EUROPEAN CENTRAL BANK
EUROSYSTEM

Robert Hofmeister*
European Central Bank

Argyris Kahros*
European Central Bank

A compression-style liquidity saving mechanism

Decreasing Liquidity Requirements and Participant Liquidity Management Needs with a Position Netting Service

*NOTE: The views expressed below are those of the authors and do not necessarily reflect those of the ECB.

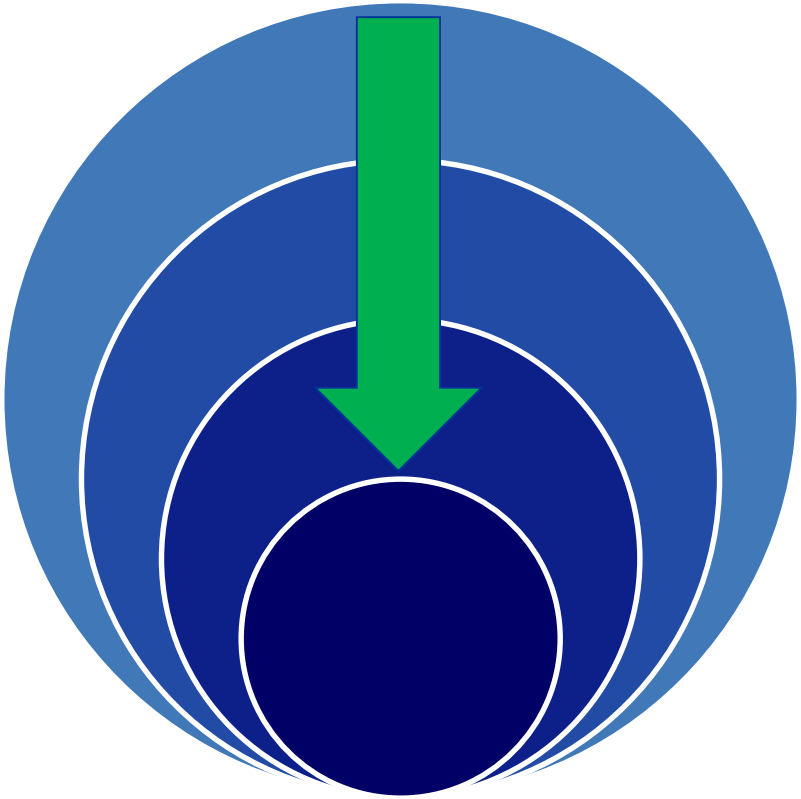
Disclaimer

The author(s) of this presentation is(are) members/alternates of one of the user groups with access to TARGET2 data in accordance with Article 1(2) of Decision ECB/2010/9 of 29 July 2010 on access to and use of certain TARGET2 data. The ECB and the PSSC have checked the paper against the rules for guaranteeing the confidentiality of transaction-level data imposed by the PSSC pursuant to Article 1(4) of the above mentioned issue.

The views expressed in the paper are solely those of the author(s) and do not necessarily represent the views of the Eurosystem.

Work in progress, results are preliminary.

Why another type of LSM?

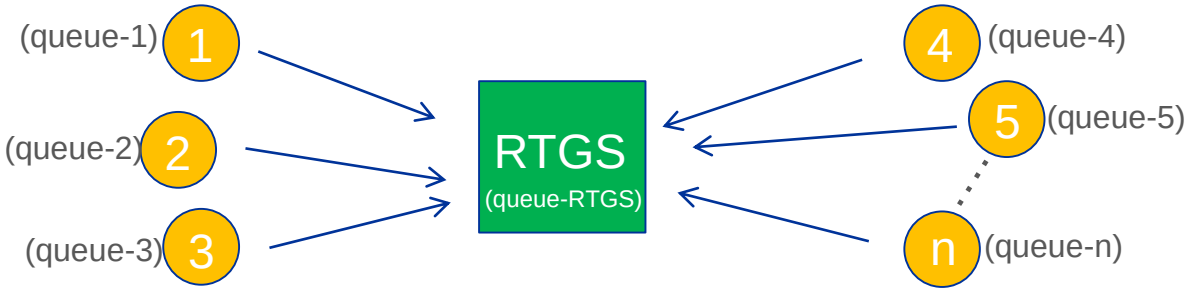


Manage your net exposure…

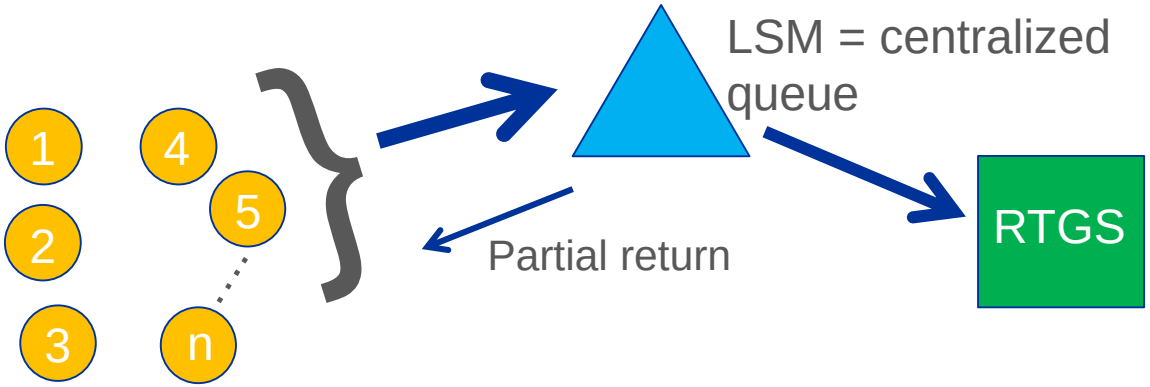
… now also for payments.

Background and motivation

- Assumption: participants of RTGS systems have actively managed internal queues in order to minimise costly liquidity usage.



- Our concept: if all payments were centrally known, offsetting payments could be identified and a large fraction could be settled with minimal liquidity usage (thanks to netting effects).



Literature

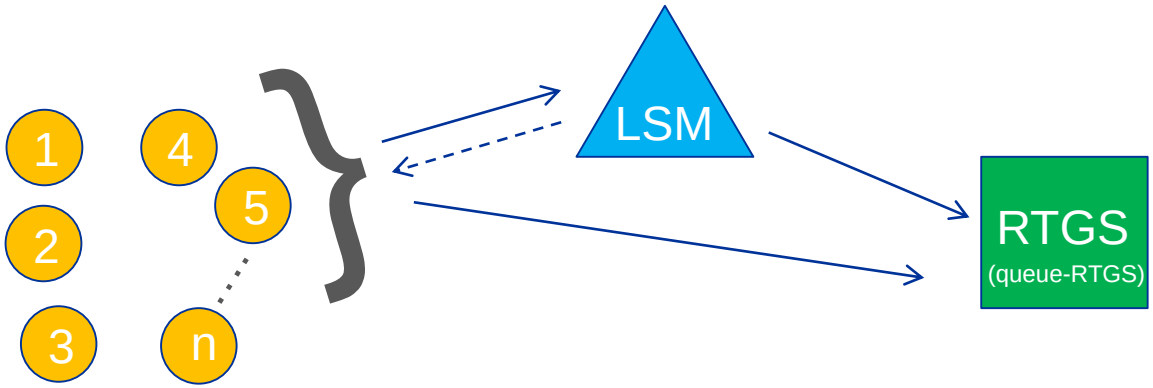
- Theoretical papers
  - Bech, Garratt (2003) “intraday liquidity management game”, Bech (2008): late settlement often an equilibrium in theoretical models
- Empirics on timing
  - McAndrews, Rajan (2000): coordination of Fedwire payments -> focal points
  - Armantier, Arnold, McAndrews (2008): payments settle later, timing of realisation unknown
  - Becher, Galbiati, Tudela (2008): peaks before times of throughput guideline
- Liquidity saving mechanisms and other enhancements
  - Martin, McAndrews (2008), Jurgilas, Martin (2010) – queues can help
  - Johnson, McAndrews, Soramäki (2004): receipt-reactive LSM

Literature

- Anecdotal evidence
  - Uncertainty about (amount and timing of) incoming payments and customer payments
  - McAndrews, Rajan (2000): banks charge customers -> incentives passed on
  - Armantier, Arnold, McAndrews (2008): waiting for Fedwire funds, delay due to operational costs
  - Becher, Galbiati, Tudela (2008): cost of delay low, payments delayed after time-critical ones made

Concept

- Establish position netting service separate from core RTGS settlement
- Participants submit payments and set liquidity thresholds
- LSM identifies and settles “feasible” payments
  - feasible payments are identified as a set of payments resulting in **net transfers within** the **given thresholds** – similar to compression of derivatives
  - **other payments** are **returned** to participants (or released into RTGS system)



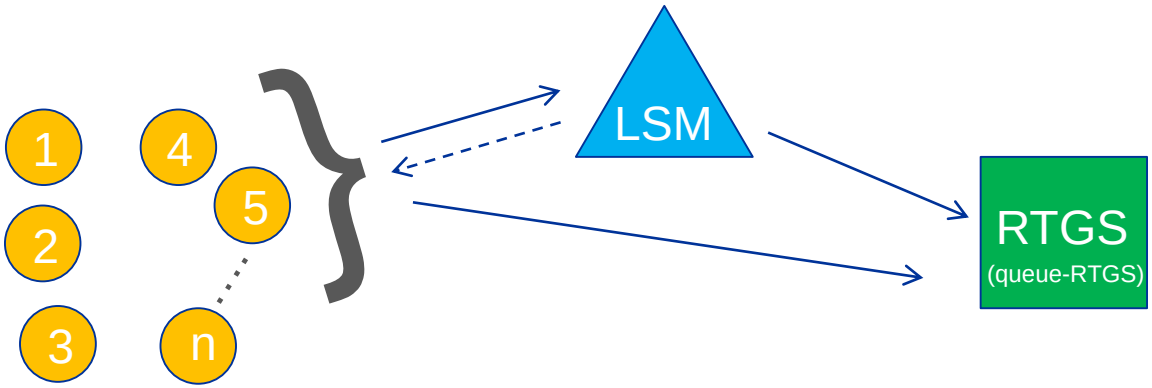
→ Participants can submit all payments while committing only a small amount of liquidity

→ Participants remain free to manage unsettled payments

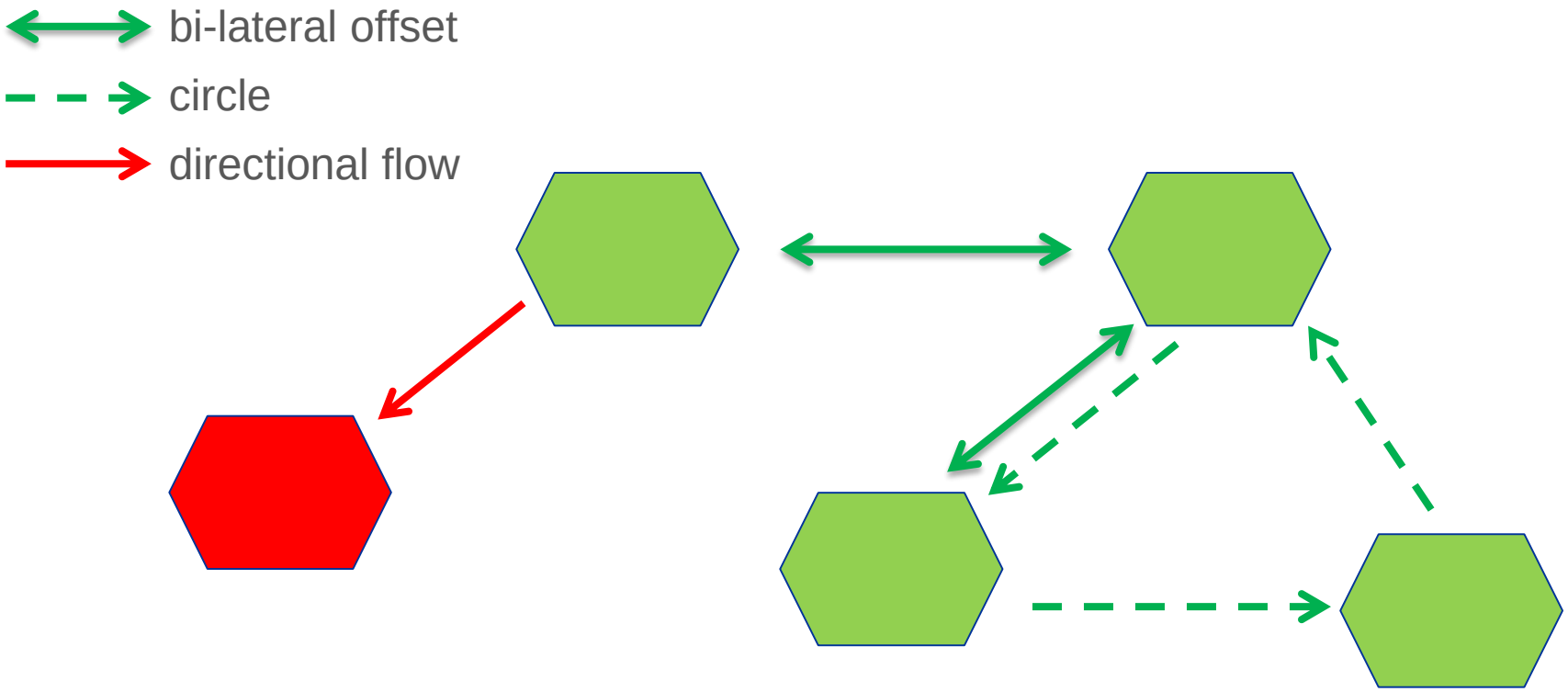
Concept

Potential timeline

- by 6:00am participants submit payments and set liquidity level
- by 7:00 compression algorithm runs, outcome is settled, remaining liquidity and net inflows are returned to participants, unsettled payments are cancelled



Compression-style LSM: identifying offsetting positions



Benefits

- Payments settle earlier
  - Reduction of operational risk
  - Participants are better informed about the liquidity situation (open outgoing and possibly incoming payments)
- Higher liquidity efficiency
- Reduced liquidity management needs

Comparison with alternative solutions

- RTGS system with queue
  - ! available liquidity to be exhausted before queues are formed
- DNS
  - ! liquidity of net receivers locked up in DNS until settlement
  - ! at settlement net senders face liquidity needs
- Receipt-reactive LSM
  - ! participants commit to recycle incoming liquidity
- Compression-style LSM
  - + partial settlement allows for **flat positions**
  - + **full control** over liquidity exposure

Compression LSM: A first implementation

Own algorithm* to identify a feasible set

- Starting assumption that all payments can be settled → usually some participants have insufficient liquidity
- Remove payments one by one, until a solution is found
  1) Remove transactions from senders in negative to participants in positive
  2) If (1) is not possible, remove transactions to participants who can pass on the shortfall to participants being in positive

→ Subset of payments with net transfers below liquidity levels

- Run post-solution optimization (future extension)
  - Add in payments on a bi-lateral basis
  - Add in single payments

* Optimization ongoing

Data for LSM

Real TARGET2 data, March 2015

- Transactions
  - customer & interbank payments (tran classes 1.1 & 1.2)
  - excluding money market transactions (Arciero et al., 2013)
  - random selection of transactions, e.g. 50%

- Liquidity
  - Fraction of daily balances and (lowest value of) intraday liquidity, e.g. 1%

Results of the baseline configuration

- Sample: random draws of 2 March 2015
- Share of payments: 50% of customer payments and 80% of interbank payments
- Liquidity level: 1% of liquidity in TARGET2*

Mean % settled	Min %	Max %	Max settable %
57.3	50.7	63.3	75.9

* participant‘s opening balance plus lowest intraday credit limit.

Varying liquidity levels

- Sample: 10 random draws of 2 March 2015 (50% of customer payments and 80% of interbank payments)

Liquidity levels	Mean % settled	Min %	Max %	Max settable %
5%	71.6	68.7	77.3	80.6
2%	62.0	57.5	65.7	78.0
1%	57.3	50.7	63.3	75.9

Rest of Day analysis

How does the removal of LSM-settled transactions affect the settlement of other transactions?

- transactions not settled by LSM to be reintroduced into regular payment day
- potential aspects: liquidity needs, payment delay, unsettled transactions

Problem: comparing the original day to one with transactions randomly removed is not meaningful.

- some participants will have incoming transactions removed (predominantly) in the morning and outgoing ones in the afternoon -> timing mismatch

Proposed solution: smart randomisation of the timing of payments for the day with reduced AND all transactions.

Rest of Day analysis, cont.

Effect of randomisation:

- timing mismatches now also present for day with all transactions
- all results expected to be worse than in original, unaltered (and not actively managed) day

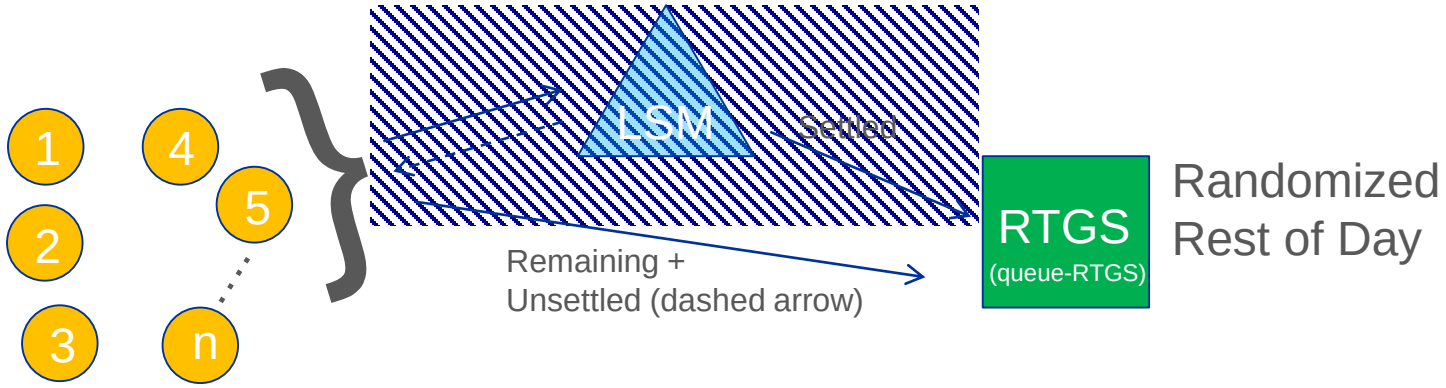
Technical implementation:

1. Group transactions into 5 classes (up to 100,000; 10^6; 10^7; …)
2. For each group, identify when 1%-quantiles are settled → 100 time points
3. Randomly allocate transactions to a quantile

→ time distribution of payments roughly identical to real timing of each value group

LSM + Rest of Day versus full day simulations

- Full day runs: randomized versions of a day in March 2015
- Rest of Day runs: remaining payments plus all transactions not settled in the LSM session.



- Data & LSM algorithm:
  - LSM: 50 draws of 2 March 2015, 50% of customer and 80% of interbank paym.
  - Rest of Day: ALL transactions on 2 March not settled by the LSM (also ancillary systems payments, etc.)
  - Full Day: All transactions from 2 March
  - Early version of the LSM, settling only 20% to 30% of submitted transactions

LSM + Rest of Day versus full day simulations: percent transactions settled

	Effective Full Day (i.e., LSM + Rest of Day)		Actual Full Day	
	By Value	**By Volume**	**By Value**	**By Volume**
Average	94.65	98.97	95.04	98.68
Std. Deviation	0.47	0.41	0.45	0.44
Max	96.02	99.70	95.96	99.33
Min	93.82	97.81	93.98	97.75

LSM + Rest of Day versus full day simulations: liquidity needs

- Liquidity needs were calculated as follows:
  - LSM session: $\sum_{i}^{n} Max\{0, (\beta_i - \varepsilon_i)\}$
  - Full Day and Rest of Day Sessions: $\sum_{i}^{n} (\beta_i - \mu_i)$

n: number of participants
β_i: beginning of day balance for bank i
ε_i: end of day balance for bank i
μ_i: minimum balance for bank i

- Maximum liquidity savings for LSM + Rest of Day sessions: 13.5%
- Average liquidity savings: ~2%

Rest of Day versus full day simulations: queue statistics

	Peak Queue Value	Avg Queue Value	Avg Queue Duration
Average	38,581,649,739	1,129,011	1,004,072,660
Std. Deviation	4,294,382,033	230,881	2,263,093,972
Max	49,127,679,391	1,805,686	9,622,801,000
Min	32,065,622,438	677,935	4,318,000

Rest of Day

	Peak Queue Value	Avg Queue Value	Avg Queue Duration
Average	37,457,738,383	1,142,294	959,191,800
Std. Deviation	3,715,359,199	198,513	2,124,106,078
Max	45,423,501,726	1,453,169	9,022,739,000
Min	28,527,199,823	705,573	3,830,000

Actual Full Day

Further work: strategic sensitivity of LSM result

- Can participants manipulate the results by strategically submitting payments?
  - e.g. by holding back one (or more) payment(s)
- If yes, what are the incentives and possible consequences?

Proposed analysis:

- Run a set of payments twice: once with and once without removing one (particular) transaction of one (particular) participant

The end

Thank you for your attention!

Questions and comments welcome!

Supplement: varying the share of payments

- Liquidity level = 1%

Customer payments %	Interbank payments %	Mean % settled	Min %	Max %	Max settable %
100	100	65.2			86.5
80	80	61.1	57.7	66.4	83.1
50	80	57.7	50.7	63.3	76.2
50	50	60.3	49.0	70.8	74.9

Supplement – grouping transactions by value

Value range	count	
(0 to 100,000]	4,696,475	
(100,000 to 1,000,000]	783,247	
(1,000,000 to 10,000,000]	379,421	
(10,000,000 to 100,000,000]	128,434	
(100,000,000 to 1,000,000,000]	41,844	Classes merged
> 1,000,000,000	3,029	

List of operation types

- 0.0 No operation type
- 1.1 Customer payments
- 1.2 Interbank payments
- 2.1 Cash operation
- 2.2 Intraday repo and similar transactions
- 2.3 Payments sent and/or received on behalf of customers
- 2.4 Inter NCB payments
- 2.5 Other transactions
- 3.1 Trade by trade settlements of SSS
- 3.2 Other settlement operations
- 3.3 EBA Euro1
- 3.4 CLS
- 3.5 EBA Step2
- 4.1 Intraday transfers with LVPS
- 4.2 Intraday transfers with retail systems
- 4.3 Intraday transfer with SSS
- 4.4 Technical transfers between different accounts of the same participant
- 4.5 Commercial transfers between different accounts of the same participant
- 4.6 Transfers to T2S
- 4.7 Transfers back to TARGET2 from T2S