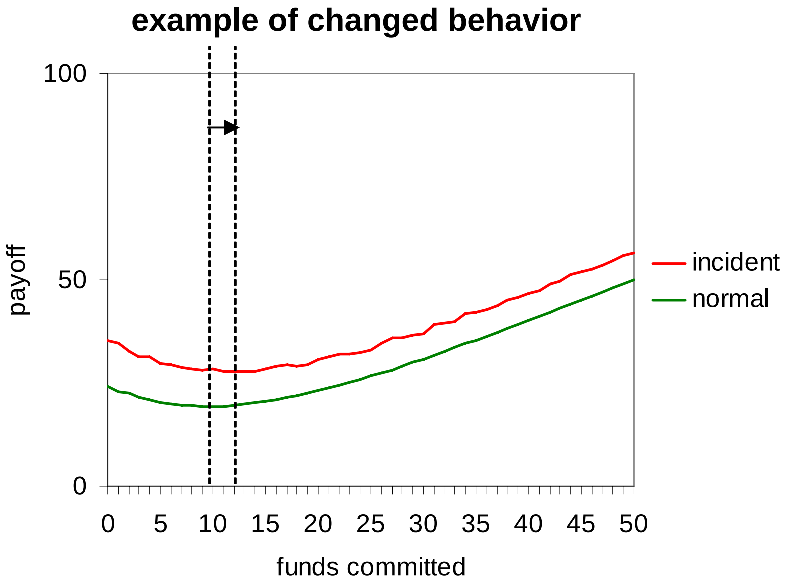
red: n=2
orange: n=5
blue: n=15
green: n=50

Result 3a - operational incident

- One bank can receive, but cannot send for first half of the day (liquidity sink)
- Delays of “non-incident” banks are increased
- More so, when liquidity is scarce

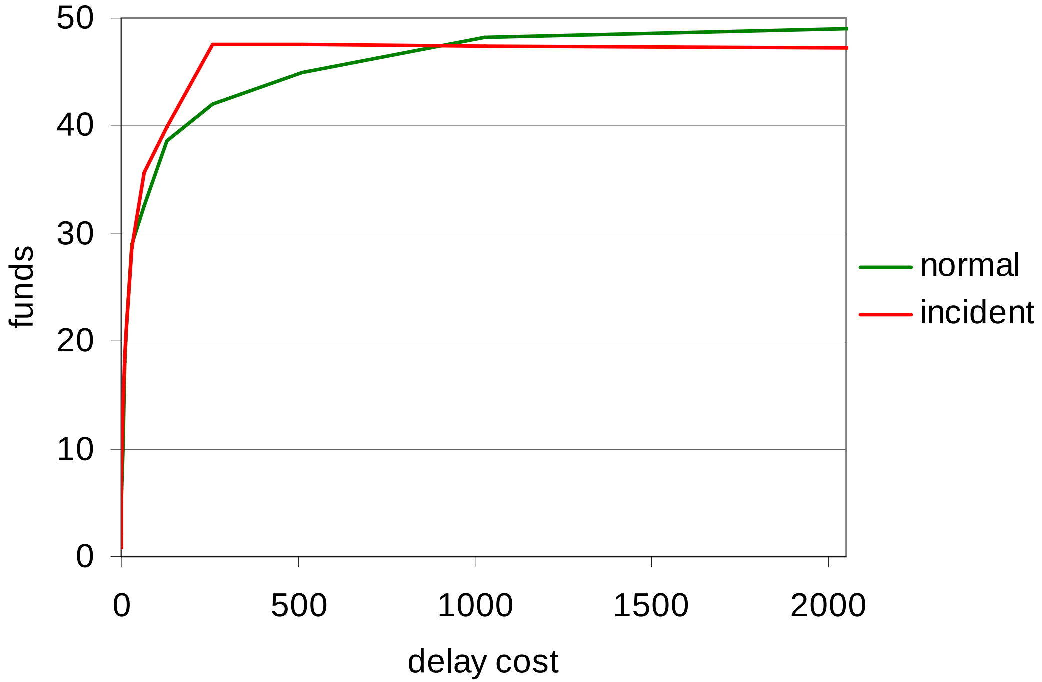


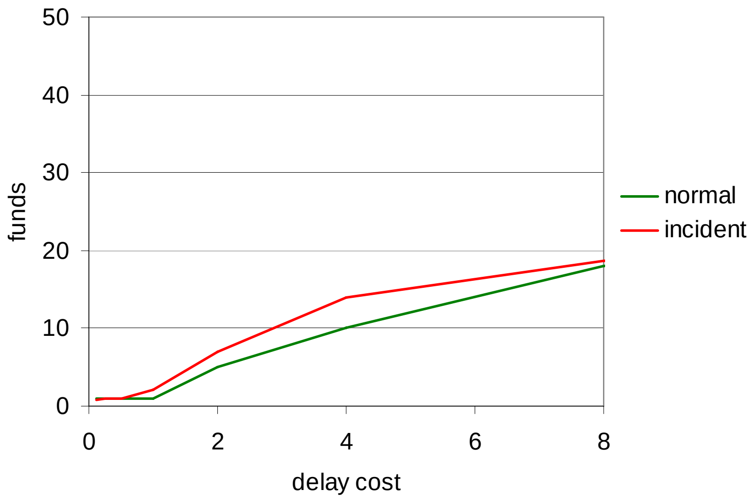
- We expect banks to choose in equilibrium a higher level of liquidity
  - e.g. (with delay cost 4)
  - if others choose 14, in normal circumstances I should choose 10, in case of an incident 14



Results 3b – operational incident

- With low delay cost, only small difference
- As delays get costlier, more liquidity is used
- At extremely high delay cost, adding funds does not





Conclusions

- We developed a model with endogenous decisions by banks on their level of funding
- We investigated the game with more “realistic” costs from settlement than analytical game theoretic models
- The type of game depends on system size and delay cost
- In equilibrium
  - more participants and higher delay costs -> more liquidity
- Operational incident can increase/decrease liquidity holdings
  - payoffs are not improved in equilibrium