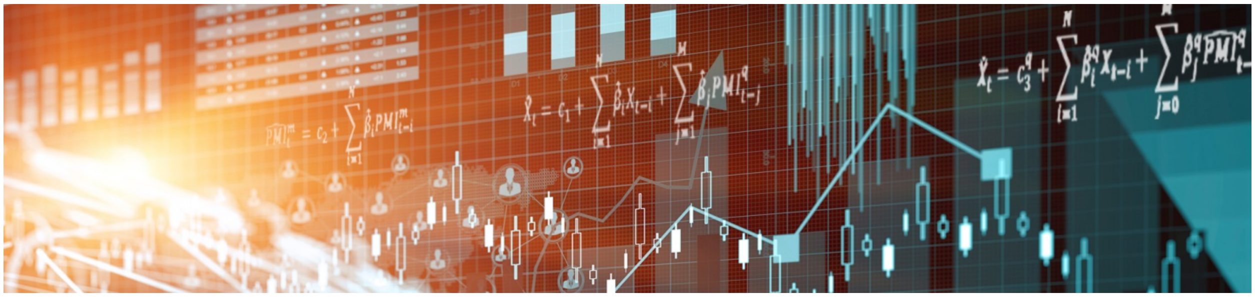
BANK OF CANADA
BANQUE DU CANADA

Staff Working Paper/Document de travail du personnel 2019-26

An Application of Shapley Value Cost Allocation to Liquidity Savings Mechanisms



by Rodney J. Garratt

Bank of Canada staff working papers provide a forum for staff to publish work-in-progress research independently from the Bank's Governing Council. This research may support or challenge prevailing policy orthodoxy. Therefore, the views expressed in this paper are solely those of the authors and may differ from official Bank of Canada views. No responsibility for them should be attributed to the Bank.

www.bank-banque-canada.ca

Bank of Canada Staff Working Paper 2019-26

July 2019

An Application of Shapley Value Cost Allocation to Liquidity Savings Mechanisms

by

Rodney J. Garratt

Department of Economics
University of California Santa Barbara
Santa Barbara, California, USA 91306
garratt@ucsb.edu

ISSN 1701-9397

Acknowledgements

I thank John Jackson, Pedro Gurrola-Perez, Andrejus Pustelnikovas, Francisco Rivadeneyra and seminar participants at the Bank of Canada, the Bank of England and the Bank of Finland for helpful remarks. This paper was written while I was a visiting scholar at the Bank of Canada. The views expressed are my own and do not necessarily reflect the views of the Bank of Canada.

Abstract

This paper describes a proposal for operating a centralized netting queue for non-urgent interbank payments that involves take-it-or-leave-it offers and allocates liquidity costs using the Shapely value. This method achieves fairness and ensures welfare maximizing netting proposals are agreeable to all providers of liquidity.

Bank topics: Payment clearing and settlement systems
JEL codes: C72, E58

1 Introduction

In most countries, the central bank provides or oversees an infrastructure for settling large-value (wholesale) payments between banks. Historically, interbank payments were settled using end-of-day netting systems, but as volumes and values increased, central banks became worried about the risks inherent in deferred net settlement systems, and most opted to implement a Real-Time Gross Settlement (RTGS) system. With RTGS, payments are processed individually, immediately and with finality during operational hours. This eliminates settlement risk and the potential unwinding of payments at the cost of increased need for liquidity to be provided by participants.

Liquidity demands in RTGS systems can be enormous. For example, the Fedwire Funds Service, which is the large-value payment system in the United States, averaged close to $3 trillion a day in 2016, compared with an annual GDP of $18.46 trillion, and TARGET2, the European Union's large-value payment system, averaged €1.7 trillion a day in 2016, compared with an annual GDP of €16.5 trillion.[1] In order to reduce the high liquidity demands in their large-value payments systems, central banks around the world have implemented liquidity savings mechanisms (LSMs).[2] LSMs include policies designed to encourage greater liquidity recycling.[3] However, the most effective LSMs are those that economize on liquidity needs by matching offsetting payments that have been submitted to a central queue and settling these payments using only the liquidity needed to cover the net obligations.

In the ideal scenario, the liquidity provided to the queue by participants whose net obligations are positive is sufficient to cover these obligations, and all payments in the queue

[1] See http://www.federalreserve.gov and http://www.ecb.europa.eu.

[2] See Norman (2010).

[3] Liquidity recycling arises from the fact that banks can use incoming liquidity to make outgoing payments, and, generally, more liquidity will be recycled if banks make payments in a timely fashion. Policies that incentivize timely payment processing (e.g., throughput guidelines) or punish delayers (e.g., time-varying tariffs) make it more likely that liquidity will be recycled and hence reduce the overall liquidity needs of the system.

can be cleared. In practice, however, this is not always the case. Rather, the queue operator often must look for a subset of payments that can be settled using available liquidity. This is a constrained integer programming problem: if there are n payments in the queue, then there are $2^n - 1$ possible combinations of payments to be considered. For large n this problem becomes NP-hard. Consequently, systems that employ queues use simple rules for winnowing down the set of payments, such as first-in, first-out (FIFO) with options for reordering or bypass.[4] It is also common for existing centralized queue algorithms to abandon FIFO altogether and instead seek to maximize value or volume settled. These algorithms are referred to as sorted-queue algorithms because they operate by sorting the payments of individual banks in ascending or descending order based on value and then sequentially removing payments from these lists until liquidity constraints are met.[5] Existing centralized queuing systems do not always achieve the optimum solution in terms of maximizing the value of payments settled given available liquidity. Even when they do, meeting this objective does not necessarily maximize system welfare.

This paper seeks to improve upon existing centralized netting queues by making two fundamental changes. First, instead of making decisions on how much liquidity to provide to the queue before netting arrangements are determined, banks receive take-it-or-leave-it offers that determine which of their payments will be settled as well as their share of the liquidity cost. This eliminates the need to solve a constrained integer programming problem. Second, rather than attempting to maximize the value or volume of payments settled in the queue, I propose using information regarding the instantaneous benefits and costs of participants in order to define a welfare measure for any set of netted payments. The full benefits of these two changes are realized through an application of the Shapley value

[4] FIFO-Reorder involves moving a payment to the end of the queue or changing its priority code. FIFO-Bypass allows the operator to bypass a large payment if it cannot be settled due to a lack of funds, without the need to reorder payments.

[5] For more detailed descriptions of the FIFO and sorted-queue offsetting algorithms, see Fugal et al. (Section 3, 2018).

cost allocation method, which ensures welfare maximizing netting proposals are always accepted.

2 Model and Notation

Denote the set of banks in the system by $N = \{1,...,n\}$. Netting proposals are made to the banks by the queue operator at the end of regularly scheduled time intervals, called netting cycles, throughout the day.[6] Define a matrix P that represents all the payment obligations submitted to the queue during the current netting cycle. Let element p_{ij} of P denote the vector of payments from bank i to bank j, with typical element p_{ijk}, where $k = 1,...,t^{ij}$, and t^{ij} denotes the number of payments from i to j. Assume each payment p_{ijk} from bank i to bank j has a constant, instantaneous delay cost that defines bank i's current per-dollar benefit b_{ijk} of processing the payment as an immediate atomic transaction versus having it remain unsettled and awaiting inclusion in another (future) proposal or being reassigned for gross settlement. Likewise, assume that each bank i has a known per-dollar cost of providing liquidity at the current settlement opportunity, which I denote by c_i. The cost of providing liquidity at the current time may differ across banks and intraday.

Using these parameters, banks can assess the instantaneous benefits and costs of any netting proposal. To demonstrate how, I must first formally define what a netting proposal is. I begin by defining some related concepts. A netting set $G \subseteq P$ is any subset of payments in the queue. Given a netting set G, let $N(G)$ denote the set of banks that send or receive payments in G. The liquidity requirement L^G of a given netting set G can be obtained by

[6] Alternatively, the timing of proposals could be event driven (i.e., occur when a certain value of payments in the queue is reached or when potential welfare benefits from clearing the queue reach a certain level). The analysis presented in this note applies to any situation where there exists a given set of payments in a queue that need to settled.

adding up the net debit positions of all banks. Specifically, let

$$d_i^G = max\left\{0, \sum_{j \neq i} \sum_{k: p_{ijk} \in G} p_{ijk} - \sum_{j \neq i} \sum_{k: p_{jik} \in G} p_{jik}\right\}. \tag{1}$$

The first double-summation term in (1) adds up all of the outgoing payments from i in G, and the second double-summation term in (1) adds up all of the incoming payments to i in G. The max operator sets credit positions to zero. Hence, $L^G = \sum_{i \in N(G)} d_i^G$.

Definition 1: A netting proposal is a set of payments $G \subseteq P$ and liquidity contributions ℓ_i^G, $i = 1, ..., N(G)$ such that $\sum_{i \in N(G)} \ell_i^G = L^G$.

Let $\ell^G = (\ell_1^G, ..., \ell_{N(G)}^G)$. The net benefit to bank i of any netting proposal (G, ℓ^G) is given by

$$U_i(G, \ell^G) = \sum_{j \neq i} \sum_{k: p_{ijk} \in G} b_{ijk} p_{ijk} - c_i \ell_i^G. \tag{2}$$

Given a netting set G, one way to specify the vector ℓ^G would be to assume that liquidity always comes from the cheapest possible sources according to the bank-specific cost parameters, c_i. This would minimize the overall cost of liquidity provision. An alternative approach is to assume that banks provide liquidity equal to their net debit positions specified by (1); i.e., set $\ell_i^G = d_i^G$, for $i = 1, ..., n$. This approach conforms most closely to existing practices, and it could be preferred in a world where the c_i parameters have to be estimated. Either approach can be used to define liquidity provision in a way that meets the feasibility requirement in Definition 1 (the latter approach does so trivially), but neither method ensures that net benefits to all participants under a given proposal are positive. To achieve this, we need side payments. In what follows, I introduce side payments in the environment where banks provide liquidity equal to their (positive) net debit positions specified by (1).

The analysis can be easily redone following the other approach.

2.1 Proposals with Side Payments

Let us begin the discussion of side payments with a very simple example in which there are only two participants in the system, bank A and bank B. Suppose bank A owes \$100 to bank B and bank B owes \$80 to bank A. If bank A and bank B are able to net these obligations, then they will need only \$20 worth of liquidity. In existing queue systems, the required \$20 in liquidity would be drawn, if it were available, from bank A, since bank A is the bank in a net debit position. Is this fair? If bank A did not submit its payment, then bank B would have to provide \$80 worth of liquidity. And, of course, if bank B did not submit its payment, then bank A would have had to provide \$100 worth of liquidity. The point is that both participants are made better off due to the netting, so perhaps they both should contribute to the liquidity cost. Furthermore, bank A might have an expectation that bank B should contribute to the liquidity cost and hence not accept a proposal that does not meet this expectation.

We need a fair way of allocating costs associated with obtaining the joint benefit of netting proposals. Economists often turn to the Shapley value (Shapley (1953)) in situations like this. There is a solid justification for using the Shapley value in cost allocation problems. Roth and Verrecchia (1979) show that, under reasonable assumptions, it provides the same expected utility the participants would expect to get from bargaining to an uncertain outcome.

To define the solution suggested by the Shapley value cost allocation mechanism, I must define the Shapley value of each participant in the game $\Gamma = (N(G), v)$ generated by the netting set G, where v is the *characteristic function* of the game. The characteristic function describes the maximum total net benefit that can be obtained by any group of banks $S \subseteq N(G)$ by settling any subset (including the full set) of their combined payment obligations on a net basis. For any $S \subseteq N(G)$, let $H(S)$ denote all nonempty sets $H \subseteq G$ that can be formed using only payments in G that are to and from members of S. Then

$$v(S) = max\left\{0, max_{H \in H(S)} \sum_{i \in S} \left[\sum_{j \neq i} \sum_{k: p_{ijk} \in H} b_{ijk} p_{ijk} - c_i d_i^H\right]\right\}. \tag{3}$$

That is, $v(S)$ is computed as the sum of the benefits from payments settled minus the combined costs of supplying the liquidity required to settle them. The first *max* operator is there because banks can always decide to settle no payments, so the value of any coalition S has to be greater than or equal to zero. The second *max* operator reflects the possibility that the maximum net benefit from settling payments in a set G may be obtained by settling a subset of these payments.

For any $G \subseteq P$, the Shapley value of the game $\Gamma = (N(G), v)$ is given by

$$w_i = \sum_{S \subseteq N(G)} \frac{(s-1)!(n-s)!}{n!} \left[v(S) - v(S-i)\right], \tag{4}$$

where s is the number of banks in group S. The Shapley value of bank i is the weighted sum of the terms $[v(S) - v(S-i)]$, which represent bank i's marginal contribution to coalition S. It can therefore be interpreted as a bank's expected marginal contribution to a coalition of banks that seek to net payments, based on the assumption that each bank's sequential arrival to the coalition is determined randomly.

Following the Shapley cost allocation method, I now define each bank i's share of the cost burden of providing liquidity to settle payments under the proposal (G, d^G), where $d^G = (d_1^G, ..., d_{N(G)}^G)$. I assume

$$G \in argmax_{H \in H(N(G))} \sum_{i \in N(G)} \left[\sum_{j \neq i} \sum_{k: p_{ijk} \in H} b_{ijk} p_{ijk} - c_i d_i^H\right], \tag{5}$$

which ensures that G is used in the specification of $v(N(G))$ in (3). This eliminates the need to introduce additional notation to to capture the case where (5) is not satisfied and is not a restriction. Under this assumption, bank i's share of the cost burden of providing liquidity to settle payments under the proposal (G, d^G), is defined as the gross benefit to bank i of having its own payments in G settled minus its Shapley value:

$$C_i^G = \sum_{j \neq i} \sum_{k : p_{ijk} \in G} b_{ijk} p_{ijk} - w_i. \tag{6}$$

A property of the Shapley value is that it assigns the total value of the coalition to its members (efficiency), so that the sum over all banks of the values defined in (4) equals $v(N(G))$. It follows that the sum of the cost shares in (6) over all of the banks cover the actual cost of the proposal: i.e., $\sum_{i \in N(G)} C_i^G = \sum_{i \in N(G)} c_i d_i^G$.

To implement these cost shares, I propose the use of side payments. That is, each bank provides liquidity equal to its net debit position and then side payments are made so that the final cost share to each bank equals the amount specified by (6). Formally, side payments from bank i to bank j associated with any solution of the type given by (6) can be defined as

$$\sigma_{ij} = \begin{cases} 0 & \text{if } c_i d_i^G \geq C_i^G \text{ or } c_j d_j^G \leq C_j^G \\ (C_i^G - c_i d_i^G) \frac{c_j d_j^G - C_j^G}{\sum_{k : c_k d_k^G > C_k^G} c_k d_k^G - C_k^G} & \text{otherwise.} \end{cases} \tag{7}$$

This takes the amount that bank i owes (if anything) and distributes it among all the banks that require side payments proportionally based on what each bank is owed, relative to what all banks that require side payments are owed. This adds up because if we assume $c_j d_j^G > C_j^G$ and sum σ_{ij} over i such that $c_i d_i^G < C_i^G$, then the summation terms in the numerator and denominator cancel out, leaving the desired amount $c_j d_j^G - C_j^G$ that bank j needs to receive. Likewise, if we assume $c_i d_i^G < C_i^G$ and sum σ_{ij} over j such that $c_j d_j^G > C_j^G$, then the summation terms in the numerator and denominator cancel out, leaving the desired amount $C_i^G - c_i d_i^G$ that bank i needs to pay.

3 Individual Rationality

Side payments defined using the Shapley value cost allocation method result in individual payoffs for each member of the netting set that are greater than what they can achieve by opting out. Let $(G, d^G, (\sigma_{ij}))$ denote a netting proposal with side payments and let $U_i(G, d^G, (\sigma_{ij})) = U_i(G, d^G) - \sum_{j \in N(G): j \neq i} (\sigma_{ij} - \sigma_{ji})$ denote bank i's corresponding utility. Then we have

Theorem 1. *Consider a netting proposal with side payments $(G, d^G, (\sigma_{ij}))$ with σ_{ij} defined according to (7) for all $i, j \in N(G)$. If $\sum_{i \in N(G)} U_i(G, d^G) \geq 0$, then $U_i(G, d^G, (\sigma_{ij})) \geq 0$ for all $i \in N(G)$.*

Proof. For any $S \subset N(G)$ and $S' \subset N(G)$, with $S \cap S' = \emptyset$, let $H \in H(S)$ denote the netting set that defines $v(S)$, and let $H' \in H(S')$ denote the netting set that defines $v(S')$, in both cases according to (3). The netting set options $H(S \cup S')$ include the netting set that is formed by combining H and H'. Hence, $v(S \cup S') \geq v(S) + v(S')$. This establishes that the characteristic function proposed in (3) is superadditive, which is a sufficient condition for the Shapley value to be an imputation (trivially, set $S' = i \notin S$). Finally, observe that under netting proposals with side payments defined according to (7) for all $i, j \in N(G)$, each bank's payoff is equal to its Shapley value. □

The importance of Theorem 1 derives from the fact that, for an arbitrary netting set $G \subseteq P$, it may be the case that $\sum_{i \in G} U_i(G, d^G) \geq 0$, but $U_j(G, d^G) < 0$ for some j. That is, a netting set delivers strictly positive total surplus but would be rejected by at least one member of the netting set in the absence of side payments. This possibility is illustrated in Example 2, below.

4 Examples

The first two examples are based on the scenario that I used to motivate side payments in Section 3. I use this simple scenario to convey not only the desirability of side payments, but also, in some cases, the necessity. The third example provides a slightly more general illustration of the main result and demonstrates the fact that the it will sometimes be optimal to exclude subsets of payments that do not generate surplus from the netting proposal.

Example 1. Suppose bank A owes \$100 to bank B and bank B owes \$80 to bank A. Assume that the benefit per dollar to each bank of settling those payments at the current time is $b = .05$. In addition, assume that the instantaneous, per-dollar cost of liquidity provision is the same for both banks and is equal to $c = .1$.[7] Consider the netting proposal $(G, \ell^G) = (\{p_{AB}, p_{BA}\}, (\ell_A^G, \ell_B^G)) = (\{100, 80\}, (20, 0))$. In this simple setting, $v(A) = v(B) = 0$ because the payoff specified inside the bracket of (3) that each bank receives from processing its own payment individually is negative. Hence the banks, acting alone, are better off not sending payments. In contrast, $v(A,B) = \$180 \times .05 - \$20 \times .1 = \$7$ and hence, acting together, the banks are better off making their payments. The marginal contribution of each bank to the coalition (A,B) is \$7, and each bank's average marginal contribution is \$3.5. The cost share of each bank is equal to the benefit the bank receives from having its individual payments settled minus its Shapley value: $C_A^G = .05 \times 100 - 3.5 = 1.5$ and $C_B^G = .05 \times 80 - 3.5 = .5$. This is achieved by having bank A provide \$20 in liquidity at cost \$2 and then having bank B make a \$0.50 side payment to bank A.

[7] Both b and c should be small relative to magnitudes of liquidity provided. They reflect only the costs and benefits of settlement now versus at an uncertain point in the near future. It is also reasonable to set b smaller than c. This is consistent with assumptions made in Bech and Garratt (2003, 2012). If it were not, then one might expect the bank to settle the payment via the standard RTGS stream.

This simple example demonstrates how side payments defined using the Shapley value create fairness. In the absence of side payments, the net benefits to each bank would be $.05 \times 100 - .1 \times 20 = 3$ for bank A and $.05 \times 80 = 4$ for bank B. With side payments, the net benefit to banks A and B are equal; they are computed as $.05 \times 100 - .1 \times 20 + .5 = 3.5$ and $.05 \times 80 - .5 = 3.5$, respectively. So with side payments both banks benefit equally from their cooperation, while without side payments they do not.

Example 2. Consider the same scenario as Example 1, but assume that the instantaneous benefits and costs for each bank are $b = .025$ and $c = .15$, respectively. It is not profitable to process payments individually, so $v(A) = v(B) = 0$. Even though I raised the cost of processing payments and lowered the benefit, the total benefit of netting payments is still positive: i.e., $v(A,B) = \$180 \times .025 - \$20 \times .15 = \$1.5$. The marginal contribution of each bank to the coalition (A,B) is \$1.5 and, hence, each bank's average marginal contribution is \$0.75. The cost shares of the banks are $C_A^G = 0.025 \times 100 - .75 = 1.75$ and $C_B^G = 0.025 \times 80 - .75 = 1.25$. This is achieved by having bank A provide \$20 in liquidity at cost \$3 and then having bank B make a \$1.25 side payment to bank A.

In this scenario, side payments are required for the netting proposal to be acceptable to both banks. In the absence of side payments, the net benefits to each bank would be $.05 \times 100 - .15 \times 20 = -.5$ for bank A and $.05 \times 80 = 2$ for bank B. With side payments, the net benefit to banks A and B are both positive (and equal); they are computed as $.025 \times 100 - .15 \times 20 + 1.25 = .75$ and $.025 \times 80 - 1.25 = .75$, respectively. So, it is only in the case of side payments that the gains to netting payments are realized.

Example 3. Consider a three-bank scenario in which each bank has entered multiple payments into the queue. Specifically, bank A owes \$70 to bank B and \$110 to bank C, bank

B owes \$10 to bank A and \$30 to bank C, and bank C owes bank \$60 to bank A and \$30 to bank B. As in the previous example, assume that all the payments are the same priority, the benefit per dollar to each bank of settling each of their payments at the current time is $b = .05$, and the instantaneous, per-dollar cost of liquidity provision is the same for all banks and is equal to $c = .1$. The maximum payoff to the grand coalition is obtained by excluding the \$70 payment from bank A to bank B and the \$10 from bank B to bank A, since these payments do not create any other netting opportunities, and, on their own, they cost more in terms of liquidity than they deliver in benefits. We therefore consider the netting proposal $(G, \ell^G) = (\{p_{AC}, p_{BC}, p_{CA}, p_{CB}\}, (\ell^G_A, \ell^G_B, \ell^G_C)) = (\{110, 30, 60, 30\}, (50, 0, 0))$. It is straightforward to show that $v(i) = 0$ for $i \in N$, $v(A,B) = 0$, $v(B,C) = 3$, $v(A,C) = 3.5$ and $v(A,B,C) = 6.5$. To demonstrate the computation of the Shapley value for each bank, I list all of the possible orderings of the banks and then take the average over all orderings of the marginal contributions of each bank to the total net benefit. Since there are three banks, there are six possible orderings, as shown in Table 1.

Table 1: Shapley value equals the average of each bank's marginal contributions over all of the different orderings.

Order	MC A	MC B	MC C
A,B,C	0	0	6.5
A,C,B	0	3	3.5
B,A,C	0	0	6.5
B,C,A	3.5	0	3
C,A,B	3.5	3	0
C,B,A	3.5	3	0
Shapley value	1.75	1.5	3.25

Using the Shapley values from Table 1, the cost shares of the banks are given by $(C^G_A, C^G_B, C^G_C) = (.05 \times 110 - 1.75, .05 \times 30 - 1.5, .05 \times (60 + 30) - 3.25) = (3.75, 0, 1.25)$. These cost shares can be realized by having bank A provide \$50 in liquidity at cost \$5 and then having bank C make a \$1.25 side payment to bank A.

5 Concluding Remarks

I make a proposal for a liquidity savings mechanism that involves take-it-or-leave-it proposals from a centralized queue operator that specify netting sets, liquidity provision and side payments, and argue that such a scheme would be welfare improving. The potential for welfare improvement relies on the ability of the center to know the benefit and cost parameters of the participants. On the cost side, this seems reasonable because the center can measure the intraday cost of liquidity and has access to each participant's overall liquidity position throughout the day. It is less likely that the center would know a participant's instantaneous benefits from processing payments. These benefits are private information that relate to customer motives for making payments. However, the bank may have information regarding the payment type, and this can reveal something about how urgent it is. In separate work, I am investigating the possibility that the center might be able to learn about banks' instantaneous costs of liquidity provision and benefits of processing payments by examining participants' responses to proposals.

There are two clear advantages to having the center make proposals as opposed to placing this responsibility in the hands of the participants (see Fugal et al. (2018)). One key advantage is that the center can see the entire payment file, which is necessary to define netting sets. Another advantage derives from the fact that the game described in this paper is not typically convex. As such there is no guarantee that allocations derived from the Shapley value are in the core (see Shapley (1971)) and hence, in a decentralized setting, there could be difficulty in reaching agreements on netting proposals.[8] The main advantage to having individual banks make proposals arises from the fact that the proposer knows its own private information and can incorporate that information into the proposal. This advantage would

[8] Consider an example with four banks where bank A owes \$10 to bank B and \$1 to bank D, bank B owes bank \$10 to bank C, bank C owes \$10 to bank A and \$10 to bank D, and bank D owes \$1 to bank A and \$10 to bank B. Set the benefit to all banks equal to $b = .1$ and assume $c > b$. Then $v(BC) = 0$, $v(ABC) = v(BCD) = 3$, and $v(ABCD) = 3.2$, thus violating convexity.

become less important as the number of participants involved in a proposal increases.

References

Morten Bech and Rodney Garratt, “The Intraday Liquidity Management Game,” *Journal of Economic Theory*, April 2003, 109(2), 198-219.

Morten Bech and Rodney Garratt, “Illiquidity in the Interbank Payment System Following Wide-Scale Disruptions,” *Journal of Money, Credit Banking*, August 2012, 44(5), 903-929.

Adam Fugal, Rodney Garratt, Zhiling Guo and Dave Hudson, “A Proposal for a Decentralized Liquidity Savings Mechanism with Side Payments,” R3 Reports, June 2018.

Ben Norman, “Liquidity Saving in Real-Time Gross Settlement Systems–An Overview,” Bank of England Financial Stability Paper, 2010, No. 7.

Alvin Roth and Robert Verrecchia, “The Shapley Value as Applied to Cost allocation: A Reinterpretation,” *Journal of Accounting Research*, Spring 1979, 17(1), 295-303.

Lloyd Shapley, “A Value for n-Person Games,” In *Contributions to the Theory of Games II, Annals of Mathematics Study 28*, Harold Kuhn and Albert Tucker, eds, Princeton University Press, 1953, 307-317.

Lloyd Shapley, “Cores of Convex Games,” *International Journal of Game Theory*, 1971, 1(1), 1126.