Trading Algorithms, Disorderly Markets and Non-Live Testing

A study of emergent behaviours supporting the case for non-live testing regulations

Nicholas Hallam and Nick Idelson, TraderServe Limited, 12-16 Laystall Street, London, EC1R 4PF.
e: research@traderserve.com w: www.traderserve.com t: 02071935717

Introduction

Regulatory concerns about the fair and orderly operation of global electronic markets have led ESMA to propose mandating more rigorous testing of algorithms and execution systems prior to live trading. The requirements envisaged go beyond mere conformance testing and also beyond, we argue, what is possible on legacy market simulation platforms which rely on passive replay of historical data. Such replay servers are fundamentally incapable of modelling the market response to orders emerging from the tested system – the factor most likely to produce market disorder – and they do not allow for the dynamic creation of extreme market conditions to identify the vulnerabilities of systems in circumstances that are met only infrequently.

We examine the behaviour of a number of algos using the TS-Arena emulated market. This platform offers a realistic and dynamic emulation which is statistically representative of the underlying market but which allows the emulated orders and those from the tested algos or systems to rest in the same order book, thus producing realistic market impact. Whilst maintaining a realistic emulation of the underlying market via its Market Emulator algo, TS-Arena also allows the simulation of extreme conditions of activity. This is valuable in assessing any hidden predisposition for an algo to contribute to a disorderly market.

We added and removed various algos from the TS-Arena as described below. In production the participant algos send their orders direct to the trading venue's test matching engine but for the purposes of this research we used our own limit and market order matching engine. All algos with the exception of the Market Emulator view only the TS-Arena data from our matching engine, and all orders are placed in price and time priority in the same order book. In this way we were able to explore the **interdependent** behaviour of multiple algos in realistic market conditions.

We illustrate how various types of predatory algos contribute to market disorder and how this is detectable in non-live testing. We then go further and demonstrate that it is not just deliberate market abuse that is problematic: we show apparently innocent algos to be conducive to the creation of a disorderly market either (1) in the presence of predators aiming to take unfair advantage of them or (2) in simulated conditions of extreme volatility or reduced liquidity.

Predatory Algos and their Market Activity

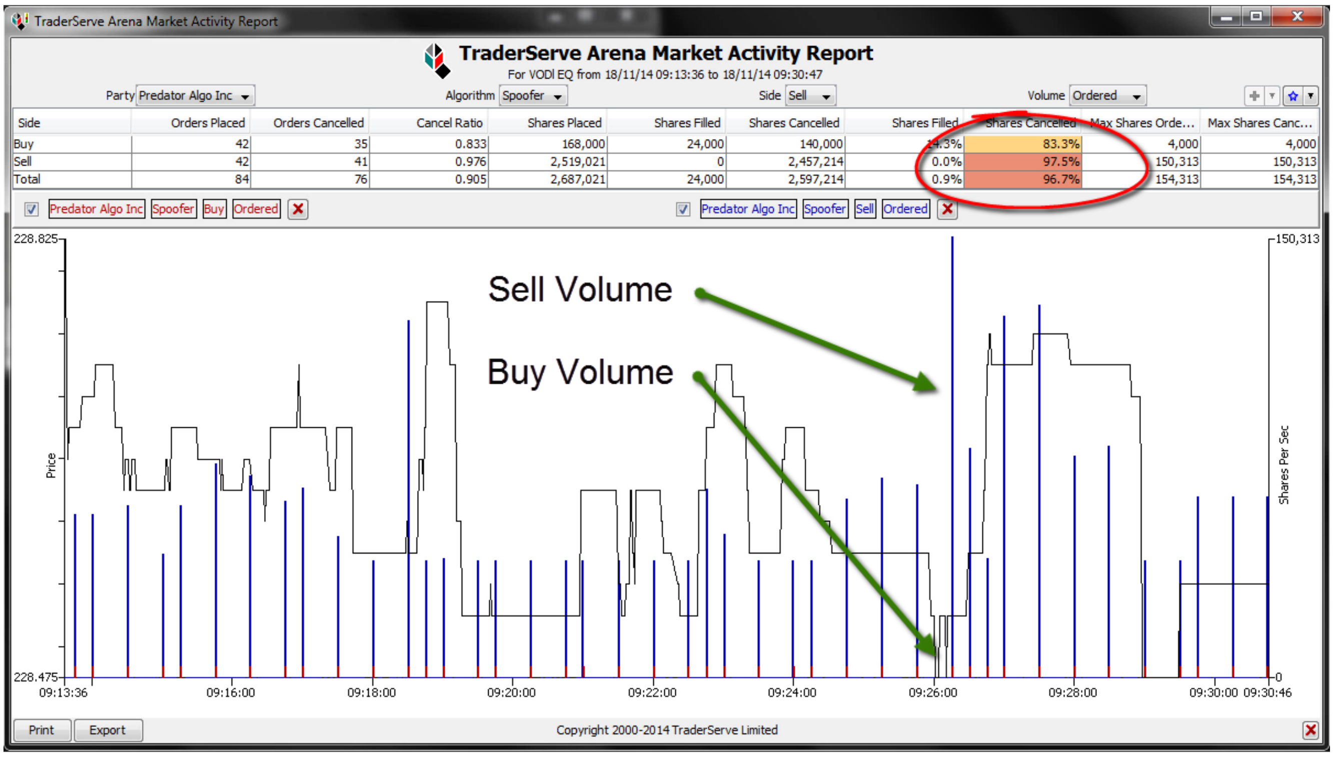
Spoofing

Spoofing is a type of abusive market behaviour. In one of its forms a large order is placed near the top of the order book to give the impression of substantial buying or selling interest. The intention is to provoke and profit from a price spike caused by the incorrect perception.

We added a simple “Spoofer” algo of this kind to the TS-Arena. This periodically posts a large sell order just above the best offer and, simultaneously, a small buy on the best bid, the aim being to fill the buy order on a spurious price move induced by the appearance of large selling interest.

After the algo had been running for a while we ran an activity report to examine trading. This report produces trading numbers, percentages of fills and cancels for the whole market, or broken down by Party and Algorithm. It generates charts of market activity and price filtered by party (here shown at one second granularity), algo, side and the type of activity (i.e. ordered, cancelled or traded). These can be overlaid to allow examination of any interdependencies.

Fig.1 shows two charts overlaid: in red, the ordering behaviour of the buy side of the algo and, in blue, the sell side order volumes. This shows that the sell orders are much larger than the buys. Also highlighted are the cancellation rates which are very high, particularly on the sell side and consistent with spoofing activity.



TraderServe Arena Market Activity Report

TraderServe Arena Market Activity Report

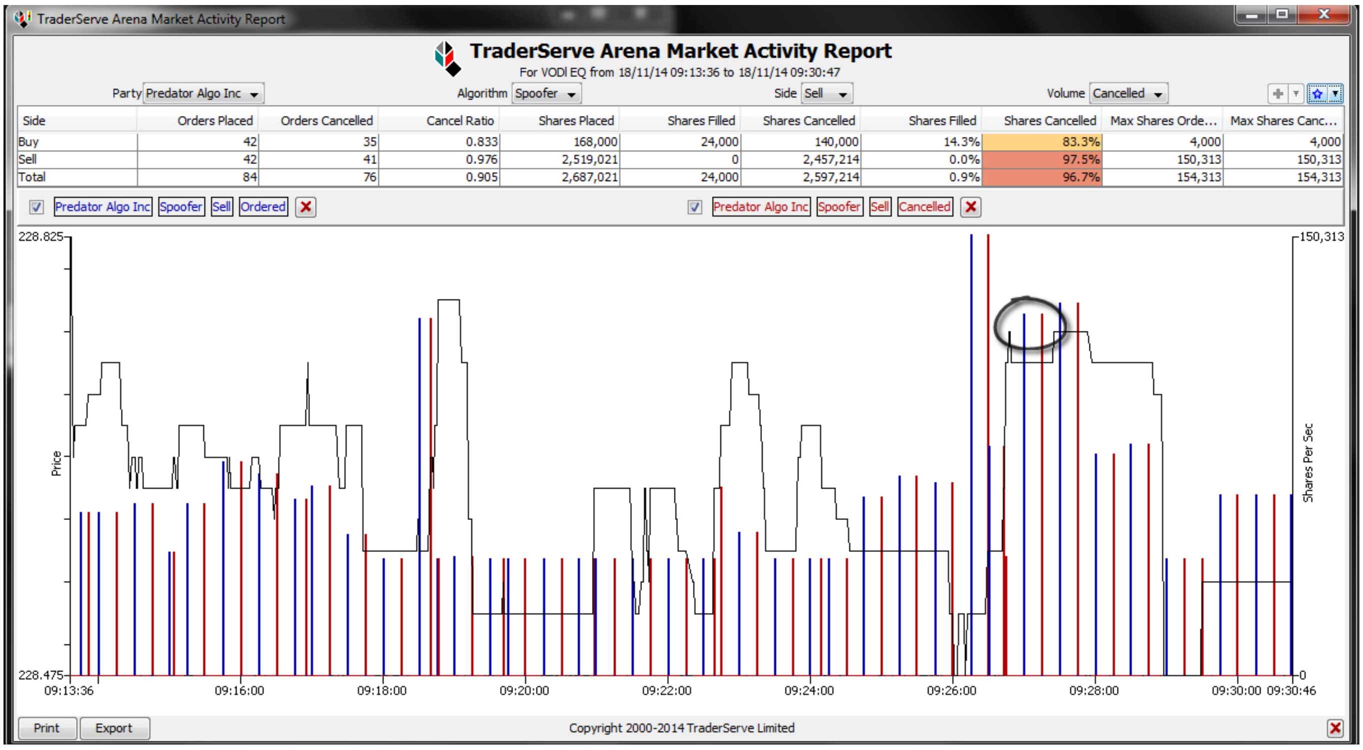
For VODl EQ from 18/11/14 09:13:36 to 18/11/14 09:30:47

Party Predator Algo Inc — Algorithm Spoofer — Side Sell — Volume Ordered

Side	Orders Placed	Orders Cancelled	Cancel Ratio	Shares Placed	Shares Filled	Shares Cancelled	Shares Filled	Shares Cancelled	Max Shares Orde...	Max Shares Canc...
Buy	42	35	0.833	168,000	24,000	140,000	14.3%	83.3%	4,000	4,000
Sell	42	41	0.976	2,519,021	0	2,457,214	0.0%	97.5%	150,313	150,313
Total	84	76	0.905	2,687,021	24,000	2,597,214	0.9%	96.7%	154,313	154,313

Fig.1

In Fig.2 the sell cancels (now in red) are shown against the sell orders. A clear pattern of orders and cancels of equal size emerges, which accounts for the very high cancellation rate. There is very little evidence of an intention to be filled on this side of the market.


TraderServe Arena Market Activity Report

TraderServe Arena Market Activity Report

For VODl EQ from 18/11/14 09:13:36 to 18/11/14 09:30:47

Party Predator Algo Inc Algorithm Spoofer Side Sell Volume Cancelled

Side	Orders Placed	Orders Cancelled	Cancel Ratio	Shares Placed	Shares Filled	Shares Cancelled	Shares Filled	Shares Cancelled	Max Shares Orde...	Max Shares Canc...
Buy	42	35	0.833	168,000	24,000	140,000	14.3%	83.3%	4,000	4,000
Sell	42	41	0.976	2,519,021	0	2,457,214	0.0%	97.5%	150,313	150,313
Total	84	76	0.905	2,687,021	24,000	2,597,214	0.9%	96.7%	154,313	154,313

Fig.2

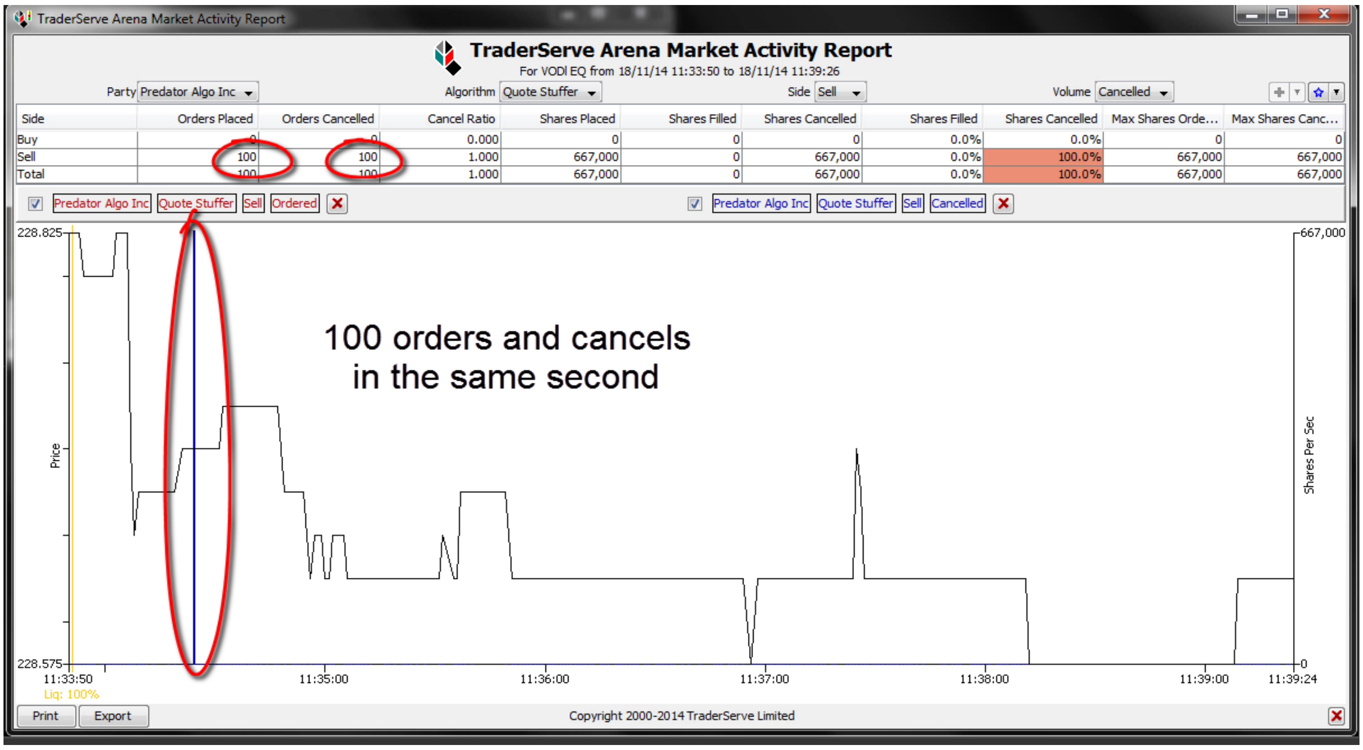
In the absence of market participants being fooled by the spoofing, there is little sign of market disorder, but as we shall see later successful spoofing as well as constituting market abuse does have market order implications.

Smoking and Stuffing

Quote stuffing is an insidious practice by which some High Frequency Traders try to gain an advantage over slower participants. In essence the idea is to increase the amount of processing required to digest the market data by posting large numbers of trades some way from the top of the order book (so there is no chance of their being filled) and cancelling them immediately.

We added a "Quote Stuffer" algo to the TS-Arena. This simply places a sell order at a distance above the best offer. When it receives the order book update showing that order in the market, it immediately cancels it. This process is repeated 100 times one after another.

Fig.3 shows the Market Activity report with a comparison between the Sell orders of the "Quote Stuffer" and the Sell cancels. The first thing to note is that there is a single order bar but 100 orders. All the orders have been posted in the same second. Secondly, the situation is identical with the cancellations, which perfectly overlay the orders. Indeed we can see that there is a 100% cancel rate on orders. These observations are strongly indicative of Quote Stuffing.


TraderServe Arena Market Activity Report

TraderServe Arena Market Activity Report

For VODl EQ from 18/11/14 11:33:50 to 18/11/14 11:39:26

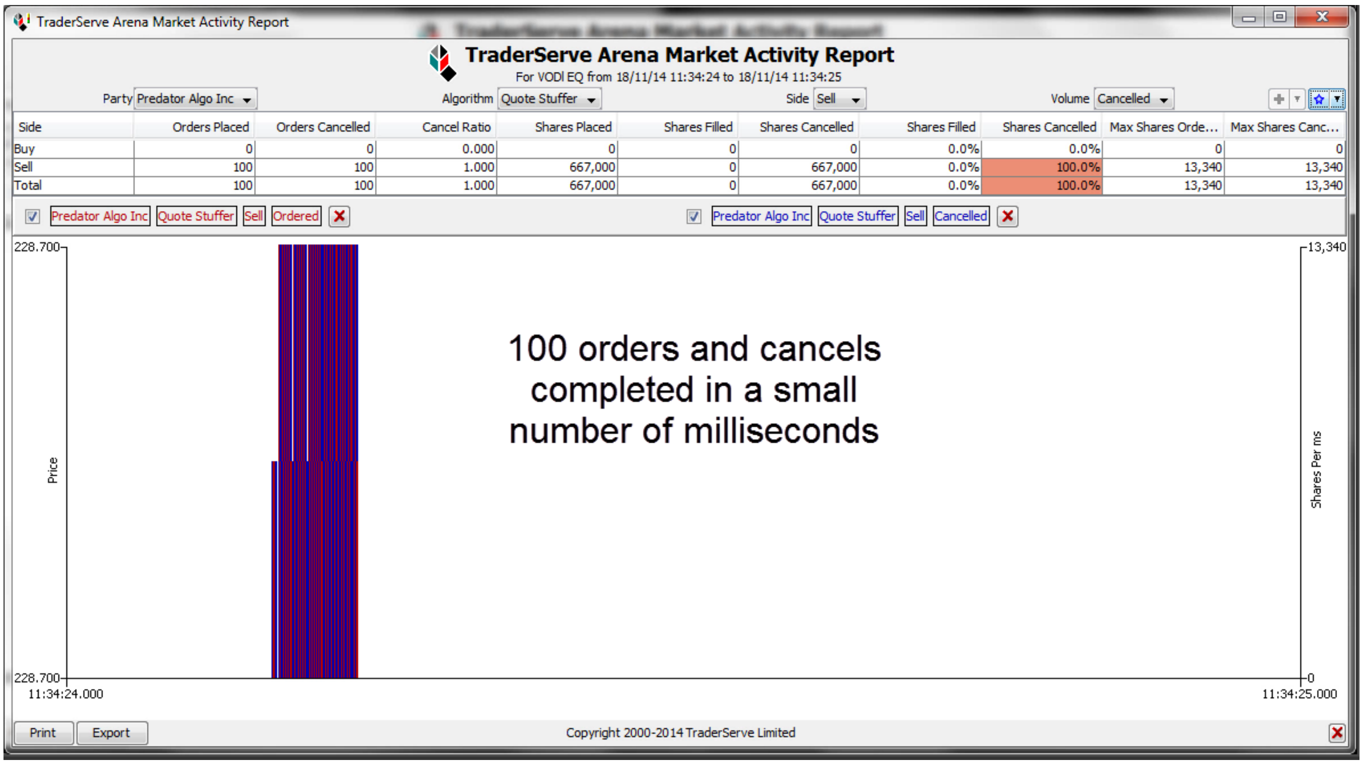
Party: Predator Algo Inc — Algorithm: Quote Stuffer — Side: Sell — Volume: Cancelled

Side	Orders Placed	Orders Cancelled	Cancel Ratio	Shares Placed	Shares Filled	Shares Cancelled	Shares Filled	Shares Cancelled	Max Shares Orde...	Max Shares Canc...
Buy	0	0	0.000	0	0	0	0.0%	0.0%	0	0
Sell	100	100	1.000	667,000	0	667,000	0.0%	100.0%	667,000	667,000
Total	100	100	1.000	667,000	0	667,000	0.0%	100.0%	667,000	667,000

Copyright 2000-2014 TraderServe Limited

Fig.3

Fig.4 shows the result of zooming into the second in question to view activity at the millisecond level. This gives a better picture of the extent of ordering and cancelling. The x-axis of the chart is one second wide, so we see that the stuffing activity (100 orders and 100 cancellations) is all complete within a few milliseconds.


TraderServe Arena Market Activity Report

TraderServe Arena Market Activity Report

For VODl EQ from 18/11/14 11:34:24 to 18/11/14 11:34:25

Party: Predator Algo Inc — Algorithm: Quote Stuffer — Side: Sell — Volume: Cancelled

Side	Orders Placed	Orders Cancelled	Cancel Ratio	Shares Placed	Shares Filled	Shares Cancelled	Shares Filled	Shares Cancelled	Max Shares Orde...	Max Shares Canc...
Buy	0	0	0.000	0	0	0	0.0%	0.0%	0	0
Sell	100	100	1.000	667,000	0	667,000	0.0%	100.0%	13,340	13,340
Total	100	100	1.000	667,000	0	667,000	0.0%	100.0%	13,340	13,340

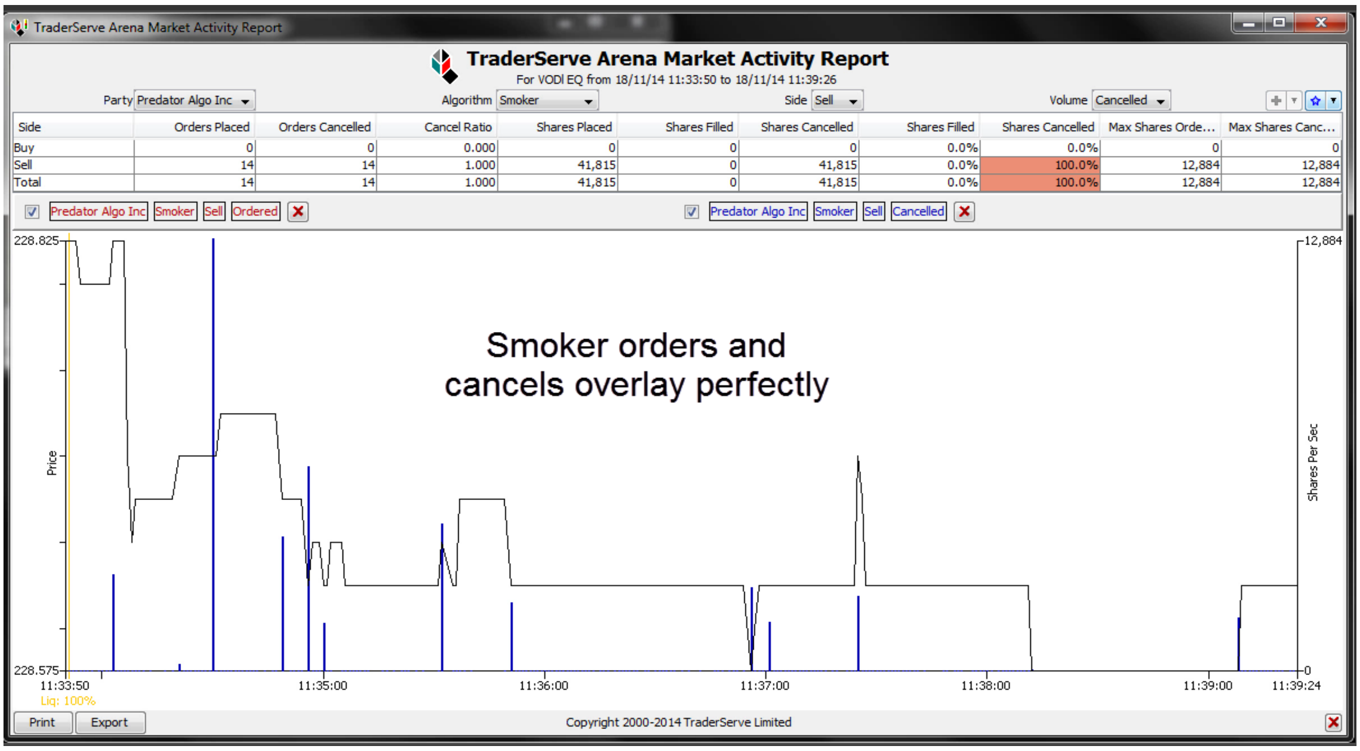
Copyright 2000-2014 TraderServe Limited

Fig.4

Another predatory algo was running at the same time. This is a “Smoker”. Smoking has many similarities to Quote Stuffing in that the predator tries to take unfair advantage by posting and immediately cancelling orders. Unlike the “Quote Stuffer” this is not performed to slow down the market but to draw orders from unwary traders by appearing to offer more attractive prices than are genuinely available.

The “Smoker” algo which we added to the TS-Arena monitors the width of the spread between the best bid and offer. When this is greater than one tick, it periodically puts in a sell order one tick below the best offer and then cancels this immediately. The idea is to draw greedy naïve traders into trades at the original best offer.

Fig.5 shows the sell side orders and cancels of the algo. We get a perfect overlay of cancels on orders (the blue volume overlay covers the red one). All orders were cancelled in the same second. Indeed, going down to the millisecond level we found that cancellations and orders occur within the same millisecond. In fact all orders are present in the market for a few microseconds only.


TraderServe Arena Market Activity Report

For VODl EQ from 18/11/14 11:33:50 to 18/11/14 11:39:26

Party: Predator Algo Inc Algorithm: Smoker Side: Sell Volume: Cancelled

Side	Orders Placed	Orders Cancelled	Cancel Ratio	Shares Placed	Shares Filled	Shares Cancelled	Shares Filled	Shares Cancelled	Max Shares Orde...	Max Shares Canc...
Buy	0	0	0.000	0	0	0	0.0%	0.0%	0	0
Sell	14	14	1.000	41,815	0	41,815	0.0%	100.0%	12,884	12,884
Total	14	14	1.000	41,815	0	41,815	0.0%	100.0%	12,884	12,884

Fig.5

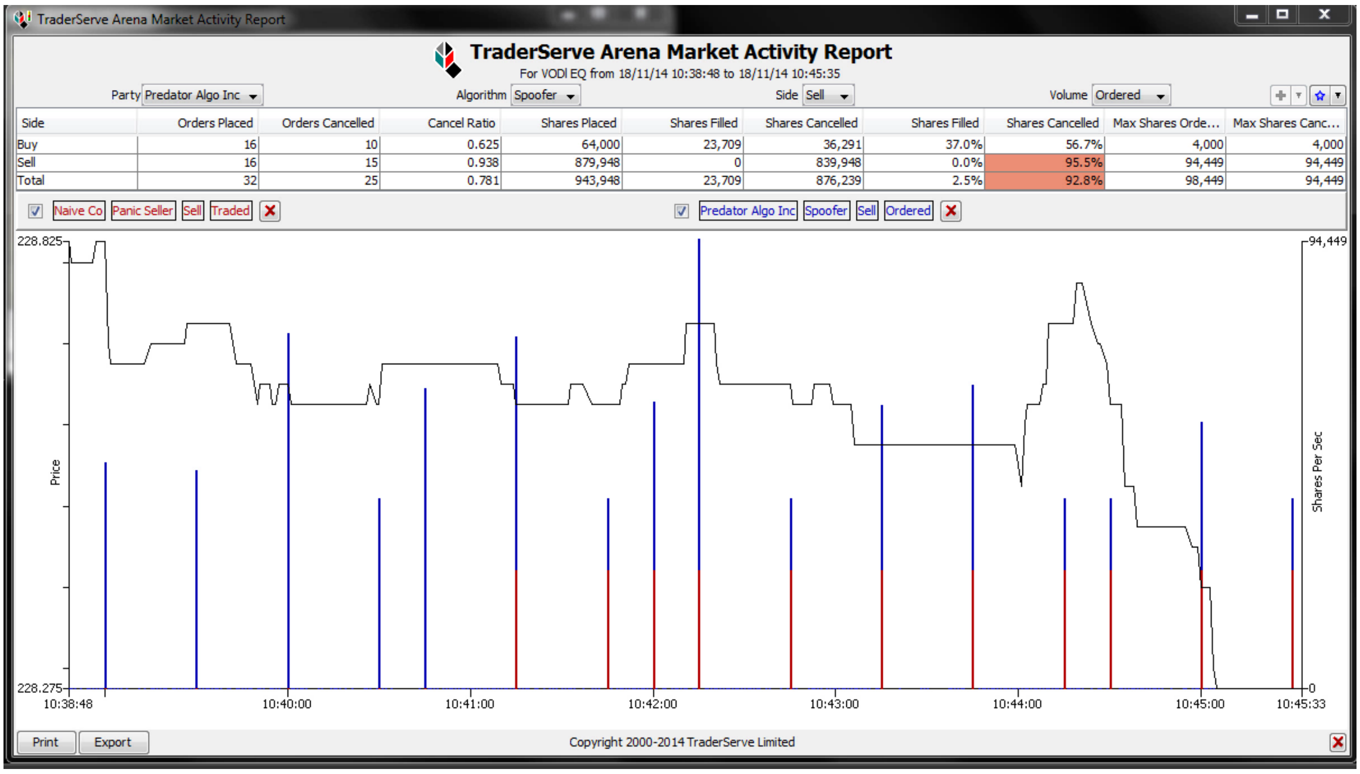
As in both of these examples, abusive behaviour often manifests itself in orders that are present in the market for a very short time. Such behaviour is typically intended to confuse and disrupt other less sophisticated participants.

Predatory Algorithms and their interaction with Innocent Algorithms can cause market disorder

Spoofer meets Spoofee

We return to the “Spoofer” algo to consider its interaction with other algos. Previously we noted that on its own the “Spoofer” was not harmful to market order.. Things changed, though, when we added another algo, which we called the “Naïve Panic Seller”. This algo is sensitive to downward pressure on price and looks to sell sizeable chunks of a long position when it judges that prices are about to fall. One of the things it is specifically looking for is sudden large increases in volume a little above the best offer.

Fig.6 overlays the sell side trades of the “Naïve Panic Seller” with the sell orders of the “Spoofer”, showing a good match between the two. The spoofing episodes are causing the “Naïve Panic Seller” to sell and we also begin to discern from the chart the downward pressure on price consequent to these events, so spoofing is not as innocuous as it previously seemed. The “Spoofer” and the “Naïve Panic Seller” have the potential together to create modest market disorder.


TraderServe Arena Market Activity Report

For VODl EQ from 18/11/14 10:38:48 to 18/11/14 10:45:35

Party Predator Algo Inc · Algorithm Spoofer · Side Sell · Volume Ordered

Side	Orders Placed	Orders Cancelled	Cancel Ratio	Shares Placed	Shares Filled	Shares Cancelled	Shares Filled	Shares Cancelled	Max Shares Orde...	Max Shares Canc...
Buy	16	10	0.625	64,000	23,709	36,291	37.0%	56.7%	4,000	4,000
Sell	16	15	0.938	879,948	0	839,948	0.0%	95.5%	94,449	94,449
Total	32	25	0.781	943,948	23,709	876,239	2.5%	92.8%	98,449	94,449

Fig.6

We conclude that spoofing is not harmless when there are naïve traders participating in the market. Later we shall see that naïve traders are capable of causing trouble on their own.

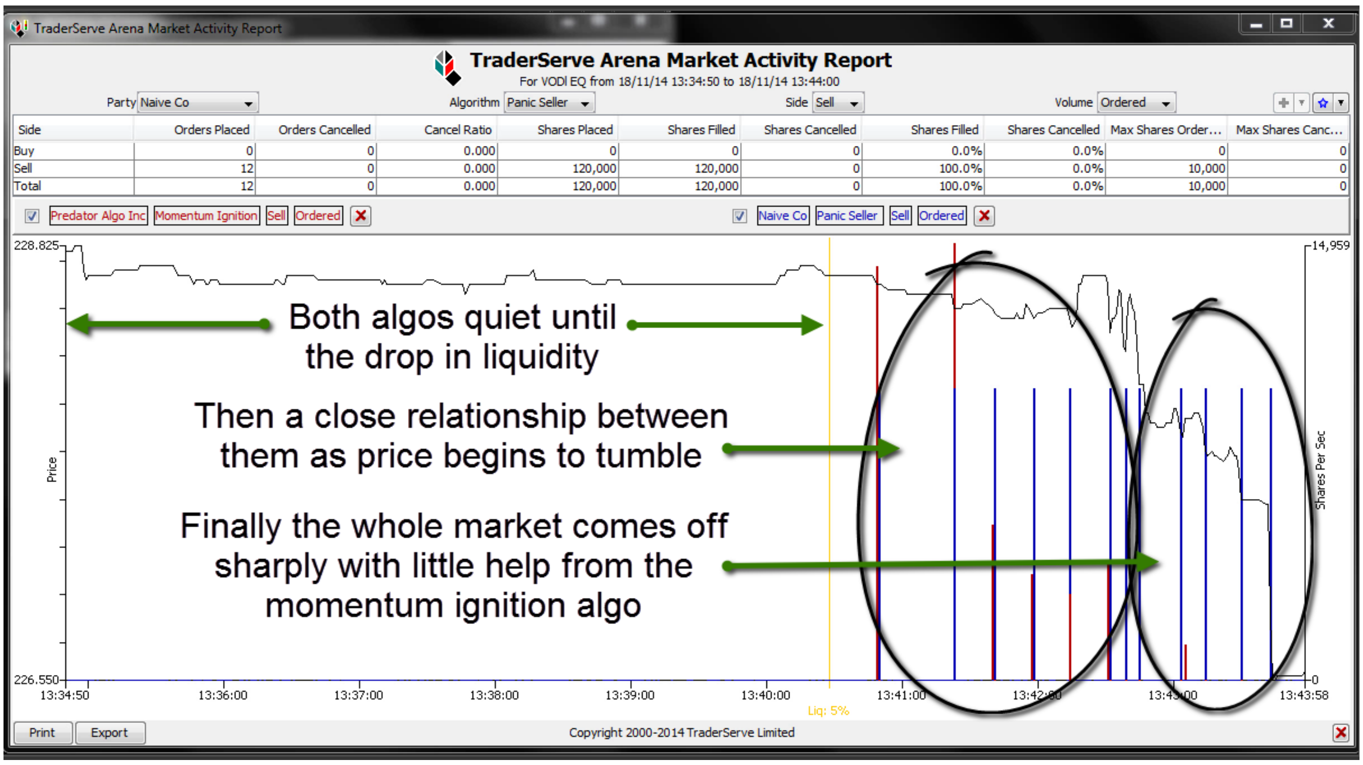
Momentum Ignition

We complete our examination of predatory market behaviour with a “Momentum Ignition” algo. Momentum Ignition is a ploy used by some predatory traders to spark a substantial price movement in conditions of low market activity. This time the idea is to buy or sell through several price levels to hit stops or provoke short term momentum traders to continue the move, then sell or buy back the same amount more cheaply.

We constructed a simple “Momentum Ignition” algo and ran it on the TS-Arena. This sells through three levels when the total volume resting there is abnormally low. It then places a buy order at the lowest of those levels, which then becomes the best bid, with the intention of buying back cheaply from other market participants. To show how dangerous for the market this can be, we also ran another “Naïve Panic Seller” algo, this one sensitive to sharp downward movements in price.

These ran together for some time without serious consequence. Nevertheless the combination is toxic as were able to demonstrate by taking advantage of a useful facility of the TS-Arena to vary the market liquidity. We dropped the liquidity to 5% of the real market. This is extreme but was adopted to induce a swift response.

Fig.7 shows what happened. The Sell orders of the “Momentum Ignition” algo are overlaid by the Sell orders of the “Naïve Panic Seller”. We see that both algos were relatively quiet before the drop in liquidity (represented by the yellow line on the chart). Thereafter the “Momentum Ignition” algo triggered and the “Naïve Panic Seller” started selling heavily. For a while there is a close relationship between the selling activity of the two algos and then the market comes off sharply with little help from the “Momentum Ignition” algo.


TraderServe Arena Market Activity Report

TraderServe Arena Market Activity Report

For VODl EQ from 18/11/14 13:34:50 to 18/11/14 13:44:00

Party: Naive Co | Algorithm: Panic Seller | Side: Sell | Volume: Ordered

Side	Orders Placed	Orders Cancelled	Cancel Ratio	Shares Placed	Shares Filled	Shares Cancelled	Shares Filled	Shares Cancelled	Max Shares Order...	Max Shares Canc...
Buy	0	0	0.000	0	0	0	0.0%	0.0%	0	0
Sell	12	0	0.000	120,000	120,000	0	100.0%	0.0%	10,000	0
Total	12	0	0.000	120,000	120,000	0	100.0%	0.0%	10,000	0

Predator Algo Inc | Momentum Ignition | Sell | Ordered

Naive Co | Panic Seller | Sell | Ordered

228.825
226.550
14,959
0
Price
Shares Per Sec
13:34:50
13:36:00
13:37:00
13:38:00
13:39:00
13:40:00
13:41:00
13:42:00
13:43:00
13:43:58
Liq: 5%
Both algos quiet until the drop in liquidity
Then a close relationship between them as price begins to tumble
Finally the whole market comes off sharply with little help from the momentum ignition algo
Print
Export

Copyright 2000-2014 TraderServe Limited

Fig.7

What this shows is that the presence of a naïve participant can be provoked by a predator into contributing to market disorder.

In our remaining scenarios we dispense with predators and examine the threat to markets posed by apparently innocuous algos on their own if inadequately tested.

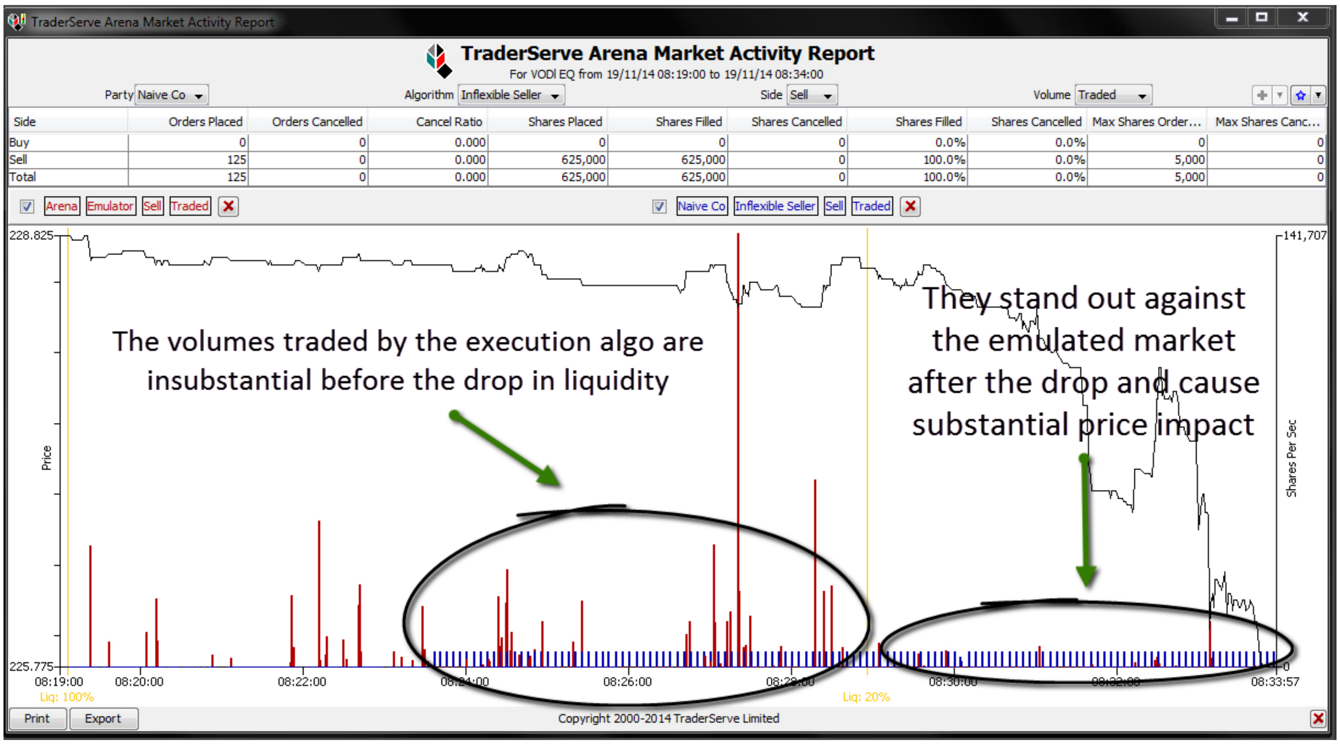
Innocent Algos without Predator Algos

Innocent Algos can <u>cause</u> Market Disorder

It is not just intentionally abusive algos that pose problems to market order. Something as innocent as an execution strategy can create disorder if it lacks sensitivity to market conditions.

To demonstrate this we added a “Naïve Inflexible Seller” algo to the TS-Arena market. This algo periodically sends a market order to sell a fixed amount. After the algo had been running for some time, we again reduced the market liquidity to a fraction of that of the real market.

Fig.8 reports the episode. Looking at trading activity before and after the drop in liquidity on the Emulator (in red) and on the Inflexible Seller (in blue), we see that while the volumes traded by the execution algo are insubstantial before the drop in liquidity, they stand out against the emulated market after the drop and the inflexibility in the order size causes substantial price impact.


TraderServe Arena Market Activity Report

For VODl EQ from 19/11/14 08:19:00 to 19/11/14 08:34:00

Party: Naive Co — Algorithm: Inflexible Seller — Side: Sell — Volume: Traded

Side	Orders Placed	Orders Cancelled	Cancel Ratio	Shares Placed	Shares Filled	Shares Cancelled	Shares Filled	Shares Cancelled	Max Shares Order…	Max Shares Canc…
Buy	0	0	0.000	0	0	0	0.0%	0.0%	0	0
Sell	125	0	0.000	625,000	625,000	0	100.0%	0.0%	5,000	0
Total	125	0	0.000	625,000	625,000	0	100.0%	0.0%	5,000	0

Arena
Emulator
Sell
Traded
Naive Co
Inflexible Seller
Sell
Traded
228.825
225.775
141,707
0
Price
Shares Per Sec
08:19:00
08:20:00
08:22:00
08:24:00
08:26:00
08:28:00
08:30:00
08:32:00
08:33:57
Liq: 100%
Liq: 20%
The volumes traded by the execution algo are insubstantial before the drop in liquidity
They stand out against the emulated market after the drop and cause substantial price impact
Print
Export
Copyright 2000-2014 TraderServe Limited

Fig.8

We conclude that a naïve algo is capable of causing market disorder on its own – in this case by failing to take into account the underlying market liquidity.

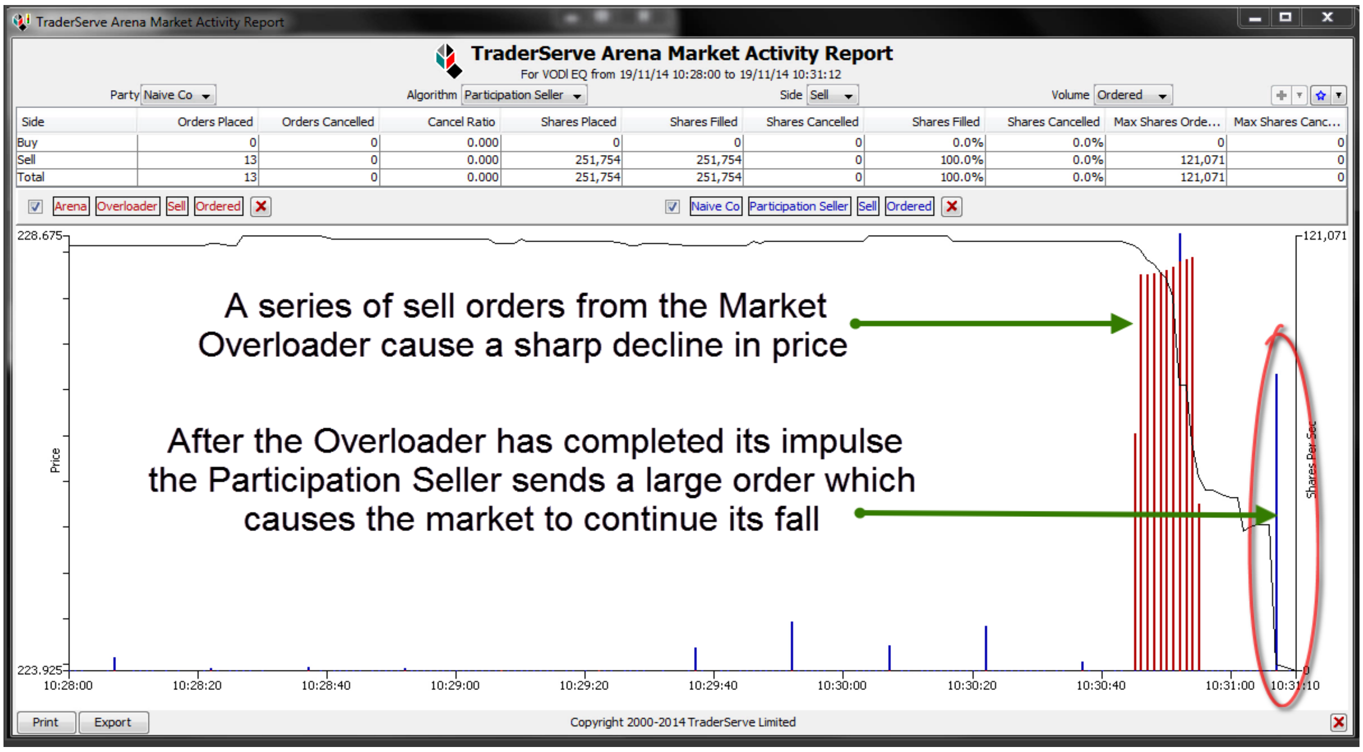
Obviously the algo has been poorly designed: the size of the orders it sends may be fine in normal conditions but far too heavy when liquidity is very low**. If live markets are to be protected from naïve algos like this, there has to be thorough non-live testing including stress testing in conditions of unusually low liquidity.**

Innocent Algos can <u>contribute</u> to Market Disorder

Problems for poorly designed and inadequately tested algos can also emerge around periods of particularly busy activity. We explored this in the TS-Arena with a different execution algo. In place of the “Naïve Inflexible Seller” we ran a “Naïve Participator” algo. This addresses the key problem of the previous algo, its inflexibility to changing patterns of liquidity, by scaling the size of its periodic orders to the volume traded in the last period: in this way the algo attempts to trade a fixed percentage of the actual volume.

Rather than apply stress by adjusting the liquidity, this time we provide a price impact to the market by applying the TS-Arena's "Market Overloader". This sends a succession of orders of specifiable size and frequency to the market over a specified period.   In this case we provided selling pressure over a period of six seconds.

In Fig.9 we compare the sell orders of the "Market Overloader" (in red) to those of the "Naïve Participator" (in blue). After the initial impulse had finished, and the price had already declined steeply, we see that a large sell order came in from the execution algo, and this had the effect of causing the market to fall even further.   The reason for this is clear: since its last order there had been heavy activity in the market from the "Market Overloader", and following its internal logic the "Naïve Participator" had to trade a fixed percentage of this volume, which, in these circumstances, was a very sizeable order.


TraderServe Arena Market Activity Report

TraderServe Arena Market Activity Report
For VODl EQ from 19/11/14 10:28:00 to 19/11/14 10:31:12

Party: Naive Co · Algorithm: Participation Seller · Side: Sell · Volume: Ordered

Side	Orders Placed	Orders Cancelled	Cancel Ratio	Shares Placed	Shares Filled	Shares Cancelled	Shares Filled	Shares Cancelled	Max Shares Orde...	Max Shares Canc...
Buy	0	0	0.000	0	0	0	0.0%	0.0%	0	0
Sell	13	0	0.000	251,754	251,754	0	100.0%	0.0%	121,071	0
Total	13	0	0.000	251,754	251,754	0	100.0%	0.0%	121,071	0

Arena
Overloader
Sell
Ordered
Naive Co
Participation Seller
Sell
Ordered
228.675
223.925
Price
121,071
0
Shares Per Sec
10:28:00
10:28:20
10:28:40
10:29:00
10:29:20
10:29:40
10:30:00
10:30:20
10:30:40
10:31:00
10:31:10
A series of sell orders from the Market Overloader cause a sharp decline in price
After the Overloader has completed its impulse the Participation Seller sends a large order which causes the market to continue its fall
Print
Export
Copyright 2000-2014 TraderServe Limited

Fig.9

In this case the execution algo was not the sole cause of the market disorder but it did contribute to it. Naïve scaling can be just as dangerous as naïve inflexibility in these market conditions, and we conclude **that there is a need for non-live testing to subject algos to overloaded markets as well**.

The demonstration that naïve algorithms either on their own or in the company of predators can contribute to disorderly markets is of the greatest significance not just to regulators but to any user concerned about the fair and orderly functioning of live markets.

Conclusion

Regulators are minded to place an obligation on exchanges to offer non-live testing to participants, and on the participants to make use of that service: the aim is to protect live markets from trading activity likely to produce or contribute to market disorder. In the course of this research we have shown that, while replay servers are not up to the task, there are tools available which meet the primary requirements on non-live testing – namely, that of providing a realistic and responsive test market which allows different algos to interact together within the same order book. **This is crucial to detect emergent behaviours of algo combinations which can imperil fair and orderly trading before they are committed to the live market.**

We have also shown that it is possible to detect classes of abusive behaviour in non-live testing: all these forms of predatory algo have a signature which is identifiable in the market activity report.

The need for such testing is reinforced by the demonstration that it is not just predatory algos that contribute to market disorder. **Apparently innocuous algos such as execution strategies can be just as problematic if they are not adequately tested with the necessary tools. This is particularly true in market conditions such as dramatically reduced liquidity and increased volatility, so it is important that non-live testing confront algos with such extreme conditions in a realistic simulation**. The ability to do stress testing of this kind is therefore a further crucial aspect of non-live testing.