BANK OF ENGLAND

Working Paper No. 352

An agent-based model of payment systems

Marco Galbiati and Kimmo Soramäki

August 2008

BANK OF ENGLAND

Working Paper No. 352

An agent-based model of payment systems

Marco Galbiati[1] and Kimmo Soramäki[2]

Abstract

This paper lays out and simulates a multi-agent, multi-period model of an RTGS payment system. At the beginning of the day, banks choose how much costly liquidity to allocate to the settlement process. Then, they use it to execute an exogenous, random stream of payment orders. If a bank's liquidity stock is depleted, payments are queued until new liquidity arrives from other banks, imposing costs on the delaying bank. The paper studies the equilibrium level of liquidity posted in the system, performing some comparative statics and obtaining: i) a liquidity demand curve which links liquidity to delay costs and ii) insights on the efficiency of alternative system configurations.

Key words: Payment systems, liquidity, RTGS, agent-based modelling, learning, fictitious play.

JEL classification: C79.

(1) Bank of England. Email: marco.galbiati@bankofengland.co.uk
(2) Helsinki University of Technology. Email: kimmo@soramaki.net

The views expressed in this paper are those of the authors, and not necessarily those of the Bank of England. We wish to thank the following people for comments: Morten Bech, Walter Beyeler, Ian Bond, Simon Debbage, Charles Kahn, Mark Manning, Stephen Millard, Erlend Nier, Jochen Schanz, Anne Wetherilt and Matthew Willison. Useful comments came from the participants of conferences and seminars: FS seminar at the Bank of England, Central Bank Policy Workshop (Basel), Computer Science Department seminar (University of Liverpool), CCFA Summer school (University of Essex), 5th Simulator Seminar (Bank of Finland), Sixth International ISDG Workshop (Rabat, Morocco), Joint Bank of England - European Central Bank Conference on Payment Systems and Financial Stability (Frankfurt, Germany). The usual disclaimer applies. This paper was finalised on 20 June 2008.

The Bank of England's working paper series is externally refereed.

Information on the Bank's working paper series can be found at
www.bankofengland.co.uk/publications/workingpapers/index.htm

Publications Group, Bank of England, Threadneedle Street, London, EC2R 8AH
Telephone +44 (0)20 7601 4030 Fax +44 (0)20 7601 3298 email mapublications@bankofengland.co.uk

ISSN 1749-9135 (on-line)

Contents

Summary

A large share of all economic transactions is ultimately settled via money transfers between banks, taking place on 'large-value payment systems' (LVPSs). In 2006, the annual value of interbank payments made in the European system TARGET totalled €533 trillion (about $670 trillion), amounting to more than 50 times the value of the corresponding countries' gross domestic products. The sheer size of these transactions, and their importance for the functioning of the economy, explains why policymakers are interested in LVPSs, and in the behaviour of their participants.

In the past, most payment systems worked on a deferred, net settlement basis. During a business day the banks would exchange promises of payments, deferring the actual transfer of funds to the end of the day, when only net positions were settled. The advantage of this arrangement was that only net debtors had to actually provide funds, and only in a quantity sufficient to cover their *net* position. Because net positions are typically small (compared to gross payments), the system as a whole would require little liquidity to function. Today instead, most LVPSs work on a gross settlement basis: there is no netting, and a payment obligation is legally discharged only when the corresponding full amount is transferred across accounts held at a central bank. This apparent backward step, strongly encouraged by monetary authorities worldwide, was motivated by credit risk concerns. Suppose indeed that, in a net system, at the end of the day a bank is unable to make good its final position. Its creditors may face losses too large to be sustained, so their payments too might have to be cancelled, creating a domino effect with significant consequences for financial stability. Gross settlement eliminates this risk but requires more liquidity, as the benefits (not only the risks) of netting are foregone. These arguments suggest that the provision of liquidity is an essential issue to modern payment systems.

Real-time gross systems are more 'liquidity hungry' than deferred net systems. However, they allow for liquidity 'recycling': when a bank receives a payment, it can use the received funds to make other payments of its own. To make an analogy, in a football game the ball can be passed between the players many times; similarly, a same unit of liquidity can be used to settle many payments. Consider however what happens if the ball is expensive to buy – maybe no one would like to pay for it in the first place. Unfortunately the analogy carries on to payment systems, where liquidity (the ball) bears a cost for commercial banks. This is an interest cost (typically

charged by the central bank) or an opportunity cost (when liquidity is obtained against a pledge of collateral). So, even though just a little liquidity could generate a large volume of exchanges, it is unclear who should provide it. Banks are thus faced with a dilemma: to act as liquidity providers by acquiring costly funds, or to wait for liquidity to arrive from other banks. In the first case a bank does not depend on its partners, and it can promptly execute payments. In the second case, a bank benefits from a free source of liquidity, but is exposed to the risk of delaying payments while waiting for funds to arrive.

This paper develops a dynamic model of liquidity provision in a payment system, where banks face a choice between: a) the costs of borrowing liquidity, and b) the cost of delaying payments. In more detail, the model is a sequence of days. At the beginning of each day, every bank chooses how much liquidity to borrow from external sources. This liquidity is then used to execute payment orders which arrive throughout the day in a random, exogenous fashion (these orders can be interpreted as being commissioned by a bank's external clients, or by some area of the bank, different from the treasury). As long as the bank has sufficient funds, payments are executed as soon as they are received; when instead a bank's liquidity balance reaches zero, payments are queued until incoming payments provide the bank with new funds. Finally, at the end of the day banks receive profits, which depend on the liquidity borrowed, and on the delays suffered in executing payment orders. Day after day, banks adapt their liquidity choices following a particular learning process. As a consequence, the banks' behaviour eventually stabilises, and the banks end up providing an equilibrium amount of liquidity.

The system's equilibrium level depends on the model's parameters. By changing these, we look at the amount of liquidity absorbed by the system in a variety of scenarios, drawing conclusions on the efficiency of the system. We find that, for a wide range of costs, efficiency could be enhanced if banks were to commit more liquidity than they do in equilibrium. This might constitute a rationale for imposing measures that encourage liquidity provision (for example, throughput guidelines). From a different perspective, systems with fewer participants are found to be more liquidity-efficient than larger ones, due to the emergence of 'liquidity pooling' effects, as described by previous studies. These results are found by varying the *size* of the system but not its *structure*: it is outside the scope of this work to look at how liquidity choices are affected by changes in the extent of 'tiering' of a payment system (that is, we do not fully investigate the case of banks 'moving out' of the system, and making their payments through other system participants).

1 Introduction

Virtually all economic activity is facilitated by transfers of claims by financial institutions. In turn, these claim transfers generate payments between banks whenever they are not settled across the books of a (perhaps third) institution. These payments are settled in interbank payment systems. In 2006, the annual value of interbank payments made in the European system TARGET totalled €533 trillion (about $670 trillion). In the corresponding US system Fedwire, the amount was $572 trillion, while the UK system CHAPS processed transactions for a value of £59 trillion (about $109 trillion). In perspective, these transfers amounted to 24 to 40 times the value of the respective countries' GDPs. The sheer size of the transfers, and their pivotal role in the functioning of financial markets and the implementation of monetary policy, make payment systems a central issue for policymakers and regulators.

At present, most interbank payment systems work on a real-time gross settlement (RTGS) modality. That is, settlement takes place as soon as a payment is submitted into the system (real time); also, a payment can be submitted only if the paying bank has enough funds to deliver the full amount in central bank money (gross settlement). Because no netting takes place, RTGS modality imposes high liquidity demands on the banks, making RTGS systems vulnerable to *liquidity risk*, ie to the risk that liquidity-short banks are unable to send their own payments. This may create delays and possibly cause gridlocks in the system (see eg Bech and Soramäki (2002)). Hence, liquidity is one of the central issues in RTGS payment systems; as such it attracts the attention of central banks and stimulates a large amount of research. This paper aims at contributing to this knowledge, offering a model of liquidity demand and circulation in an RTGS system. To our knowledge, this is the first paper that explores this question using an 'agent-based' approach, ie combining elements of game theory and numerical simulations.

The amount and the distribution of liquidity in a payment system is the result of a complex interaction between the system's participants. Indeed, during the day, each bank has to make a stream of payments, that can only be partly predicted. To cover the liquidity needs generated by these payments, banks typically rely on two sources: a) reserve balances or credit acquired from the central bank and b) funds received from other settlement banks during the course of the day. The first source can be seen as providing *external* (to the system) *liquidity*, while the second is a source of *internal liquidity*. In normal conditions a bank can draw freely on external liquidity.

This however has a cost, which gives incentives to economise on its use.[1] Internal liquidity on the other hand carries no cost, but its arrival is out of the bank's control. Hence, reliance on internal liquidity exposes the bank to the risk of having to delay its own payment activity – something which is also costly.[2] As a consequence, a bank has to optimally decide how much external liquidity to acquire, trying to forecast when and how much internal liquidity it will receive, trading off external liquidity costs against (expected) delay costs. The fact that banks i) delay some payments, and yet ii) do not wait till the very end of the day to make all their payments, shows that this trade-off indeed exists.

Two main difficulties emerge when studying the behaviour of banks in a payment system. First, when modelled in sufficient detail, liquidity flows in RTGS systems follow complex dynamics, making the bank's liquidity management problem anything but trivial. Indeed, recent work by Beyeler *et al* (2007) shows that, when the level of external liquidity is low, payments lose correlation with the arrival of payment orders; as a consequence, it is difficult to gauge the precise relationship between liquidity and delays, making it hard to determine the optimal usage of external funds. Second, the actions of each bank produce spillover effects on the rest of the system, so no system participant can solve its optimal liquidity demand problem in isolation. As strategic interactions are widespread, banks interact in a fully fledged 'game', jointly determining the performance of the system.

This paper studies this liquidity game, putting particular effort into modelling liquidity flows. We thus build a payments model where external liquidity is continuously 'recycled' among many banks, with delays and costs generated in a non-trivial way by a realistic settlement process. Such realism will inevitably force us to abandon the analytical approach and instead to use simulations. In particular, we use numerical methods to compute a crucial element of the game, the pay-off function, or a relationship between i) a bank's own external liquidity, ii) the external liquidity of other banks, and iii) the resulting settlement delays and costs.

We are interested in the equilibria of the liquidity game, or the choices that banks may be seen to adopt in a consistent fashion. To do so we solve the model adopting a dynamic approach. That is, we assume that banks change their actions over time, using an adaptive process whereby actions

[1]The costs of acquiring liquidity are opportunity costs (returns that the bank would obtain if it could employ this liquidity differently), and interest costs (costs from borrowing the liquidity itself).

[2]Delays usually carry two types of cost. First, formal agreements often penalise late delivery; if a delay extends over the end of the due day, penalties may apply. Second, delays may entail reputational costs, which are difficult to quantify but potentially large.

are chosen on the basis of past experience. We then simulate the resulting dynamics and we look at the limit, or equilibrium, behaviour. This depends on the specific form of the adaptive rule, so we choose the learning process in such a way that, on the one hand, it embeds some rationality on the part of the banks; on the other, it leads to a meaningful equilibrium. A convergence point of our dynamics will be a Nash equilibrium of the liquidity game.

Given its game-theoretic approach, this paper is related to recent work by Angelini (1998), Bech and Garratt (2003, 2006), Buckle and Campbell (2003) and Willison (2005). These papers model various 'liquidity management games' with a few agents and a small number of periods (respectively, two and three). While these models improve our understanding of the incentives in payment systems, the actual pay-off functions may be too simple to describe costs in real payment systems accurately. As we said, in RTGS systems liquidity can circulate many times and between many banks, generating dynamics that cannot be captured by these simple, but analytically tractable, models.

Recently, a growing literature has used simulation techniques to investigate efficiency and risk issues payment systems (see eg James and Willison (2004) and the volumes edited by Leinonen (2005, 2007)). Simulation studies have been widely used in comparing alternative central bank policies, or testing the impact of new system features before their implementation in payment systems. A common shortcoming of such studies has been, however, that participant behaviour is rarely endogenised in the models. The behaviour of banks has either been assumed to remain unchanged across alternative scenarios, or to change in a predetermined manner, leaving aside (or largely simplifying) the strategic aspects studied by the game-theoretic studies.

Recognising the strengths and disadvantages of these two approaches, the present paper tries to build a bridge between them, combining the strength of each of them. Of course, we have to leave something behind: the realism of historical data (which may however be inappropriate to study counterfactual scenarios), and the sharpness of analytical results.

The paper is organised as follows: Section 2 provides a formal description of the model, describes some properties of the cost function, and illustrates the *tatônnement* process towards equilibrium. Section 3 presents the results of the experiments and Section 4 concludes.

2 Description of the model

The model is a stylised representation of a day in RTGS, where the banks (players) engage in the following game.

2.1 Banks and liquidity choices

At the beginning of the day, each of N banks (denoted by $i = 1...N$) chooses its reserves, say $l_i(0)$, to be used in the course of the settlement day.[3] To simplify, we assume that these reserves, the external liquidity, can only be acquired once, at the beginning of the day. Once reserves have been (simultaneously) chosen, the settlement day begins: banks start receiving payment orders, and execute them using available liquidity. In game-theoretic terms, $l_i(0)$ is bank i's *action* and the vector $l = (l_1(0), l_2(0) ..l_N(0))$ is an action profile. The next subsection illustrates how payments are received and executed, generating the outcome of the game.

2.2 Payments and delays

The outcome of the day-game is determined as follows. The day is modelled as a continuous time interval $[0, T]$. Payment orders arrive according to a Poisson process with parameter $\lambda = 1$, so the system as a whole receives, on average, T orders per day. The payor and the payee of these payment orders are determined by (uniform) random draws: for any order, the probability that banks i and $j \neq i$ are respectively the payor and the payee is $\frac{1}{N}\frac{1}{N-1}$. Equivalently, each single bank receives payment orders according to a Poisson process with parameter $\lambda = 1/N$, and the payee of each such order is determined by a random (uniform) draw. These orders can be seen as generated *outside* the bank, by a bank's clients, or *within* the bank, by some area which is different from the treasury department. Whatever the interpretation, payment orders are exogenous for the agent choosing $l_i(0)$.

Let us call $z_i(t)$ the number of payment orders *received* by bank i up to time t, and $x_i(t)$ the number of payment orders *executed* by i up to t. At t, bank i's queue (its backlog of outstanding orders) is therefore

$$q_i(t) = z_i(t) - x_i(t)$$

[3] In the simulations, we assume that $l_i(0)$ is an integer between 0 and some large L.

where we set $z_i(0) = x_i(0) = 0$. Payments orders are executed using available liquidity. Bank i's available liquidity at time t is defined as:

$$l_i(t) = l_i(0) - x_i(t) + y_i(t)$$

where $y_i(t)$ is the amount of payments that i has received from other banks up to time t. For simplicity, we assume that every i adopts the following payment rule:[4]

$$\begin{array}{c} \text{if } l_i(t) > 0, \text{ execute new and queued payments as FIFO;} \\ \text{if } l_i(t) = 0, \text{ queue new payment orders.} \end{array} \quad \textbf{(1)}$$

Bank i's incoming payments $y_i(t)$ are just other banks' outgoing payments, so the settlement process is fully described by the above equations.

As mentioned in the introduction, even this simple model generates extremely complex dynamics of liquidity $l_i(t)$ and queues $q_i(t)$.[5] However, the model can be simulated numerically. A given action profile $l = (l_1(0), ...l_N(0))$ pins down the initial conditions of the system; from there, the exogenous arrival of payment orders mechanically generates liquidity fluxes, queues and delays. All this can be numerically simulated, to determine how delays depend on liquidity choices. For example, Chart 1 shows the (average) amount of delays obtained for different levels of total liquidity in the system, when $l_i(0)$ is the same for each i.[6]

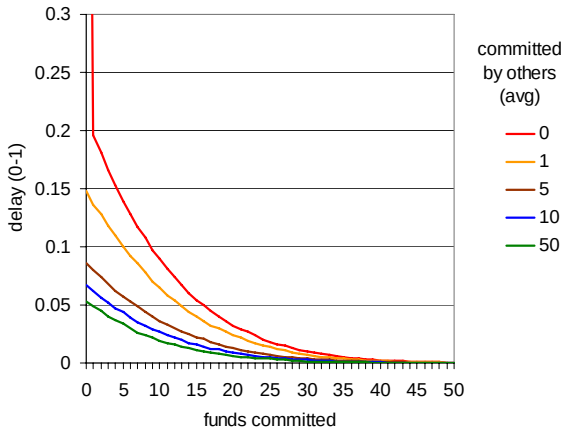
As system liquidity is reduced, delays increase non-linearly due to what are often referred to as 'deadweight losses' (Angelini (1998)) or 'gridlocks' (Bech and Soramäki (2002)). Intuitively, a bank that reduces its liquidity holdings might have to delay its outgoing payments; as a consequence, the receivers of the delayed payments may in turn need to delay their own payments, causing further downstream delays and so on. These delay chains are more likely and more extended the lower the liquidity in the system. Thus, the total effect of liquidity reduction acts in a compounded fashion.

[4]Such a rule is optimal for the cost specification given in the next section: banks need to pay upfront for liquidity, so they have no incentive to delay payments if liquidity is available. Under other cost specifications (eg heterogeneous payment delay costs) this would, however, not be the case.

[5]Queues do not form only when $l_i(0)$ is very high. Then, $\Delta x = \Delta z$ so executed payments essentially follow a Poisson process which mirrors the arrival of payment orders.

[6]Delays are normalised such that 1 reflects a situation where all payments are delayed until the end of the day, and 0 a situation where no delays take place.

Chart 1: Delays as a function of total liquidity



2.3 Costs

At the end of the settlement day, banks receive pay-offs that depend on the liquidity posted at the beginning of the day, and on the delays generated by the settlement algorithm illustrated in the above section. More precisely, we assume that acquiring initial liquidity l_i (0) imposes a liquidity cost equal to:

$$C\left(l_i\left(0\right)\right) = \lambda l_i\left(0\right), \qquad \lambda > 0 \tag{2}$$

This is the first component of a bank's pay-off (cost). We then suppose that a payment order received at t and executed at t' carries a penalty equal to

$$c\left(t', t\right) = \kappa(t' - t), \qquad \kappa > 0 \tag{3}$$

Such penalties are summed over all received payment orders of the day, to give a bank's delay cost. A bank's total pay-off is then the sum of delay and liquidity costs.

The random arrival of payment orders generates random delays; hence, pay-offs too are a random function of the action profile l (0). As anticipated above, the analytical form of this pay-off is exceedingly complex to determine; hence, we simulate the settlement process many times for every action profile, to obtain a numerical estimate of expected costs, as a function of l (0).[7] The resulting pay-off function is plotted in Charts 2 and 3 for two levels of delays costs; 'low' (2) and 'high' (3).

The simulations also show an interesting fact:

[7]We assume banks are risk-neutral, ie they care about expected pay-offs.

Chart 2: Costs as a function of own initial funds - low delay costs

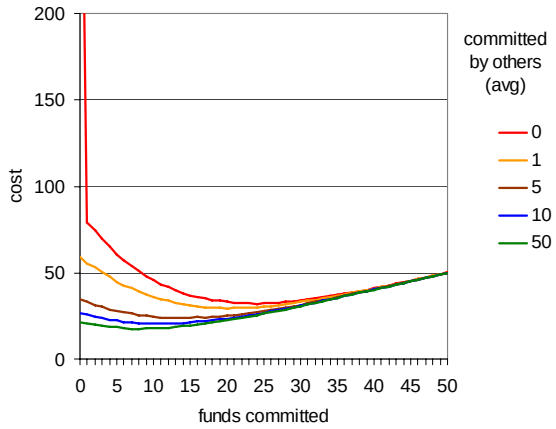
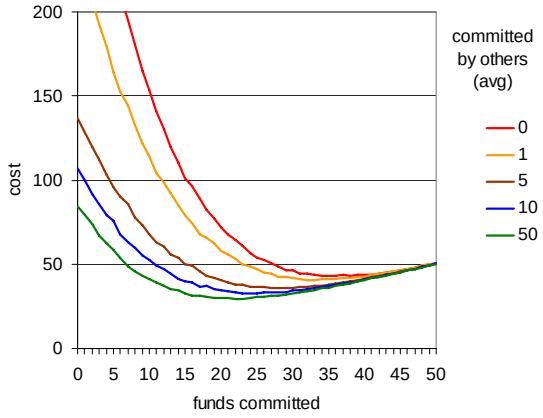


Chart 3: Costs as a function of own initial funds - high delay costs



Remark 1 Bank i's cost is (essentially) a function of its own action and of the sum of others' actions.

This important empirical finding, which probably depends on the fact that the day is 'long', greatly simplifies the analysis.[8] First, in certain respects it leaves us with a game with only two players: bank i playing against 'the rest of the system'. Second, it allows us to derive some analytical results, to be discussed in the next section.[9]

We find this result by comparing two sets of simulations. In the first set, we vary the total amount of liquidity, while spreading it uniformly across banks (ie we simulate the settlement day for different values of $\sum l_i(0)$, imposing every time $l_i(0) = \frac{1}{N}\sum l_i(0)\ \forall i$). In the other, we change again the total liquidity, but we distribute it randomly across banks, so $l_i(0)$ varies across banks. Comparing the total costs in the two sets, we found that the differences are small – around 2% or less. We suspect this can be explained by two facts: i) the assumption of a complete symmetric network (every bank exchanges payments to any other with similar intensity), and ii) the relatively large number of payments quickly redistributes liquidity, flushing out the initial conditions. Both assumptions are realistic in many systems, for example in the UK CHAPS system (see Soramäki *et al* (2007)).

2.4 Equilibrium

To find the equilibrium of the liquidity game, we use the so-called *fictitious play* tatônnement process (Brown (1951)). Largely studied in evolutionary game theory, fictitious play is a specification of how players change their actions in time, learning from experience. A precise description of this process is in the appendix; the reason to adopt this particular dynamic is twofold. First, despite its simplicity the fictitious play rule is in a sense rational and thus not too unrealistic, corresponding to Bayesian updating of beliefs about others' actions.[10] Second, fictitious play can indeed be a useful tool to compute equilibria. Indeed, when fictitious play converges to a stable action profile, this is a Nash equilibrium of the underlying game.[11]

[8] We do not have a rigorous proof, but we suspect the following. When many payments are made (ie the day is 'long'), liquidity is soon spread among banks according to a stable distribution. Hence the initial distribution does not matter, only the total liquidity does.

[9] Games with this property are known as *aggregation games*. They have the convenient feature that a number of adjustment dynamics applied to them are 'well behaved' (see eg Mezzetti and Dindo (2006)).

[10] See eg Fudenberg and Levine (1998, page 31) for details.

[11] It is well known that fictitious play may fail to converge. However this is not the case here, as shown by the simulations. Interestingly, convergence in aggregation games was shown by Kukushkin (2004) for a dynamic similar to fictitious play.

Summing up, fictitious play can be seen either as a computational device, or as a 'story' with an appealing economic meaning.

A key question is whether the game has a unique equilibrium and, if not, which equilibrium will be uncovered with fictitious play. The appendix discusses this in more detail. The bottom line is that, although our model does have different equilibria (depending on the initial conditions the simulations will pick one or the other), all equilibria are characterised by exactly the same total level of liquidity $\sum_i l_i(0)$, which can therefore be rightly called *the* equilibrium liquidity. This allows us to perform comparative statics, where we change parameters of the cost function and other elements of the model.

3 Results

3.1 Liquidity demand and efficiency of the equilibrium

We start with a base case scenario with 15 banks; this number is chosen so that our system 'looks like' the UK CHAPS.[12] In all of the simulations banks interact in a complete network, ie each sends payments to every other bank in the system – another fairly realistic assumption for CHAPS.

First, we obtain a 'liquidity demand function', relating the (equilibrium) amount of external liquidity $\sum_i l_i(0)$ to unit delay costs κ, for λ normalised to 1.[13] As expected, the amount of liquidity acquired by the banks is low for relatively inexpensive delays (Chart 4). When k grows, so does liquidity demand, roughly following a logarithmic pattern up to a certain point. However, as liquidity grows, delays become increasingly rare. As a consequence, decreasing returns on liquidity eventually prevail, causing liquidity demand to eventually flatten out.

An important question is whether the equilibrium of the liquidity game is efficient; that is, whether the self-interested behaviour of banks can be improved upon by some co-ordinated action. To answer this question, one should ideally compare the equilibrium outcome of the game, to what would result if banks were *jointly* minimising the *total* costs of the system. To

[12]The length of the day is 3,000 'time ticks', so on average each bank makes 200 payments a day.

[13]Only λ/κ matters for the banks' decisions; hence our demand function is essentially equivalent to a 'traditional' liquidity demand, where the demand $\sum_i l_i(0)$ depends on the cost λ.

Chart 4: Equilibrium external liquidity as a function of delay costs

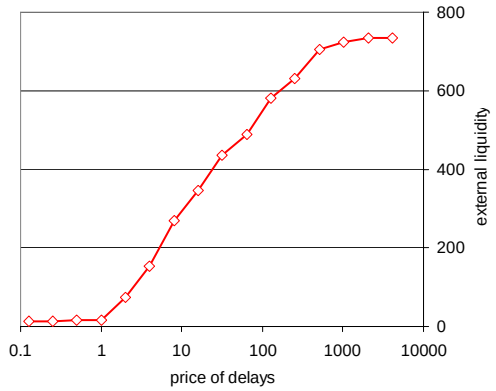
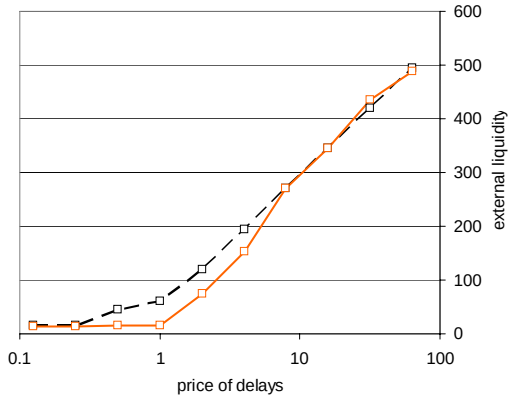


Chart 5: Cost-minimising common action (dashed) versus Nash-equilibrium outcome



simplify computations, we search for optimal liquidity levels under the constraint $l_i(0) = l_j(0)\ \forall i, j$ – all banks are given the same amount of funds.[14] We find that the equilibrium outcome roughly coincides with the collective cost minimising choice for extreme values of k (delay costs), as shown in Chart 5 (the continuous line represents the liquidity minimising total cost). However, for intermediate unit delay costs, the outcome reached by independent banks is dominated by the co-ordination outcome, where more liquidity is provided as a whole.

At the origin of such inefficiency are positive externalities in liquidity provision: external liquidity is used by all banks, but of course an individual institution only cares about private costs

[14]This constraint should not be binding: returns to own liquidity are decreasing, so redistribution from a liquidity-rich to a liquidity-poor bank should on average reduce total delays. Hence, an efficient allocation of liquidity should assign the same $l_i(0)$ to all banks.

and benefits. Competitive banks then free-ride on others, leading to insufficient provision of external liquidity.

3.2 Relative efficiency of different size networks

Is a system with more participants preferable to a smaller one? This question can be considered from different points of view: from a risk / financial stability perspective,[15] or from a cost-efficiency perspective. Here, we concentrate on the second aspect. We then run experiments varying the number of banks in the model. In the first experiment, we increase the number of participants while keeping the number of payments per bank constant. In the second experiment, we increase the number of participants while keeping constant the system-wide number of payments (so per-bank payments are decreased). We measure efficiency using the netting ratio, that is the average amount of *external* liquidity required for each payment:

$$netting\ ratio = \frac{total\ external\ liquidity}{total\ payments}$$

The lower the netting ratio, the higher is the level of 'liquidity recycling' in the system.

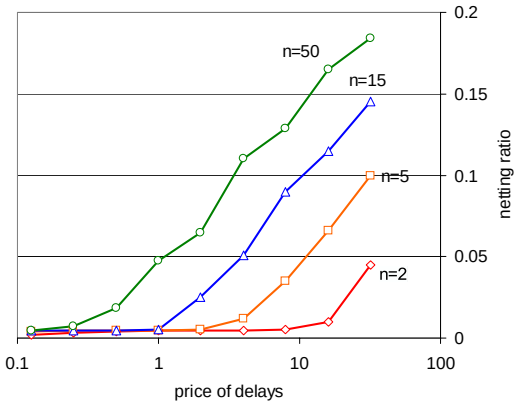
A caveat: while we change the size of the system, we maintain the assumption of a complete and symmetric payment network. This type of change (a pure 'rescaling') is convenient to analyse, but is just a simplified description of what happens in real payment systems. There, changes in the number of banks are usually accompanied by changes in the topology of the system, as some banks *de facto* merge their payment activity with others (giving rise to the so-called 'tiering'). When this happens, liquidity demand is influenced in a complex way by a number of factors, that we do not need to consider in our simplified 'rescaling' case. The interaction of liquidity demand (and costs) and tiering is outside the scope of this paper; it is instead studied in Jackson and Manning (2007).

3.2.1 Size effects I – constant individual bank payments

Here we vary N (the number of banks), while keeping the number of payments per bank constant – so the number of system-wide payments changes accordingly. The number of system participants has a dramatic effect on liquidity choices and efficiency. As the system size increases, liquidity demand grows while efficiency falls, and increasingly so as delays become

[15] For example, fewer participants could imply that the failure of one bank implies disruption of a larger share of payments. On the other hand, fewer participants might also mean safer participants, making it non-trivial to draw financial stability conclusions.

Chart 6: Equilibrium external liquidity with alternative system sizes and fixed turnover per bank



expensive. As Chart 6 shows, for low delay costs the netting ratio[16] is virtually unaffected by the network size. But, at higher unit delay costs, differences are amplified and systems with fewer participants are more liquidity-efficient than systems with a higher number of participants.

The following is an intuitive explanation of this result (as we said, liquidity flows are too difficult to be described analytically, so we can only rely on intuition to interpret the simulations). Consider a bank i and suppose N is increased from, say, 2 to 3. Because the number of payments per bank are kept constant and equally distributed over all banks, both outgoing and incoming expected payments remain constant for i in any time interval.[17] However, the *variance* of i's incoming payments increases: at each t, a bank i can now receive 0, 1 or 2 payments instead of only 0 or 1. Faced with a more unstable source of internal liquidity, the banks find it convenient to rely more on external liquidity.

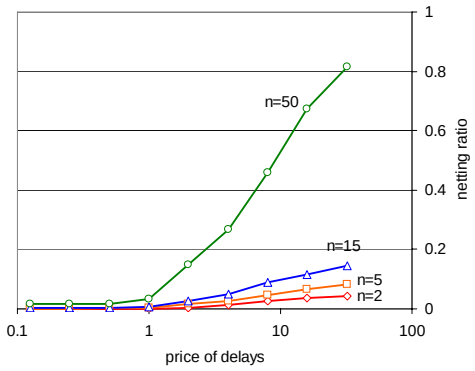
3.2.2 Size effects II – constant total volume

In a second experiment, we keep constant the system total volume, distributing it over a varying number of banks. Note however that we keep constant the number of payments *between* banks. The results, illustrated in Chart 7, show a pattern similar to the previous case: systems with fewer members are seen to absorb less liquidity.

[16]The average liquidity required for each payment, or the ratio (total bank external liquidity) / (total payments).

[17]If Z is the number of a bank's *out*going payments, the total outflow out of all $j \neq i$ is $(N-1)Z$. By construction, i captures a fraction $1/(N-1)$ of this flow, ie Z, which is kept constant.

Chart 7: Equilibrium external liquidity with alternative system sizes and fixed total turnover



Our (again, intuitive) explanation of this finding is as follows. A reduction of the number of banks from N to say N' can be seen as taking place in two 'steps': first, a reassignment of the payments as to involve N' banks only; second the elimination of the banks left with no payments. The second stage is neutral, as the eliminated banks are 'dummy'. Instead, the first stage brings about liquidity savings, due to the so-called liquidity 'pooling effect' (see eg Jackson and Manning (2007)). In turn, the liquidity pooling can be explained as follows: suppose the payments to/from two different banks are settled by one bank only. The volatility of the liquidity balance of this one bank increases, but by a factor less than two. Thus, a liquidity buffer of less than two times the original liquidity buffers is sufficient to settle all payments; a more precise explanation is given in the appendix.

4 Conclusions

In this paper we build and simulate an agent-based model of an RTGS system, paying special attention to the complex liquidity flows exchanged by the participating banks. The simulations demonstrate that a complete, symmetric RTGS system can be described as an aggregation game, whose convenient features allow us to compute the equilibrium behaviour of the system, and to perform various comparative statics exercises.

First, we retrieve a liquidity demand function, relating the system's liquidity to the costs faced by banks in their payment activity (liquidity versus delay costs). Then we consider the question of whether such liquidity demand, expressed by non-cooperating banks, is efficient. We find that, for a wide range of costs, efficiency (measured by the netting ratio) could be *enhanced* if banks

were to commit *more* liquidity than they do in equilibrium. This might constitute a rationale for imposing measures that encourage liquidity provision (for example, throughput guidelines). From a different perspective, systems with fewer participants are found to be more liquidity-efficient than larger ones, due to the emergence of 'liquidity pooling' effects, as described by previous studies. We privileged complexity and realism, over analytical solvability. Consequently, we used a numerical, agent-based approach. Besides being useful when closed-form results are difficult to obtain, our approach is flexible and modular, allowing the present work to be extended to alternative scenarios. Further research may look at different network structures, at more elaborated liquidity management rules, at banks that differ in their costs or payment orders. Finally, our model of a 'vanilla' RTGS system could be easily extended to 'hybrid' systems like the European TARGET, which features liquidity-saving mechanisms.

5 Appendix

Fictitious play

Consider a sequence of daily games (settlement days) running from $t = 0$ to potentially infinity. The actions chosen on day t are a vector $l^t = \left\{l_1^t, l_2^t, ..., l_N^t\right\}$.[18] Fictitious play assumes that, over the sequence of days, every i forms a belief of what others will play next, choosing l_i^t as a best reply to such belief:

- i's belief at time t is a vector $p_i^t(.) = \left(p_i^t(1), p_i^t(2) ...\right)$, where $p_i^t(x)$ is the probability that i attaches to $\sum_{j \neq i} l_j^t = x$ being played at t.
- a bank updates its belief according to the following rule:
$$p_i^t(k) = \frac{1 + \sum_{s=1...t-1} I_k(s)}{t + \Lambda}$$
where $\Lambda = NL$ (N being the number of banks, L the maximum liquidity each can post), and $I_k(s)$ is defined to be 1 if $\sum_{j \neq i} l_j^i = k$, and zero otherwise.[19]
- at t, bank i chooses $l_i^t = \arg\max_l \sum_{x=1}^{L} f_i(l, x)\, p_i^t(x)$ − where $f_i(l, x)$ is the cost incurred by i playing l, if the others play $\sum_{j \neq i} l_j^t = x$.

Equilibria in the simulations

Most of the equilibria found with the simulations have banks switching between two or more actions, depending on the evolution of their beliefs. This is due to the fact that, in the simulations, liquidity choices are discrete. For example, at the lowest delay price level banks oscillate between $l = 0$ and $l = 1$, chosen with probabilities 8.6% and 91.4%, respectively. As banks become sufficiently confident that other banks chose $l = 1$ each, the best reply is $l = 0$. As the probability of others choosing $l = 0$ is thereby increased, banks switch back to $l = 1$. In this case, the game is a classic 'hawk-dove' game. If no one commits any liquidity, all will experience very high delays as no payments can be settled. If everyone commits one unit of liquidity, payment settlement can take place. From an individual bank's perspective, however, a

[18] Here l_i^t denotes the action $l_i(0)$ chosen at time zero in day t. We are not interested in the intraday timing now, but rather in sequence of days, so we slightly change notation.

[19] On the first day ($t = 0$), all banks believe that each $\sum_{j \neq i} l_j^i$ is equally likely: $p_i^0(k) = 1/\Lambda$. Then, the more frequently a $\sum_{j \neq i} l_j^i$ is played, the more frequently it is 'believed' to be played again.

better outcome would be not to commit any liquidity while others do. As the cost for delays is increased, the probability of banks committing no liquidity is reduced gradually until, at delay price of one, a pure equilibrium emerges, where each bank chooses $l = 1$. At higher cost levels banks either reach a pure equilibrium, or a mixed equilibrium where they mix between a narrow range of different liquidity levels.

Uniqueness of equilibrium liquidity level

We now show that all equilibria feature the same level of aggregate liquidity. This allows us to speak about *the* equilibrium liquidity, even though the game may possess many different equilibria.

Recall that

- $f(l_i, l_{-i})$ is the expected pay-off (cost) of bank i at strategy profile (l_i, l_{-i}).

By Remark 1 (page 12), we can also consider $f(l_i, l_{-i})$ a function of two variables. So, from now on l_{-i} is no longer a vector but a scalar, $l_{-i} = \sum_{j \neq i} l_j$. We need some new notation:

- $l_i^*(l_{-i})$ is bank i's best reply to l_{-i}.
- $\Delta_i = l_i' - l_i$, to be used when l_{-i}' and l_{-i} are clear from the context. Similarly, $\Delta_i^* = l_i^*\left(l_{-i}'\right) - l_i^*(l_{-i})$, and $\Delta_{-i} = \sum_{j \neq i}(l_j' - l_j)$ and $\Delta = \sum_{i \in N}(l_i' - l_i)$.
- $z(l_i, l_{-i})$ is the amount of delays suffered by i at strategy profile (l_i, l_{-i}), so total pay-offs are $f = \lambda l_i + \kappa z(l_i, l_{-i})$.

We can now prove our result:

Theorem 1 All equilibria feature the same total liquidity.

Proof. The argument proceeds in two steps.

Step (1) For each l_{-i} and l'_{-i}, *we have* $\Delta_i^* \leq -\Delta_{-i}$.

That is, a bank optimally 'under-reacts' to a change in others' liquidity. To show this, first note that when we take second derivatives of $f = \lambda l_i + \kappa z\left(l_i, l_{-i}\right)$ only the second term survives, so eg $\frac{\partial^2 f(l_i,l_{-i})}{\partial l_i \partial l_{-i}} = \kappa \frac{\partial^2 z(l_i,l_{-i})}{\partial l_i \partial l_{-i}}$.[20] The diagram on page 22, shows how the liquidity balance of a bank may evolve in time (kinked line). Delays z are measured by the area below the zero liquidity line (balances cannot become negative, so the 'depth' below the zero line represents the length of a queue). From the picture it is evident that $\frac{\partial^2 z(l_i,l_{-i})}{\partial l_i^2} < 0$, as also found in the simulations (Chart 1). Hence, l_i^* satisfies the first-order condition $\frac{\partial z(l_i,l_{-i})}{\partial l_i} = g\left(l_i, l_{-i}\right) = \lambda$, and the standard result applies: $\frac{dl_i^*}{dl_{-i}} = -\frac{\partial g(.)}{\partial l_{-i}} / \frac{\partial g(.)}{\partial l_i}$. Close examination of the diagram also reveals that $\frac{\partial^2 z(l_i,l_{-i})}{\partial l_i \partial l_{-i}} \leq \frac{\partial^2 z(l_i,l_{-i})}{\partial l_i^2}$, so $\frac{dl_i^*}{dl_{-i}} \leq -1$, which is the statement of Step (1).[21]

Step 2) If l and l' are *equilibria, then* $\Delta l = \sum l'_i - \sum l_i = 0$.

To reach a contradiction, suppose l and l' are equilibria but $\sum l'_i > \sum l_i$ ie $\Delta > 0$. If it were so, there should be a non-empty set of banks $S : \Delta_k > 0$ for all $k \in S$. By Step 1) we can write $\Delta_k = -\left(\Delta_{-k} + \varepsilon_k\right)$ (with $\varepsilon_k \leq 0$), so the total change in liquidity between the two equilibria is:

$$\Delta = \left[\sum_{k \in S} \Delta_k\right] + \left[\sum_{k \in N \setminus S} \Delta_k\right] = -\left[\sum_{k \in S} \left(\Delta_{-k} + \varepsilon_k\right)\right] + \left[\sum_{k \in N \setminus S} \Delta_k\right]$$

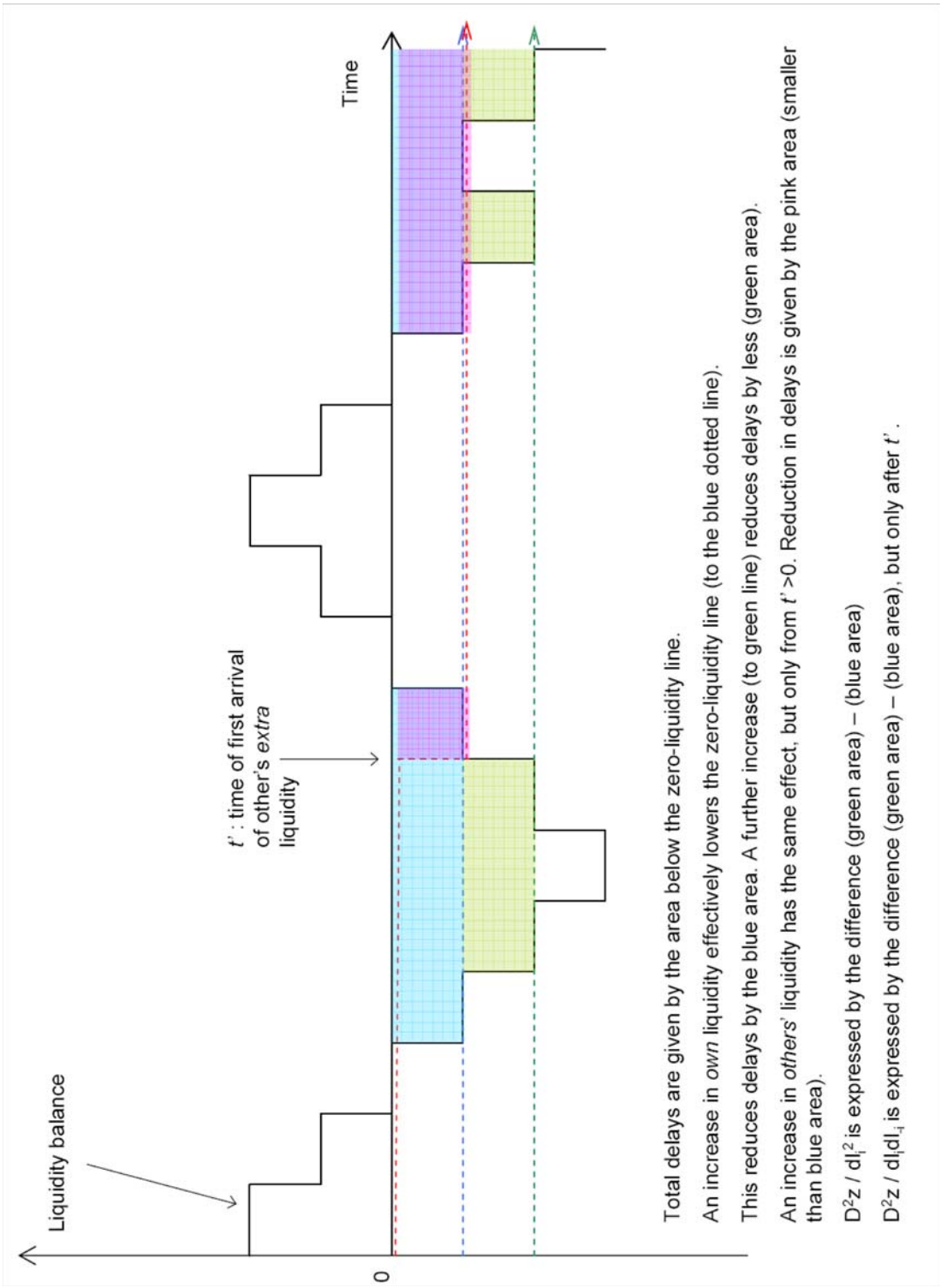
Now, given a set $S = \{x_1, x_2, x_3..\}$ it is clear that $\sum_{i \in S} x_{-i} = (|S| - 1) \sum_{i \in S} x_i$. Similarly, if x_{-i} comes from a larger set $R \supseteq S$, then $\sum_{i \in S} x_{-i} = (|S| - 1) \sum_{i \in S} x_i + |S| \sum_{i \in R \setminus S} x_i$. So the above expression can be written as

$$\begin{aligned}
\Delta &= -\left[(|S| - 1) \sum_{k \in S} \Delta_k + |S| \sum_{k \in N \setminus S} \Delta_k - \overbrace{\sum_{k \in S} \varepsilon_k}^{\varepsilon}\right] + \left[\sum_{k \in N \setminus S} \Delta_k\right] \\
&= -\left[|S| \left(\sum_{k \in S} \Delta_k + \sum_{k \in N \setminus S} \Delta_k\right) - \sum_{k \in S} \Delta_k - \varepsilon\right] + \left[\sum_{k \in N \setminus S} \Delta_k\right] \\
&= (1 - |S|) \Delta l + \varepsilon \\
&\Rightarrow \Delta = \frac{\varepsilon}{|S|}
\end{aligned}$$

But $\varepsilon \leq 0$, so this contradicts $\Delta > 0$.

[20] Strictly speaking, we should not be using derivatives, as payments and liquidity choices are discrete. The argument in terms of difference is similar, just more cumbersome.

[21] Because $\frac{\partial^2 z(l_i,l_{-i})}{\partial l_i \partial l_{-i}} < \frac{\partial^2 z(l_i,l_{-i})}{\partial l_i^2}$ everywhere, this inequality extends to non-infinitesimal changes in l_{-i}.


Liquidity balance
Time
0
t' : time of first arrival of other's extra liquidity
Total delays are given by the area below the zero-liquidity line.
An increase in own liquidity effectively lowers the zero-liquidity line (to the blue dotted line).
This reduces delays by the blue area. A further increase (to green line) reduces delays by less (green area).
An increase in others' liquidity has the same effect, but only from t' >0. Reduction in delays is given by the pink area (smaller than blue area).
D²z / dl_i² is expressed by the difference (green area) – (blue area)
D²z / dl_i dl_-i is expressed by the difference (green area) – (blue area), but only after t' .

System size and pooling effect

In the main text we said that when payments are distributed over more banks, the liquidity needs of the system increase. This is due to the liquidity pooling effect, that we now illustrate for the (simpler) case where liquidity is abundant, so queues do not form.

When liquidity is abundant, a bank's net liquidity balance is a random walk: over a time interval Δt, on average, $p\Delta t$ payments are made (pushing 'down' the liquidity balance), and $p\Delta t$ payments are received (pushing 'up' the balance). Hence, the average balance change is zero, with a standard deviation $\sigma = \sqrt{p\Delta t}$. Suppose the number of participants N is increased to $N' = Nx$ (with $x > 1$), but turnover is kept constant. Payments are now distributed over more banks, so their arrival rate is reduced from p to p/x. As a consequence, the variance in a bank's balance is reduced to $\sigma' = \sqrt{p/x\,\Delta t} > \sigma\sqrt{1/x}$.

Suppose now that a bank's optimal liquidity l_i is proportional to its balance variance (say $l_i = z\sigma$, which is exactly the case if a bank chooses l_i as to cover z standard deviations from the average balance). Then, the fall in variance (factor $\sqrt{1/x}$) is not enough to offset the increase in system's size (factor x), so the larger system absorbs more liquidity: $N'z\sigma' = (Nx)\,z\sigma\sqrt{1/x} > Nz\sigma$.

References

Angelini, P (1998), 'An analysis of competitive externalities in gross settlement systems', *Journal of Banking and Finance*, Vol. 22, pages 1-18.

Bech, M L and Garratt, R (2003), 'The intraday liquidity management game', *Journal of Economic Theory*, Vol. 109 (2), pages 198-219.

Bech, M L and Garratt, R (2006), 'Illiquidity in the interbank payment system following wide-scale disruptions', *Federal Reserve Bank of New York Staff Report no. 239*.

Bech, M L and Soramäki, K (2002), 'Liquidity, gridlocks and bank failures in large value payment systems', *E-money and Payment Systems Review*, Central Banking Publications, London.

Beyeler, W, Bech, M, Glass, R and Soramäki, K (2007), 'Congestion and cascades in payment systems', *Physica A*, Vol. 384, Issue 2, pages 693-718.

Brown, G W (1951), 'Iterative solutions of games by fictitious play', in Koopmans, T C (ed), *Activity analysis of production and allocation*, New York: Wiley.

Buckle, S and Campbell, E (2003), 'Settlement bank behaviour and throughput rules in an RTGS payment system with collateralised intraday credit', *Bank of England Working Paper no. 209*.

Fudenberg, D and Levine, D K (1998), *The theory of learning in games*, MIT Press, Cambridge, Massachusetts.

Jackson, J and Manning, M J (2007), 'Central bank intraday collateral policy and implications for tiering in RTGS payment systems', *DNB Working Paper no. 129*.

James, K and Willison, M (2004), 'Collateral posting decisions in CHAPS Sterling', Bank of England *Financial Stability Review*, December, pages 99-104.

Kukushkin, N S (2004), 'Best response dynamics in finite games with additive aggregation', *Games and Economic Behaviour*, Vol. 48, pages 94-110.

Leinonen, H (2005) (ed), 'Liquidity, risks and speed in payment and settlement systems – a simulation approach', *Bank of Finland Studies*, E: 31.

Leinonen, H (2007) (ed), 'Simulation studies of liquidity needs, risks and efficiency in payment networks', *Bank of Finland Studies*, E: 39.

Mezzetti, C and Dindo, M (2006), 'Better-reply dynamics and global convergence to Nash equilibrium in aggregative games', *Games and Economic Behaviour*, Vol. 54, pages 261-92.

Soramäki, K, Bech, M L, Arnold, J, Glass, R J and Beyeler, W E (2007), 'The topology of interbank payment flows', *Physica A*, Vol. 379, pages 317-33.

Willison, M (2005), 'Real-Time Gross Settlement and hybrid payments systems: a comparison', *Bank of England Working Paper no. 252*.