Online-Appendix to
Inferring Trade Directions in Fast Markets

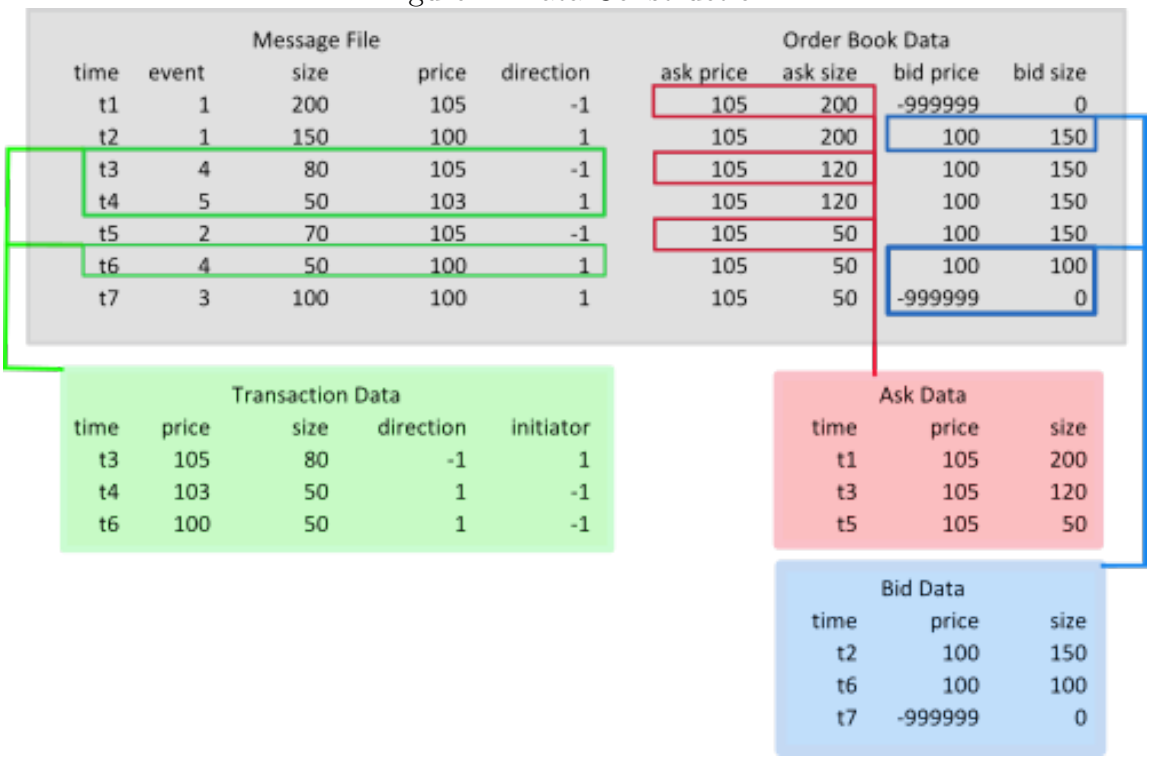
Simon Jurkatis*

*E-mail: simon.jurkatis@bankofengland.co.uk. Tel.: +44 203 461 8588.

1 Extracting Trade and Quote Data from LOBSTER's Message Files

The software LOBSTER reconstructs from the original NASDAQ TotalView-ITCH data feed the full limit order book, as well as a message file containing information on the events causing the changes in the order book.[1]

Figure 1: Data Construction



Message File

time	event	size	price	direction
t1	1	200	105	-1
t2	1	150	100	1
t3	4	80	105	-1
t4	5	50	103	1
t5	2	70	105	-1
t6	4	50	100	1
t7	3	100	100	1

Order Book Data

ask price	ask size	bid price	bid size
105	200	-999999	0
105	200	100	150
105	120	100	150
105	120	100	150
105	50	100	150
105	50	100	100
105	50	-999999	0

Transaction Data

time	price	size	direction	initiator
t3	105	80	-1	1
t4	103	50	1	-1
t6	100	50	1	-1

Ask Data

time	price	size
t1	105	200
t3	105	120
t5	105	50

Bid Data

time	price	size
t2	100	150
t6	100	100
t7	-999999	0

Notes: Each row in the Message File describes the cause of the change in the order book from the previous row to the next. Event types 1, 2 and 3 refer to the submission, partial cancellation and total deletion of a limit order, events 4 and 5 to the execution of a visible and hidden limit order, respectively. The direction 1 (-1) refers to a buy (sell) limit order.

The gray shaded area in Figure 1 provides an example of the design of LOBSTER's message and order book files. The k-th row of the message file describes the cause of the change in the order book from the $(k-1)$-th row to the k-th row. The events 1, 2 and 3 refer to the submission, partial cancellation and total deletion of a limit order. The events 4 and 5 refer to the execution of a visible and hidden limit order, respectively. The direction indicates whether a buy (+1) or sell (-1) limit order is affected. If a hidden order is executed, the order book is not visibly

[1]For more information on the TotalView-ITCH data feed and the order book reconstruction by LOBSTER, see Hautsch and Huang (2012) and Huang and Polak (2011).

affected. In that case, to maintain a symmetric output, the LOBSTER order book data displays the order book's state after the execution of the hidden order.

As an example take the first row of the order book and message file. We start here with an empty order book which I indicated by negative quotes. At `t1` the message file indicates a submission of a limit sell order for a price of `105` per share for a total of `200` shares. In the same row, the order book displays its new state. The bid side is still empty and the ask side is now displaying the price and volume of the limit sell order.

Below the gray shaded area in Figure 1, it is illustrated how I extract the trade and quote data from the order book and message file. I construct the trade data by extracting all visible and hidden executions of limit orders (events 4 and 5) from the message file, with the respective information on the price and volume of the transactions. As the direction in the message file refers to the limit order, the initiator is given by the opposite party to the trade. Note that I omit the active counter-party to each trade from the trade data. In doing so, however, I do not omit any relevant information as the counter-party simply mirrors the passive trade with opposite trade direction. I remind the reader of that decision in the main text at any point where it is relevant, and discuss the alternative of including the counter-party to each trade.

The data for the ask side of the order book is constructed by extracting the state of the order book at any point the ask side is affected by the submission, cancellation, deletion or execution of a visible sell limit order (events 1, 2, 3, 4). Any event that is related to a hidden order is omitted. The construction of the bid side follows analogously.

Table 1: Ticker Names

AAPL	AA	ABB	ABT	ACE	ACN	ADBE	ADM	ADP
ADS	AEP	AGN	AGU	AIG	AKAM	ALK	ALL	AME
AMGN	AMT	AMX	AMZN	AN	AON	AOS	APC	APD
APH	ASH	ASR	AVGO	AVY	AXP	AYI	AZN	BAC
BAM	BAX	BA	BBL	BBT	BCE	BEAV	BEN	BHI
BHP	BIDU	BIIB	BK	BLK	BLL	BMS	BMY	BP
BRFS	BR	BTI	BT	BUD	BX	CAJ	CAT	CCK
CELG	CF	CHA	CHL	CHRW	CHT	CHU	CLX	CL
CMCSA	CMCSK	CME	CMI	CM	CNI	CNQ	COF	COP
COST	CPA	CPRT	CP	CRH	CRM	CSCO	CSGP	CSX
CTRP	CTSH	CUK	CVS	CVX	C	DAL	DCM	DD
DEO	DE	DHR	DISH	DIS	DOW	DTV	DUK	DVN
D	EBAY	ECL	EL	EMC	EMR	ENB	ENR	EOG
EPD	ESRX	ETE	ETN	EXC	EXPD	E	FCX	FDX
FIS	FLT	FMX	F	GD	GE	GG	GILD	GIS
GLW	GMCR	GM	GOOG	GPK	GPN	GPRO	GSK	GS
GT	GWR	HAL	HD	HMC	HON	HPQ	HSY	IBM
IBN	IGT	ILMN	IMO	INFY	INTC	IP	IR	ITW
JAH	JBHT	JBLU	JCI	JPM	KAR	KMB	KMX	KO
KR	KSU	K	LBTYA	LBTYK	LEG	LFL	LLY	LMT
LNKD	LOW	LO	LUV	LVS	LYB	MA	MCD	MCK
MELI	MET	MGA	MHK	MJN	MMC	MMM	MON	MOS
MO	MPC	MRK	MSCI	MSFT	MS	MT	NCR	NEE
NGG	NKE	NLSN	NOC	NSC	NTES	NTT	NUE	NVO
NVS	ODFL	ORCL	OXY	PAC	PBR	PCAR	PCLN	PCP
PEP	PFE	PG	PHG	PH	PKG	PKX	PM	PNC
POT	PPG	PRU	PSA	PTR	PX	QCOM	RAI	REGN
RIO	RKT	ROP	RTN	RYAAY	SAP	SAVE	SBUX	SCCO
SCHW	SIAL	SLB	SNE	SNP	SNY	SON	SO	SPB
SPG	SRE	STO	STT	STZ	SU	SWFT	SYT	SYY
TEF	TEL	TEVA	TGT	TJX	TMO	TM	TOT	TRP
TRV	TSLA	TSM	TSS	TS	TTM	TWC	TWX	TXN
T	UAL	UL	UNH	UNP	UN	UPS	USB	UTX
VALE	VFC	VLO	VMW	VRX	VZ	V	WFC	WHR
WIT	WMB	WMT	WM	WPZ	WU	XOM	XRX	YHOO
YUM	Z							

Notes: This table provides the ticker names of all stocks included in the sample. However, not all of these stocks are analyzed over the whole range of the sample as some stock-days may not have fulfilled the criteria mentioned in the data section (day-end price $\geq$ 1 \$ and number of trades $\geq$ 10), or due to an initial public offering during the sample period (e.g. Z).

2 Adjusting the FI Algorithm to Different Data Structures

This section presents an intermediate data structure to the ones that are presented in the main paper, keeping assumption (ii) from Data Structure 1, but using assumption (i) from Data Structure 2:

Data Structure 3. *Aggregated Quote Changes*

(i) At the time of a trade, the order book displays the new state of the order book after the completion of all transactions that were carried out due to the same buy or sell order.

(ii) Trades and quotes are reported in the correct order.

The change in the data structure means that we cannot use the strict equality between the transaction volume and the change in volume at the quote to eliminate potential matches. If an order for 100 shares trades against two limit orders for 50 shares each, posted at the same price, the trade data record two transactions for 50 shares each, while the order book data show a decrease in volume at the respective quote by 100 shares.

Hence, we change the search for a match among the ask quotes in Step 2 (see main paper) of the algorithm to

$$\boldsymbol{\alpha} = \min\{j \in \mathcal{J}_a \colon p_i = a_j \text{ and } v_i \leq \Delta v_j^a\},$$

and analogously for the bid.

If we find a match at one or both sides of the order book, the algorithm proceeds as before. If, however, we cannot find a match on either side of the order book, we need to insert two additional steps before we can conclude that we apparently face a transaction involving a hidden order, which would be classified under Step 4.

Consider a market order for a number of shares greater than what is available at the best quote. The trade data will show the corresponding transaction at the next-best quote, but the order book data will not show any decrease in volume at that quote. In the extreme case, where the market order is so large that it will go through several levels of the order book, the order book data will not even show the quotes against which the order executed on its way to the last quote.

To accommodate these cases the adjusted algorithm injects two additional searches for a match at the ask or bid side before it proceeds with Step 4. The first search (demonstrated for the ask) under Step 4a is conducted as

$$\widetilde{\boldsymbol{\alpha}} = \min\{j \in \mathcal{J}_a \colon p_i = a_j \text{ and } a_{j-1} < a_j\}.$$

In case we find a match on one or both sides of the order book we proceed as prescribed by Step 2.[2]

The second additional search for a match among the quotes if we cannot find one under Step 4a, is conducted under Step 4b (again demonstrated for the ask) as

$$\hat{\boldsymbol{\alpha}} = \min\{j \in \mathcal{J}_a \colon p_i > a_j \text{ and } a_{j+1} < p_i\},$$

and analogously for the bid, proceeding exactly as under Step 4a if a match on one or both sides can be found. If again neither a match at the ask side nor the bid side can be found, we are likely facing a hidden order and the classification is derived under Step 4 as before.

2.1 Classification Accuracy

Table 2 presents the classification accuracy of the FI algorithm adjusted to Data Structure 3 (FI_3), as well as for the FI version adjusted to Data Structure 2 presented in the main paper (FI_2) which could be used here as well, although it does not take full advantage of the information in this data structure.

[2]Note that if we classify the transaction according to the interpolated time, we do not adjust the volume at the matched quote, as there was no corresponding volume change to begin with.

Table 2: Classification Accuracy at Different Timestamp Precisions - DS 3

	timestamp precision 10^{-i} of a second with $i =$					
	0 (s)	1	2	3 (ms)	4	9 (ns)
Panel A: total volume						
FI_3	93.37	96.16	97.71	98.30	98.25	98.21
FI_2	91.73	94.84	96.82	97.92	98.22	98.21
EMO	90.14	93.63	96.08	97.61	98.17	98.19
CLNV	90.13	93.64	96.08	97.61	98.18	98.20
LR	89.84	93.56	96.02	97.51	98.09	98.09
EMOi	69.69	72.36	77.58	86.09	93.14	93.11
CLNVi	69.28	72.01	77.29	85.91	93.10	93.07
LRi	67.93	71.06	76.58	85.41	92.87	92.84
Panel B: average volume						
FI_3	93.42	95.93	97.18	97.65	97.54	97.45
	(2.64)	(2.02)	(1.96)	(2.06)	(2.05)	(2.09)
FI_2	92.33	94.86	96.43	97.40	97.52	97.45
	(2.97)	(2.24)	(2.00)	(2.05)	(2.05)	(2.09)
EMO	89.39	92.69	94.92	96.55	97.42	97.42
	(3.40)	(3.01)	(2.85)	(2.54)	(2.07)	(2.10)
CLNV	89.49	92.75	94.93	96.51	97.45	97.43
	(3.42)	(2.99)	(2.86)	(2.61)	(2.07)	(2.11)
LR	89.27	92.64	94.77	96.29	97.24	97.19
	(3.57)	(3.10)	(3.02)	(2.78)	(2.28)	(2.34)
EMOi	73.81	76.75	81.10	87.57	92.81	92.86
	(8.26)	(7.96)	(6.69)	(4.32)	(2.60)	(2.63)
CLNVi	72.48	75.59	80.16	86.97	92.70	92.75
	(7.68)	(7.52)	(6.51)	(4.45)	(2.76)	(2.80)
LRi	69.78	73.44	78.46	85.81	92.18	92.23
	(6.67)	(6.68)	(6.11)	(4.58)	(3.04)	(3.06)

Notes: This table shows the percentage of correctly classified trading volume under Data Structure 3 by the FI algorithm and the traditional algorithms using the last quotes from before the time of the trade (EMO, CLNV and LR) and using the interpolated time of trades and quotes (EMOi, CLNVi and LRi) as suggested by Holden and Jacobsen (2014). The algorithms are applied to the data with reduced timestamp precisions (10^{-i} of a second for $i = 0, \ldots, 4$) and using the original precision of nanoseconds (10^{-9} of a second). Panel A shows the percentage of correctly classified volume summed over the entire sample. Panel B shows the average of correctly classified volume over the 19842 stock-days with the standard deviations in brackets.

3 Measuring Order Imbalances

A frequent application where the initiator label enters the analysis is the estimation of the order imbalance

$$OI = \frac{V_B - V_S}{V},$$

where $V = V_B + V_S$ is the total trading volume, V_B is the volume of buyer-initiated trades and V_S the volume of seller-initiated trades. The order imbalance is often used, either directly or indirectly, as a measure of informed trading (see, e.g., Easley et al., 1996, 2012; Bernile et al., 2016; Brennan et al., 2018) or may be itself the variable of interest (e.g. Chordia et al., 2002; Dorn et al., 2008; Chordia et al., 2016).

To construct the order imbalance I split each stock-day into $\tau = 10, 100$ bins of equal volume size giving us $\tau \times 19842$ order imbalance estimates in total, with varying volume sizes across the stock-days.[3] Within each volume bin, the order imbalance is computed according to the above formula using the true trade initiator label and the labels provided by the classification algorithms.

As the order imbalance is often used as a measure of informed trading indicated by a large absolute order imbalance, we may be particularly interested in the estimation performance depending on the level of the true order imbalance.

Tables 3 presents the estimated absolute mean order imbalance and the root-mean-square error of the estimated order imbalance over all intervals and stock-days. The algorithms were applied to the data timestamped to the second. Figure 2 to 4 present the bias and root-mean-square-error of the estimated order imbalance depending on the level of the true order imbalance.

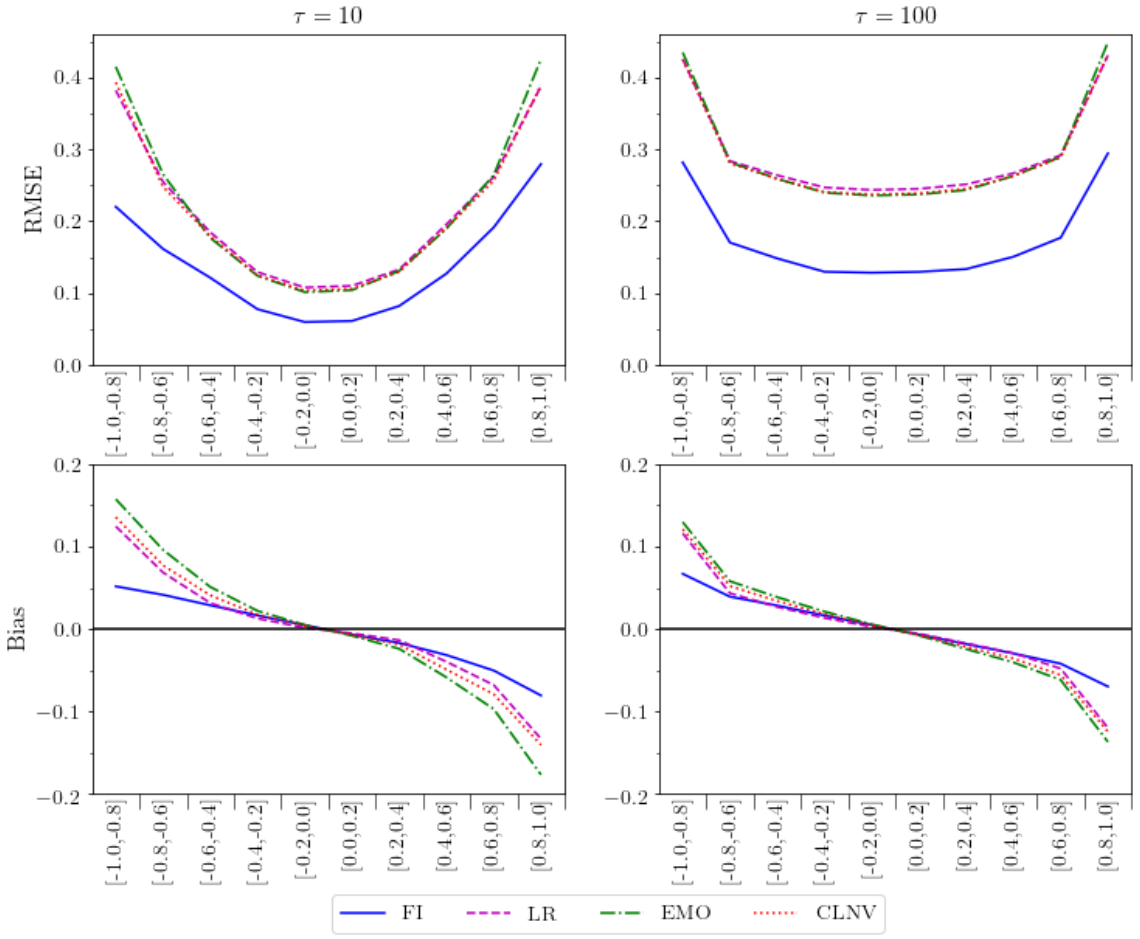
[3]Single trades are not split between intervals so certain differences in total transaction volumes between bins may remain.

Table 3: Mean absolute order imbalance and RMSE

		Mean absolute order imbalance					RMSE			
β	τ	true	FI	LR	EMO	CLNV	FI	LR	EMO	CLNV
Panel A: Data Structure 1										
	10	18.57	18.22	19.91	19.06	19.43	7.95	13.19	12.86	12.81
	100	38.30	37.92	41.76	40.77	41.12	16.38	28.00	27.79	27.58
Panel B: Data Structure 2										
0.0001	10	18.57	18.73	19.91	19.03	19.42	9.81	13.32	12.98	12.92
	100	38.30	39.12	41.76	40.75	41.11	21.23	28.26	28.03	27.82
0.0010	10	18.57	18.61	19.74	18.89	19.24	10.19	13.84	13.39	13.38
	100	38.30	39.00	41.63	40.61	40.95	22.09	29.18	28.77	28.67
0.0100	10	18.57	18.33	19.61	18.62	18.98	10.73	14.41	13.98	13.98
	100	38.29	38.61	41.54	40.25	40.61	23.32	30.13	29.90	29.82
0.1000	10	18.56	17.58	19.11	17.90	18.26	12.38	15.46	15.15	15.18
	100	38.23	37.38	40.95	39.25	39.65	27.47	32.71	32.80	32.79
Panel C: Data Structure 3										
	10	18.57	18.43	19.91	19.06	19.43	8.97	13.19	12.86	12.81
	100	38.30	38.35	41.76	40.77	41.12	18.54	28.00	27.79	27.58

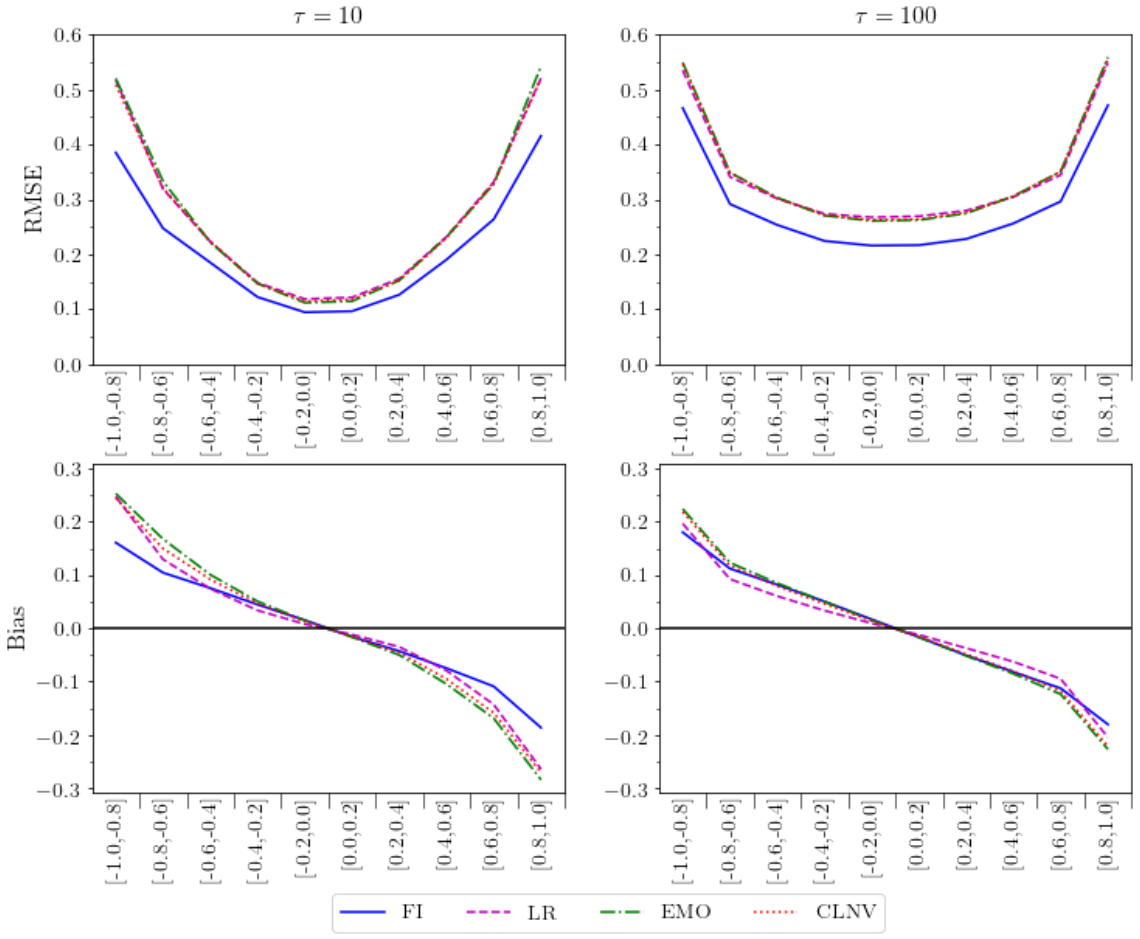
Notes: This table shows the mean absolute order imbalances measured using the true trade initiator label or estimated from the inferred trade direction by the respective algorithm. The algorithms have been applied to the data timestamped to the second. Under Data Structure 2 timestamps are subject to noise, $\varepsilon \sim \text{Exp}(1/\beta)$ and $\beta \in \{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}$. To measure the order imbalance each stock-day is split into τ equal volume bins. The mean absolute order imbalance is given $\sum_{\tau \times \mathcal{D}} |OI_i| / (\tau \times |\mathcal{D}|)$ where $\mathcal{D}$ is the set of all stock-days. The root-mean-square-error (RMSE) is given by $\sqrt{\sum_{\tau \times \mathcal{D}} (\widehat{OI}_i - OI_i)^2 / (\tau \times |\mathcal{D}|)}$, where $\widehat{OI}_i$ is the estimated order imbalance.

Figure 2: Order imbalance estimation depending on the true level of the order imbalance - DS1



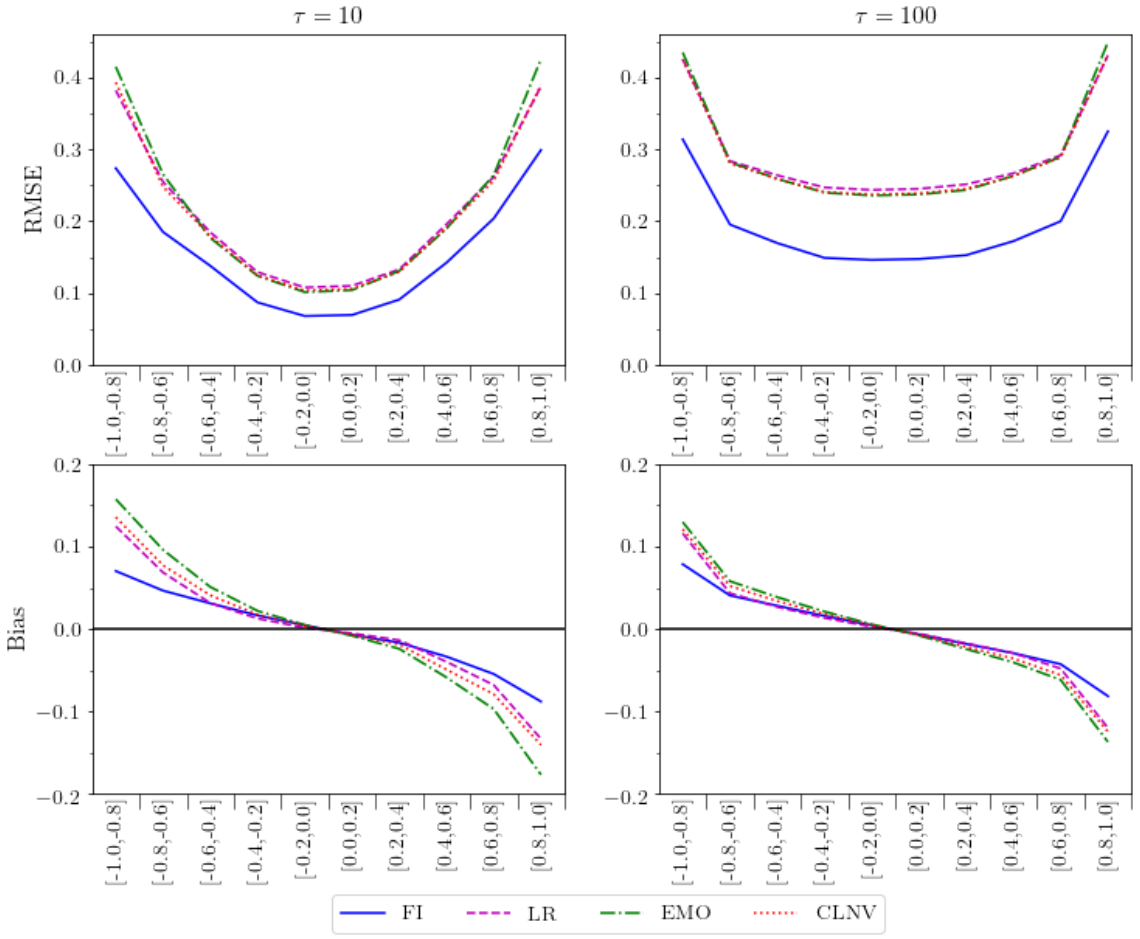
Notes: This Figure shows the bias and root-mean-square-error (RMSE) for the estimated order imbalances over different levels of the true order imbalance. The algorithms have been applied to the data timestamped to the second under Data Structure 1. To measure the order imbalance each stock-day is split into τ equal volume bins. The RMSE is given by $\sqrt{\sum_{j\in\mathcal{J}}(\widehat{OI}_j - OI_j)^2/|\mathcal{J}|}$, where $\mathcal{J}$ is the set of all bins with the true order imbalance OI being in a given range (x-axis) and $\widehat{OI}_j$ is the estimated order imbalance. The bias is given $\sum_{j\in\mathcal{J}}(\widehat{OI}_j - OI_j)/|\mathcal{J}|$.

Figure 3: Order imbalance estimation depending on the true level of the order imbalance - DS2



Notes: This Figure shows the bias and root-mean-square-error (RMSE) for the estimated order imbalances over different levels of the true order imbalance. The algorithms have been applied to the data timestamped to the second under Data Structure 2 with strong noise, $\varepsilon \sim \text{Exp}(1/\beta)$ and $\beta = 0.1$. To measure the order imbalance each stock-day is split into τ equal volume bins. The RMSE is given by $\sqrt{\sum_{j\in\mathcal{J}}(\widehat{OI}_j - OI_j)^2/|\mathcal{J}|}$, where $\mathcal{J}$ is the set of all bins with the true order imbalance OI being in a given range (x-axis) and $\widehat{OI}_j$ is the estimated order imbalance. The bias is given $\sum_{j\in\mathcal{J}}(\widehat{OI}_j - OI_j)/|\mathcal{J}|$.

Figure 4: Order imbalance estimation depending on the true level of the order imbalance - DS3



Notes: This Figure shows the bias and root-mean-square-error (RMSE) for the estimated order imbalances over different levels of the true order imbalance. The algorithms have been applied to the data timestamped to the second under Data Structure 2. To measure the order imbalance each stock-day is split into τ equal volume bins. The RMSE is given by $\sqrt{\sum_{j\in\mathcal{J}}(\widehat{OI}_j - OI_j)^2/|\mathcal{J}|}$, where $\mathcal{J}$ is the set of all bins with the true order imbalance OI being in a given range (x-axis) and $\widehat{OI}_j$ is the estimated order imbalance. The bias is given $\sum_{j\in\mathcal{J}}(\widehat{OI}_j - OI_j)/|\mathcal{J}|$.

References

Bernile, G., Hu, J., Tang, Y., 2016. Can information be locked up? Informed trading ahead of macro-news announcements. Journal of Financial Economics 121, 496–520.

Brennan, M.J., Huh, S.W., Subrahmanyam, A., 2018. High-frequency measures of informed trading and corporate announcements. The Review of Financial Studies 31, 2326–2376.

Chordia, T., Goyal, A., Jegadeesh, N., 2016. Buyers versus sellers: Who initiates trades, and when? Journal of Financial and Quantitative Analysis 51, 1467–1490.

Chordia, T., Roll, R., Subrahmanyam, A., 2002. Order imbalance, liquidity, and market returns. Journal of Financial economics 65, 111–130.

Dorn, D., Huberman, G., Sengmueller, P., 2008. Correlated trading and returns. The Journal of Finance 63, 885–920.

Easley, D., Kiefer, N.M., O'hara, M., Paperman, J.B., 1996. Liquidity, information, and infrequently traded stocks. The Journal of Finance 51, 1405–1436.

Easley, D., de Prado, M.M.L., O'Hara, M., 2012. Flow toxicity and liquidity in a high-frequency world. Review of Financial Studies 25, 1457–1493.

Hautsch, N., Huang, R., 2012. Limit Order Flow, Market Impact and Optimal Order Sizes: Evidence from NASDAQ TotalView-ITCH Data, in: Abergel, F., Bouchaud, J.P., Foucault, T., Lehalle, C.A., Rosenbaum, M. (Eds.), Market Microstructure: Confronting Many Viewpoints. Wiley, pp. 136–161.

Holden, C.W., Jacobsen, S., 2014. Liquidity measurement problems in fast, competitive markets: expensive and cheap solutions. The Journal of Finance 69, 1747–1785.

Huang, R., Polak, T., 2011. LOBSTER: Limit order book reconstruction system. Technical Report. School of Business and Economics, Humboldt Universität zu Berlin. URL: `https://lobsterdata.com/info/docs/LobsterReport.pdf`.