Non-live Testing: Why trading venues need to test for Emergent Market Disorder and for vulnerability to stressed market conditions

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.

To conduct this research we examined the behaviour of algorithms 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 algorithms 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 algorithm, TS-Arena also allows the simulation of extreme conditions of activity. This is important in assessing any hidden predisposition for an algorithm to contribute to a disorderly market.

In production the participant algorithms 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 algorithms with the exception of the Market Emulator view only the TS-Arena data from our matching engine (i.e. the underlying production market is only seen by the Market Emulator algorithm), 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 algorithms in realistic market conditions.

We show that apparently innocent algorithms, if insufficiently tested, may be conducive to the creation of a disorderly market in two different ways: first, in the presence of other algorithms – for example, predators aiming to take unfair advantage of them; and, secondly, in extreme market conditions. We conclude that, to maximise the value of non-live testing for protecting live markets from disorderly trading, the platform must be capable of examining the disposition of algorithms (1) to produce emergent market disorder and (2) to misbehave in stressed market conditions of extreme liquidity or volatility.

Emergent Market Disorder

Disorderly trading conditions can materialise in a number of ways. In this section we focus on one of these: the disorder that can result from the interplay between two or more algorithms each of which is innocuous on its own. We call this "Emergent Market Disorder" and illustrate it with reference to a form of abusive behaviour that has been relatively common: Momentum Ignition.

Momentum Ignition is a ploy used by some predatory traders to spark a substantial price movement in conditions of low market activity. 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. This is one of a number of predatory algorithms whose impact on market order we investigated in our research note "Trading Algorithms, Disorderly Markets and Non-Live Testing: A study of emergent behaviours supporting the case for non-live testing regulations" (available at www.traderserve.com/reports.php) which is an extended version of the current paper.

We constructed a simple "Momentum Ignition" algorithm 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" algorithm, 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.

The market activity report in Fig.1 shows what happened. The volumes of the Sell orders of the "Momentum Ignition" algorithm in red are overlaid by the volumes of the Sell orders of the "Naïve Panic Seller" in blue. Both are viewed over a price chart showing the movement of the mid during the same interval. We see that both algorithms were relatively quiet before the drop in liquidity (represented by the yellow vertical on the chart). Thereafter the "Momentum Ignition" algorithm triggered and the "Naïve Panic Seller" started selling heavily. For a while there is a close relationship between the selling activity of the two algorithms and then the market comes off sharply with little help from the "Momentum Ignition" algorithm as can be seen from the relative height of the red verticals (Momentum Ignition Sell Orders) and blue verticals (Naive Panic Seller Sell Orders).

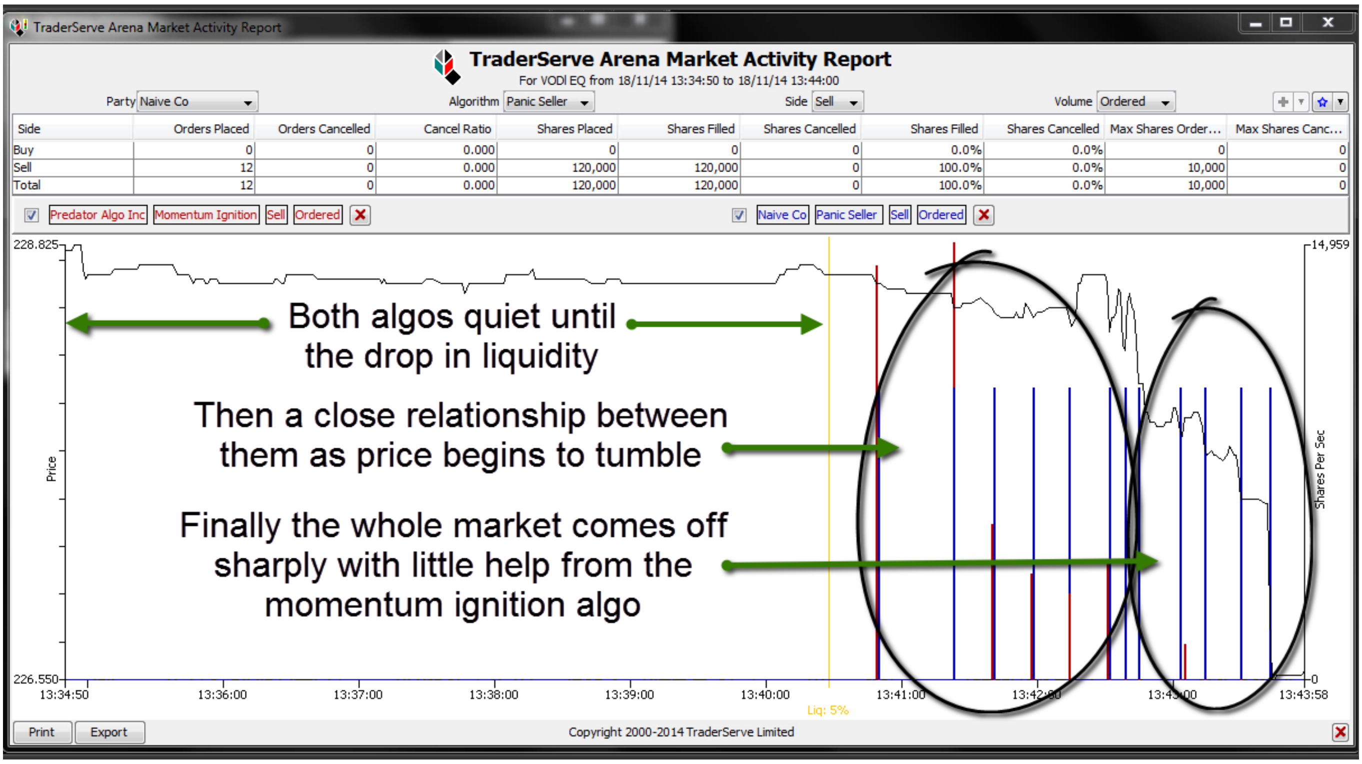

Fig.1

Fairly substantial market disorder results from the interaction between the two algorithms, and though this would not have occurred without the presence of the Momentum Ignition algorithm, it is the naïve algorithm rather than the predator that does most of the damage. This Emergent Market Disorder would not materialise in any non-live test of the "Naïve Panic Seller" that depended on historical replay. It is essential that the orders of the participant algorithms be placed in the same order book so that this synergy can occur, and non-live testing will not be effective in protecting live markets from disorder of this kind if it doesn't allow for such interactions.

Stressed Market Conditions

Problems for poorly designed and inadequately tested algorithms can also materialise in periods of extreme market activity. We explored this in TS-Arena with a simple execution algorithm, a "Naïve Participator". This responds 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 algorithm attempts to trade a fixed percentage of the actual volume.

We provided an impact to the market by applying the TS-Arena's "Market Overloader" which 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.2 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 algorithm, 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.

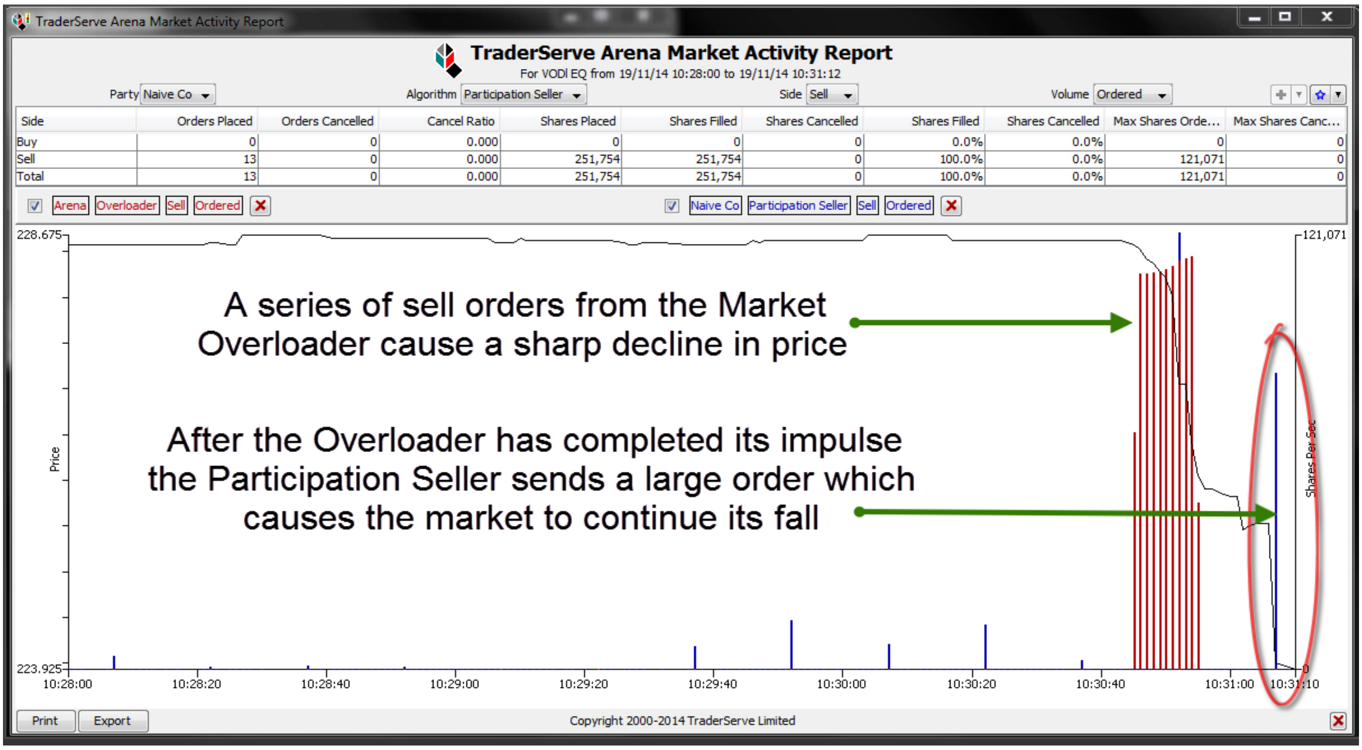

Fig.2

Here the execution algorithm was not the sole cause of the market disorder but it did contribute to it, and in so doing exhibited a weakness in its design that had not been apparent before. In the longer research note referred to above we show the production of market disorder by a similar execution algorithm in conditions of severely reduced liquidity. These illustrations show the importance of including stress tests in the non-live testing regime.

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 algorithms to interact together within the same order book. **This is crucial to detect emergent behaviours of algorithm combinations which can imperil fair and orderly trading before they are committed to the live market.**

The need for such testing is reinforced by the demonstration that it is not just predatory algorithms that contribute to market disorder. **Apparently innocuous algorithms 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 algorithms 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

