Adaptive Trade Execution using a Grammatical Evolution Approach

Wei Cui

Natural Computing Research & Applications Group,
School of Business,
University College Dublin, Ireland,
E-mail: wei.cui.1@ucdconnect.ie

Anthony Brabazon

Natural Computing Research & Applications Group,
School of Business,
University College Dublin, Ireland,
E-mail: anthony.brabazon@ucd.ie

Michael O'Neill

Natural Computing Research & Applications Group,
School of Computer Science and Informatics,
University College Dublin, Ireland,
E-mail: m.oneill@ucd.ie

Abstract:

Trade execution is concerned with the actual mechanics of buying or selling the desired amount of a financial instrument. Investors wishing to execute large orders face a tradeoff between market impact and opportunity cost. Trade execution strategies are designed to balance out these costs, thereby minimising total trading cost. Despite the importance of optimising the trade execution process, this is difficult to do in practice due to the dynamic nature of markets and due to our imperfect understanding of them. In this paper we adopt a novel approach, combining an evolutionary methodology whereby we evolve high-quality trade execution strategies, with an Agent-based Artificial Stock Market, wherein the evolved strategies are tested. The evolved strategies are found to outperform a series of benchmark strategies and several avenues are suggested for future work.

Keywords: Algorithmic Trading, Trade Execution, Artificial Stock Market, Evolutionary Computation

Reference

1 Introduction

Algorithmic trading (AT) can be broadly defined as the use of computers to automate aspects of the investment process. Hence, AT can encompass the automation of decisions ranging from stock selection for investment, to the management of the actual purchase or sale of that stock. A significant proportion of all financial asset trading is now undertaken by AT systems with this form of trading accounting for approximately 20-25% of total US market volume in 2005. Boston-based research firm Aite Group predicts that AT will account for more than half of all shares traded in the U.S. by the end of 2010 (Kim, 2007). AT is also common in European financial markets with approximately 50% of trading volumes being accounted for by algorithmic trading programs (Engle et al., 2008). Significant volumes in Asian markets are similarly traded (Decovny, 2008). Algorithmic trading is seen in multiple financial markets ranging from equities to FX (foreign exchange), to derivative (futures, options etc.) markets. In this paper we restrict attention to one aspect of financial trading to which AT can be applied, namely efficient trade execution.

A practical issue that arises for investors is how they can buy or sell large quantities of a share (or some other financial asset) as efficiently as possible in order to minimise market impact and information leakage. The obvious strategy to minimise market impact is to break up the order up into smaller lots and spread it over several purchases. While this will reduce the market impact, it incurs the risk of suffering opportunity cost, that market prices may start moving against the trader during the multiple purchases. This produces what is referred to as the *trader's dilemma* (Kissell & Glantz, 2003).

Added to this, the steady flow of small orders over time may inform other market participants of the presence of a large order and so encourage them to 'run ahead' of the investor. Hence, the design of trade execution strategies is intended to balance out the total cost of market impact and opportunity cost while maintaining a tight control over information leakage. In selecting a trade execution strategy, the investor must not only balance her preferences but must also be prepared to adapt and change strategy as market conditions evolve with the aim of selecting best trading tactic under current market conditions.

The task in devising an efficient execution strategy is complex as it entails multiple sub-decisions including how best to split up the large order, what style to adopt in executing each element of the order (aggressive or passive), how execution performance is to be measured, and what type of order to use. When an asset is traded simultaneously on multiple markets a further decision must be made as to how to split up the order amongst these markets. In addition, the electronic order books faced by the investor are constantly changing. In the past the task of designing an execution strategy was typically undertaken by human experts but this is infeasible as the number of sub-decisions increases, suggesting that an automated approach is required. Critically, any approach must be dynamic and responsive to underlying market conditions in real-time.

In this study we adopt an evolutionary methodology, called Grammatical Evolution (GE) (O'Neill & Ryan, 2003), to 'evolve' a high-quality trade execution strategy. GE is an Evolutionary Automatic Programming (EAP) technique which allows the generation of computer programs (or 'rule sets') in an arbitrary language. GE can conduct an efficient exploration of a search space, and notably permits the incorporation of existing domain knowledge in order to generate ‘solutions’ with a desired structure. In finance (for example), this allows the users to seed the evolutionary process with their current trading strategies in order to see what improvements the evolutionary process can uncover. Recently GE has been successfully applied to a number of financial problems. These include financial time series modelling, intraday financial asset trading, corporate credit rating, and the uncovering of technical trading rules (Brabazon & O’Neill, 2006).

To test the performance of the evolved trade execution strategies, we employ an artificial stock market (ASM). A critical aspect of real-world design and testing of execution strategies, is that they cannot be easily backtested using historical data as it is very difficult to assess the impact that an execution strategy would have produced *ex ante*. Apart from this issue, another practical problem is that historical order book information only represents a single sample path through time and hence, using this information to estimate the likely future utility of any specific execution strategy is problematic. By implementing an artificial stock market environment, it is possible to create a closed world which allows the testing of new execution strategies over multiple runs, potentially allowing us to develop robust execution strategies.

1.1 Structure

This paper is organised as follows. The next section provides some background on the typical operation of an electronic double auction marketplace; Section 3 discusses trade execution strategies; Section 4 describes the evolutionary approach adopted in this study, namely Grammatical Evolution; our experimental approach, including the artificial stock market model used to test the evolved execution strategies is outlined in Section 5; Section 6 provides our results, with conclusion and some suggestions for future work being presented in the final section of this paper.

2 Background

In this section we provide an introduction to the market structure most commonly found in large international equity markets, and we also discuss some relevant concepts from the market microstructure literature.

2.1 Market Mechanism

Today most equity markets operate an *electronic double auction limit order book*. Examples include the Electronic Communication Networks (ECNs) in the United States, the Toronto Stock Exchange, and the Hong Kong Stock Exchange. Electronic trading platforms in derivative markets have also gained popularity in recent years over the traditional open-outcry auctions, such as Chicago Mercantile Exchange’s (CME) Globex platform and International Securities Exchange’s electronic option trading platform. One advantage of an open limit-order book is the greater transparency offered by these systems when compared with dealer market settings. Price quotes and transactions are visible to all participants which generally improves the efficiency of price discovery, thus promoting market confidence. It also promotes competition as dealers/market makers are encouraged to post the best prices to attract order flow (CFA, 2009).

In a limit order market, traders can either submit *a limit order* or *a market order*. A market order is an order to buy or to sell a specified number of shares. It guarantees immediate execution but provides no control over its execution price. In contrast, a limit order is an order to buy or to sell a specified number of shares at a specified price. It provides control over its execution price but does not guarantee its execution.

Table 1 Order Book 1

Bid		**Ask**	
Shares	Prices	Prices	Shares
300	50.19	50.22	200
200	50.18	50.23	300
400	50.17	50.24	100
500	50.16	50.25	300
300	50.15	50.26	200
100	50.14	50.27	400

Table 2 Order Book 2

Bid		**Ask**	
Shares	Prices	Prices	Shares
300	50.19	50.22	**200**
500	50.18	50.23	300
400	50.17	50.24	100
500	50.16	50.25	300
300	50.15	50.26	200
100	50.14	50.27	400

Table 3 Order Book 3

Bid		**Ask**	
Shares	Prices	Prices	Shares
300	50.19	50.22	**100**
500	50.18	50.23	300
400	50.17	50.24	100
500	50.16	50.25	300
300	50.15	50.26	200
100	50.14	50.27	400

Table 4 Order Book 4

Bid		**Ask**	
Shares	Prices	Prices	Shares
300	50.19	**50.23**	**100**
500	50.18	50.24	100
400	50.17	50.25	300
500	50.16	50.26	200
300	50.15	50.27	400
100	50.14	50.28	300

Table 1 shows a sample order book, where all the buy and sell orders are visible/transparent to traders in the market. It consists of two queues which store buy and sell limit orders, respectively. Buy limit orders are called *bids*, and sell limit orders are called *offers* or *asks*. The highest bid price on the order book is called *best bid*, and the lowest ask price on the order book is called *best ask*. The difference between best bid and best ask is called *bid-ask spread*. Prices on the order book are not continuous, but rather change in discrete quanta called *ticks*.

Beginning in 2001, the US equity markets changed their minimum tick size from one sixteenth of a dollar to one cent. Since decimalisation, the average trade size has declined from 1,200 shares per transaction in 2000 to 300 shares today. This in turn has led to an explosion in the number of trades executed and a narrowing of spreads with large institutional orders taking longer to execute. Due to these changes, Wall Street firms (both buy-side and sell-side) have started to embrace AT for trade execution over the last few years (Kissell, 2006). The emergence of AT has resulted in a substantial increase in the speed of trade execution and a significant reduction in the average value of each trade (ASX, 2010).

In a limit order market, orders arrive randomly in time. The price limit of a newly arrived order is compared to those of orders already held in the system to ascertain if there is a match. If so, the trade occurs at the price set by the first order. The set of unexecuted limit orders held by the system constitutes the dynamic order book, where limit orders can be cancelled or modified at any time prior to their execution. Limit orders on the order book are typically (depending on market rules) executed strictly according to (1) price priority and (2) time priority. Bid (ask) orders with higher (lower) prices get executed first with time of placement being used to break ties. A buy (sell) market order is executed at the best ask (bid) price. The limit order book is highly dynamic, because new limit orders will be added into the order book, and current limit orders will get executed or cancelled from the order book throughout the trading day. Table 2 shows the order book after a trader submits a buy limit order with 300 shares placed at price 50.18. Table 3 shows the order book after a trader submits a buy market order with 100 shares. Table 4 shows the order book after a trader submits a buy market order with 300 shares.

Apart from market and limit orders, some stock exchanges also offer *hidden/iceberg orders* to allow traders to conceal the total size of a large limit order. Such orders consist of two components, a small component whose size is visible in the order book and a larger hidden component with a size known only to the order submitter. The hidden component is exposed to the market gradually through execution of the visible part of the order (Aitken et al., 2001; Biais et al., 1995). Many electronic trading platforms have introduced this kind of order, including Euronext, the Toronto Stock Exchange, the London Stock Exchange, and XETRA. Hidden limit orders are often used by large liquidity traders to hide their intent to trade (Bongiovanni et al., 2006). However, iceberg orders exhibit a less favorable time priority compared with pure limit orders (Bessembinder et al., 2009; Esser & Monch, 2007). After the visible portion of an iceberg order is completely matched, other visible limit orders at the same limit price that were entered before the new portion is displayed take priority.

An alternative form of market is provided by a *crossing network* (CN). This network may be operated internally by a large financial trading house, or by an independent entity. In a crossing network, buyers and sellers are electronically paired for an agreed-on quantity, and the trade is priced by reference to a price determined in another market. Unlike visible order books, the level of liquidity is unknown *a priori* in a crossing network which is a *dark pool.* The investor's trade is not revealed until after the order has been filled. One advantage of trading in a dark pool is that the trade will usually take place within the bid-ask spread thereby providing a better price than might be available elsewhere. Another advantage of trading large orders in a crossing network is that market impact costs are minimized since investors avoid adverse price movements which would otherwise occur upon the revelation of such large orders in the market (CFA, 2009). A disadvantage of submitting a trade to a crossing network is that the investor does not usually know the level of liquidity available in the network for the financial asset of interest, hence it is possible that a trade will not be fully executed (Gresse, 2006). The main CNs currently operating in North America are Reuters' Instinet Crossing Network, Pipeline Trading System, Liquidnet, ITG's POSIT (Portfolio System for Institutional Traders) and the New York Stock Exchange's after-hours Crossing Network.

As technology has developed, investors are able to implement more sophisticated execution strategies whereby orders can be routed to multiple markets simultaneously, and then adapted on the fly.

2.2 Trading Cost

Trading cost is one of the two important factors which reflect investment performance while the other is the underlying investment strategy of the portfolio manager (Borkovec et al., 2009; Keim & Madhavan, 1995; Keim, 1999). Execution costs can have a significant impact on investment performance. For example, Perold (1988) observed that a hypothetical or 'paper' portfolio constructed according to the Value Line rankings outperforms the market almost 20% per year during the period from 1965 to 1986, whereas the actual portfolio (the Value Line Fund) outperformed the market by only 2.5% per year, the difference arising from execution costs. 40% of market participants believe that alpha is lost primarily through trading costs, while 14% of respondents attribute the loss of alpha to bad timing of their transactions (Sussman, 2008). Today, 98% of large institutions and 88% of medium-sized institutions globally are consumers of transaction cost research (Borkovec et al., 2009).

Consideration of trading costs is vital in determining a quality trade execution strategy (Kissell & Glantz, 2003). If trading costs are not properly managed they could cause a superior opportunity to become only marginally profitable or a normally profitable opportunity to turn into a loss (Keim & Madhavan, 1995; Kissell, 2006).

Trading costs can be decomposed into direct costs and indirect costs. Direct costs are observable and include commissions, fees and taxes. Indirect costs are typically more difficult to estimate and can be divided into three main components:

- market impact cost,
- opportunity cost, and
- bid-ask spread.

Execution needs to balance all of these components (Almgren, 2008). There is ample empirical evidence dating from Kraus & Stoll (1972) to Chan & Lakonishok (1995) and Chan & Lakonishok (1997), indicating that price impact is the dominant component of transaction costs. As markets become increasingly automated, fixed costs of trading are driven down, and this effect is more pronounced. Thus, there is no loss in generality in equating transaction costs with market impact cost (Hora, 2006).

Market impact occurs when the actions of an investor start to move the price adversely against themselves. Hence, market impact is the difference between a transaction price and what the market price would have been in the absence of the transaction. For example, an order may be executed as quickly as possible through sweeping all orders posted to the limit-order book, however this would incur significant cost and drive the price of the security against the investor. In this case the investor avoids market risk but, by demanding instantaneous liquidity, incurs significant market impact cost. Market impact can be divided into permanent impact cost caused by information leakage and instantaneous impact cost caused by liquidity demand, where the latter is typically about 95% of entire market impact cost (Kim, 2007).

Opportunity cost arises due to two primary reasons (Keim & Madhavan, 1998). One reason is that an order is only partially filled or in a more extreme case, is not executed at all. This is most likely to occur when using passive trading strategies, such as limit order strategies.

There are many factors which influence trading costs. It is less expensive to trade a small volume of stock than a larger quantity (Yang & Jiu, 2006). Trading costs are also driven by the desired trading time frame (it costs more to transact over shorter time intervals as the investor has to pay for liquidity). Trading costs vary during the day as the amount of market liquidity typically varies between mid-session and the open/close. Buying/selling high capitalisation stocks typically yields lower price impact cost for a given order size due to the liquidity available in a high cap stock (Lim & Coggins, 2005a). Price impact costs typically increase as order imbalance increases (Lim & Coggins, 2005a). Furthermore, research has shown that important determinants of the price effect are investment style, trade type (agency, single, or principal), market momentum, stock price volatility, and trading venue (Keim & Madhavan, 1998). Execution needs to balance all of these factors and the resulting strategy should ideally be dynamically adaptive depending on observed liquidity and other market data (Almgren, 2008).

2.3 Market Impact Asymmetry

Many empirical studies document asymmetries in the trading costs (market impact costs) of large order purchases and sales. Kraus & Stoll (1972), Holthausen et al. (1987, 1990), Chan & Lakonishok (1993, 1995), Gemmill (1996), Keim & Madhavan (1995, 1996, 1997), Saar (2001), Bikker et al. (2007) find that buys have larger price impact than sells. Not only is the price impact larger for buys, but the subsequent reversal is also much smaller than for sells. While prices go up on buys and down on sells, they revert after sells but remain high after buys, creating a permanent price impact asymmetry. Thus there is an asymmetry in the overall impact of buys and sells.

Several explanations appear in the literature to account for this phenomenon. Chan & Lakonishok (1993) and Keim & Madhavan (1996) argue that sells are more often liquidity-motivated rather than information-based than buys. Buys create new long-term positions and thus imply a preference to hold a particular stock. Saar (2001) provides a different explanation for the price impact asymmetry. He builds a theoretical model which demonstrates how the price impact asymmetry can arise. The main implication of the model is that the history of price performance of a stock affects the degree of asymmetry: the longer the run of price appreciations, the less positive the difference in permanent price impact between buys and sells. When the price run-up is long enough, sells may even have higher price impact than buys. Another explanation for the price impact asymmetry is given by Chiyachantana et al. (2004). They find that the asymmetry depends on particular market conditions: price effects of buyer-initiated trades are greater in bull markets (as in 1997-1998) whereas those of seller-initiated trades are larger in bear markets (as in 2001). Boscuk & Lasfer (2005) take a different view and show that the type of investor and the combination of the size of the trade and the trader's resulting level of ownership are the major determinants of price impact asymmetry at the London Stock Exchange.

3 Trade Execution Strategy

A dilemma facing traders is that trading aggressively is associated with high cost and trading passively is associated with high risk. Traders can not reduce one component without adversely affecting the other. Hence, these factors are balanced in trade execution strategies.

Trade execution strategies are predefined sets of instructions for trade execution, which include how best to break up a large order for execution and the actual mechanisms for placing and managing the orders. When deciding whether or not to submit an order to the market place an algorithm must decide on an order's:

- Submission Time - when should the order be placed?
- Size - what size order should be sent to the market?
- Type - should the order be a market, limit or a hidden order?
- Pricing - at what price should the order be placed?
- Destination - there are many market destinations and types, which one will provide the best conditions of execution for the order?
- Management - if a limit order has been submitted, how should this order be managed post submission?

The total trading volume of the order to be traded is often expressed as a percentage of the *average daily volume* (ADV) of the stock (Kissell & Glantz, 2003). An order of less than 5% of ADV can generally be traded over a day without much price impact. On the contrary, if the target volume is larger than 25% of ADV, it may require execution over several days in order to minimise market impact. Orders between 5 and 15% of ADV can be traded during the course of a trading day but with some work; orders between 15 and 25% of ADV are difficult trades to execute which require traders to minimize information leakage pertaining to their trading intentions (Kissell & Glantz, 2003). Cai & Sofianos (2006) also suggest that the limit of one day trading typically ranges from 10 to 30% of ADV.

We assume that the order to be traded consists of V shares. The order is usually sliced into N smaller or *child* orders (each of which will be submitted to the market according to our trading strategy), with an order size of s_1, s_2, $\ldots$, s_N respectively, where

$$V = \sum_{i=1}^{N} s_i \quad .$$

Commonly these child orders are submitted at regular intervals. Intervals of half an hour and fifteen minutes are usually adopted in practice. The sizes of these child orders (s_1, s_2, ..., s_N) can be (conceptually) determined using the following formula (Kissell & Glantz, 2003)

$$Min : Cost(x) + \lambda * Risk(x) \quad ,$$

which seeks to balance the trading cost and associated risk. In the formula, $Cost(x)$ is the forecasted trading cost estimate, $Risk(x)$ is timing risk which represents the associated uncertainty surrounding the forecasted trading cost estimates due to price volatility and liquidity risk, λ is the risk aversion factor depicting an investor's desired tradeoff between cost and risk, and x is the corresponding order schedule of the large order represented by a size vector in the form

$$x = (s_1, s_2, \ldots, s_N) \quad .$$

The *trade schedule problem* has been investigated by many researchers including Bertsimas & Lo (1998), Almgren & Chriss (1999), Almgren & Chriss (2000), Almgren (2003), Coggins et al. (2003), Obizhaeva & Wang (2005), Almgren & Lorenz (2007), and Alfonsi et al. (2010).

The actual implementation of the child orders consists of several decisions. For example, if a market order is being placed, the trader must decide which market to use (as most large companies will trade in multiple locations). If a limit order is being placed, additional considerations are, what limit price to set and when (and how) this price will be adjusted if the order does not execute within a certain time frame.

An important characteristic of an order is its level of *aggressiveness*. At the most aggressive end of the spectrum are market orders; at the least aggressive end of the spectrum are limit orders with prices that are far away from the current market price. Empirically, traders determine their order aggressiveness depending on their trading preferences and on available market information when submitting their orders (Ng, 2010).

3.1 Order Book Information Content

There are a large number of studies analysing the relationship between order aggressiveness and the information content of limit order books. It has been reported that limit orders submitted at the best quote or inside the spread have lower trading costs than market orders (Harris & Hasbrouck, 1996). Similarly, Griffiths, Turnbull and White (2000) recommend placing buy (sell) limit orders at the best bid (ask) as an optimal strategy for minimising the implementation shortfall. As far as the probability of a limit order execution is concerned there is evidence that it is primarily determined by the distance of the limit price from the best quote. Orders closer to the best quote have a higher execution probability and a shorter time to execution (Cho & Nelling, 2000; Lo et al., 2002). Lo & Sapp

(2010) find that traders are hesitant to submit aggressive orders. They are less likely to submit market orders, than limit orders at the best price and limit orders improving the best price.

Table 5 Definitions of Market Variables

Variables	**Definitions**
BidDepth	Number of shares at the best bid
AskDepth	Number of shares at the best ask
RelativeDepth	Total number of shares at the best five ask prices divided by total number of shares at the best five bid and ask prices
Spread	Difference between the best bid price and best ask price
Volatility	Standard deviation of the most recent 20 mid-quotes
PriceChange	Number of positive price changes within the past ten minutes divided by the total number of quotes submitted within the past ten minutes

It has been found that if one side of the order book is dominant, where the dominant side is the one with more depth, then there is an imbalance between supply and demand, and limit orders on the dominant side take longer to execute (Al-Suhaibani & Kryzanowski, 2000) and have a higher risk of an adverse price movement leading to non-execution. Consequently, traders on the same side of the market as the dominant side of the book are more likely to submit market orders to achieve an immediate execution (Cao et al., 2009; Griffiths et al., 2000; Omura et al., 2000; Ranaldo, 2004; Verhoeven et al., 2004).

Traders are more willing to place market orders when the market depth on the same side of the order book is large. If the market depth on the opposite side is larger, traders prefer to submit limit orders (Cao et al., 2008; Duong et al., 2009; Ranaldo, 2004; Xu, 2009). When the bid-ask spread widens, traders prefer to submit limit orders in order to avoid large bid-ask spread cost (Biais et al., 1995; Cao et al., 2008; Duong et al., 2009; Pascual & Verdas, 2009; Ranaldo, 2004; Verhoeven et al., 2004; Xu, 2009). Prior research is inconclusive on the effect of market volatility on order aggressiveness. Pascual & Verdas (2009) show that higher historic volatility suggests limit order submission in mid cap stocks, but the opposite phenomenon is observed in large cap stocks. It has also been reported that limit orders are submitted more frequently when price volatility is high (Chung et al., 1999). Hall & Hautsch (2006) observe an increase of all kinds of order submission during periods of high volatility. Ranaldo (2004) supports an inverse relation between order aggression and volatility, while Lo and Sapp (2010) report a positive relationship between order aggression and volatility. Cao et al. (2008) find that volatility has a minimal effect on order aggression. Verhoeven et al. (2004) argues that greater price volatility implies that a trader has a greater chance of executing his order at a better price. The inconsistency concerning the effects of market volatility in previous studies can be partially explained by the notable changes of market structure in recent years.

From the above, it can be seen that prior literature suggests a range of possible explanatory variables, but indicates that we have an incomplete theoretical understanding of how these factors interact. This suggests that there will be particular utility for the application of evolutionary methods to uncover a suitable model structure (trade execution strategy). Other explanatory variables considered in previous literature include time of day effects and waiting time of trading process. The six information indicators (Table 5), *BidDepth*, *AskDepth*, *RelativeDepth*, *Spread*, $Volatility$ and *PriceChange*, are the explanatory variables mostly employed in recent literature of market microstructure. Thus, we use these six information indicators to construct a dynamic trade execution strategy

3.2 Performance Evaluation

The standard industry metric for measuring trade execution performance is the *VWAP measure*, short for *Volume Weighted Average Price*. The VWAP price as a quality of execution measurement was first developed by Berkowitz et al. (1988). They argue that 'a market impact measurement system requires a benchmark price that is an un-biased estimate of prices that could be achieved in any relevant trading period by any randomly selected trader' and then define VWAP as an appropriate benchmark that satisfies this criteria.

The VWAP is calculated as the ratio of the value traded and the volume traded within a specified time horizon (Berkowitz et al., 1988)

$$VWAP = \frac{\sum(Volume * Price)}{\sum(Volume)}$$

where $Volume$ represents the number of shares in each trade and $Price$ represents its corresponding traded price. An example is shown in Table 6.

Table 6 VWAP Calculation of A Sample Buy Strategy

	Submission Time	Shares		Traded Price		Value
Child Order 1:	t_0	400	$*$	50.15	=	20,060
		600	$*$	50.16	=	30,096
Child Order 2:	$t_1(t_0+\Delta t)$	1,000	$*$	50.40	=	50,400
Child Order 3:	$t_2(t_0+2\Delta t)$	200	$*$	50.34	=	10,068
		800	$*$	50.36	=	40,288
Child Order 4:	$t_3(t_0+3\Delta t)$	1,000	$*$	50.39	=	50,390
Child Order 5:	$t_4(t_0+4\Delta t)$	1,000	$*$	50.68	=	50,680
Child Order 6:	$t_5(t_0+5\Delta t)$	1,000	$*$	51.10	=	51,100
Child Order 7:	$t_6(t_0+6\Delta t)$	1,000	$*$	50.87	=	50,870
Child Order 8:	$t_7(t_0+7\Delta t)$	700	$*$	50.98	=	35,686
		300	$*$	51.00	=	15,300
Child Order 9:	$t_8(t_0+8\Delta t)$	1,000	$*$	50.39	=	50,390
Child Order 10:	$t_9(t_0+9\Delta t)$	1,000	$*$	50.26	=	50,260
Total:		10,000				505,588
VWAP $= 505,588/10,000 = 50.5588$						

In order to evaluate the performance of a trade execution strategy, its VWAP is compared against the VWAP of the overall market. The rationale here is that performance of a trade execution strategy is considered good if the VWAP of the strategy is more favorable than the VWAP of the market within the trading period and poor if the VWAP of the strategy is less favorable than the VWAP of the market within the trading period. For example, if the VWAP of a buy strategy ($VWAP_{strategy}$) is lower than the market VWAP ($VWAP_{market}$), it is considered a good trade execution strategy. Conversely, if the $VWAP_{strategy}$ is higher than the $VWAP_{market}$, it is considered a poor trade execution strategy. Although this is a simple metric, it largely filters out the effects of volatility, which composes market impact and price momentum during the trading period (Almgren, 2008). The performance evaluation functions are as follows (which were used by Lim & Coggins (2005b)):

$$VWAP\ Ratio = \begin{cases} \frac{10^4 * (VWAP_{strategy} - VWAP_{market})}{VWAP_{market}} & (BuyStrategy) \\ \\ \frac{10^4 * (VWAP_{market} - VWAP_{strategy})}{VWAP_{market}} & (SellStrategy) \end{cases}$$

where $VWAP_{market}$ is the average execution price which takes into account all the trades over the day excluding the strategy's trades. This corrects for bias, especially if the order is a large fraction of the daily volume (Lim & Coggins, 2005b). For both buy and sell strategies, the smaller the VWAP Ratio, the better the strategy is.

3.3 Prior Evolutionary Approaches for Trade Execution

Despite the importance of optimising trade execution, there has been relatively little attention paid in the literature to the application of evolutionary methods for this task. One notable exception is Lim and Coggins (2005b) who applied a Evolutionary Algorithm (discussed in the next section) to evolve a trade execution strategy using order book data from a fully electronic limit order market, the Australian Stock Exchange (ASX). In their study, a large order is to be completed within one trading day. The order is divided into ten child orders which are submitted to the market at regular intervals of half an hour. The relative sizes of these child orders are determined according to share volume trading patterns, which typically follow a U-shaped pattern with increased volumes trading at the open and close. The child orders are placed into the market as limit orders at the best available price and the Evolutionary Algorithm is used to find the optimal lifetime that a limit order would remain on the order book (if it had not already been executed) before it was automatically ticked over the spread to close out the trade. The fitness function was the VWAP performance of that strategy relative to the benchmark daily VWAP. Each strategy was trained on three months' worth of transaction-level data using a market simulator. The results were tested out of sample on three highly liquid stocks and tested separately for sell side and buy side. The in sample and out of sample performances were better than those of the two benchmark strategies.

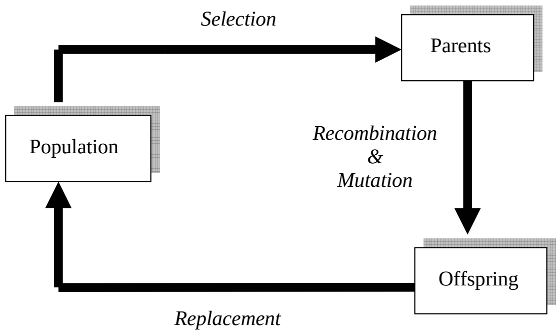
While the above study represents an interesting application of an evolutionary methodology for trade execution, it should be noted that the process was employed as an optimisation heuristic within a fixed, static, trade execution strategy.

However, evolutionary methodologies are also capable of uncovering/discovering entire structures, such in this case as an entire trade execution strategy. This is the approach that is adopted in this paper, which novelly applies a variant of genetic programming called Grammatical Evolution (GE) in order to uncover a dynamic execution strategy.

4 Evolutionary Approach

Evolutionary processes represent an archetype, whose application transcends their biological root. In biological evolution, species are positively or negatively selected depending on their relative success in surviving and reproducing in the environment. Differential survival, and variety generation during reproduction, provide the engine for evolution (Darwin, 1859; Spencer, 1864) (Figure 1).

Figure 1 Evolutionary cycle



These concepts have metaphorically inspired the field of evolutionary computation (EC). Algorithm 1 outlines the evolutionary meta-algorithm. There are many ways of operationalising each of the steps in this meta-algorithm, consequently, there are many different, but related, evolutionary algorithms (EA). Just as in biological evolution, the selection step is a pivotal driver of the algorithm's workings. The selection step is biased in order to preferentially select better (or 'more fit') members of the current population. The generation of new individuals creates children which bear some similarity to their parents but are not identical to them. Hence, each individual represents a trial solution in the environment, with better individuals having increased chance of influencing the composition of individuals in future generations. This process can be considered as a 'search' process, where the objective is to continually improve the quality of individuals in the population. In financial markets, EC methodologies have been used for solving a broad selection of problems, ranging from prediction, asset selection, portfolio optimization, derivatives pricing and credit risk assessment. An overview of some EC applications in finance can be seen in Brabazon et al. (2008).

Initialise the population of candidate solutions;
repeat
  Select individuals (parents) for breeding from the current population;
  Generate new individuals (children) from these parents;
  Replace some / all of the current population with the newly-generated individuals;
until *until terminating condition*;

Algorithm 1: Evolutionary Algorithm

4.1 Grammatical Evolution

Grammatical Evolution (GE) (O'Neill & Ryan, 2003) is an evolutionary methodology, and can be used to evolve structures or 'rule sets'. These rule sets can be as general as a functional expression which produces a good mapping between a series of known input-output data vectors. A particular strength of the methodology is that the form of the model need not be specified *a priori* by the modeler. This is of particular utility in cases, such as in this study, where we have a theoretical or intuitive idea of the nature of the relevant explanatory variables, but a weak understanding of the functional relationship between the explanatory and the dependent variable(s). GE does not require that the model form is linear, nor does the method require that the measure of model error used in model construction is a continuous or differentiable function. Another useful feature of a GE approach is that it produces human-readable rules that have the potential to enhance understanding of the problem domain.

4.1.1 Genotype-phenotype Mapping

A genotype-phenotype mapping is employed such that each individual's variable length binary string, contains in its codons (groups of 8 bits) the information to select production rules from a Backus Naur Form (BNF) grammar. The user can tailor the grammar to produce solutions that incorporate domain knowledge by biasing the grammar to produce very specific forms of sentences. BNF is a notation that represents a language in the form of production rules. It is comprised of a set of non-terminals that can be mapped to elements of the set of terminals (the primitive symbols that can be used to construct the output program or sentence(s)), according to the production rules. A simple example BNF grammar is given below, where `<expr>` is the start symbol from which all programs are generated. The grammar states that `<expr>` can be replaced with either `<expr><op><expr>` or `<var>`. An `<op>` can become either `+`, `-`, or `*`, and a `<var>` can become either `x`, or `y`.

```
<expr> ::= <expr><op><expr> (0)
        | <var>         (1)

   <op> ::= +            (0)
        | -            (1)
        | *            (2)

   <var> ::= x            (0)
```

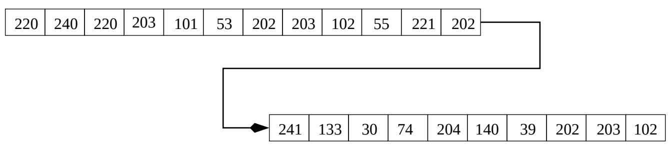
```
            | y            (1)
```

The grammar is used in a developmental process to construct a program by applying production rules, selected by the genome, beginning from the start symbol of the grammar. In order to select a production rule in GE, the next codon value on the genome is read, interpreted, and placed in the following formula:

$$Rule = Codon\ Value\ Mod\ Num.\ Rules$$

where Mod represents the modulus operator. Given the example individual's genome (where each 8-bit codon has been represented as an integer for ease of reading) in Fig.2, the first codon integer value is 220, and given that we have 2 rules to select from for `<expr>` as in the above example, we get $220\ Mod\ 2 = 0$. `<expr>` will therefore be replaced with `<expr><op><expr>`.

Figure 2 An example GE individual's genome represented as integers for ease of reading.



Beginning from the left hand side of the genome codon integer values are generated and used to select appropriate rules for the left-most non-terminal in the developing program from the BNF grammar, until one of the following situations arise:

- A complete program is generated. This occurs when all the non-terminals in the expression being mapped are transformed into elements from the terminal set of the BNF grammar.
- The end of the genome is reached, in which case the *wrapping* operator is invoked. This results in the return of the genome reading frame to the left hand side of the genome once again. The reading of codons will then continue unless an upper threshold representing the maximum number of wrapping events has occurred during this individual's mapping process.
- In the event that a threshold on the number of wrapping events has occurred and the individual is still incompletely mapped, the mapping process is halted, and the individual assigned the lowest possible fitness value.

Returning to the example individual, the left-most `<expr>` in `<expr><op><expr>` is mapped by reading the next codon integer value 240 and used in $240\ Mod\ 2 = 0$ to become another `<expr><op><expr>`. The developing program now looks like `<expr><op><expr><op><expr>`. Continuing to read subsequent codons and always mapping the left-most non-terminal the individual finally generates the expression `y*x-x-x+x`, leaving a number of unused codons at the end of the individual, which are deemed to be introns and simply ignored.

4.1.2 Mutation and Crossover

The representation of the individuals to which the genetic operators in GE are applied are variable-length linear strings. Due to this representation, the standard genetic operators such as crossover and mutation in the canonical Genetic Algorithm (Holland, 1975) can be applied to the underlying genotypic representation irrespective of the form of the phenotype. Mutation changes a bit or an integer to another random value, and one-point crossover swaps sections of the genetic code between parents. However, because of the mapping process, the effects on the phenotype can be complex.

In the case of mutation events at the codon level, the integer value of a codon will be modified. However, this change may or may not have effects at the phenotypic level. For example, given the following BNF production rule:

```
<variables> ::= a
             | b
```

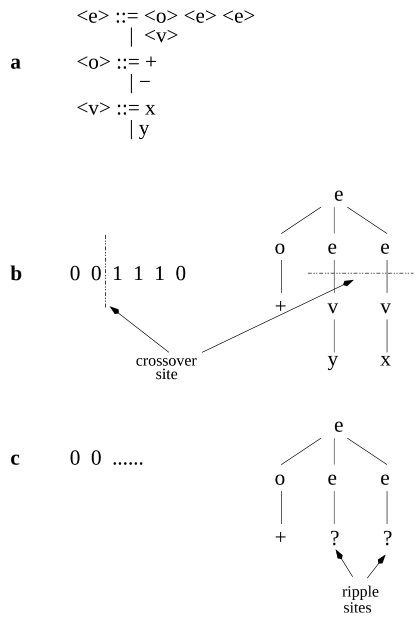
where the non-terminal $< variables >$ can be replaced with the variables a or b. Using this rule, an integer codon value of 10 will result in $< variables >$ being replaced with a; after mutation to 11 it will results in the use of b. On the other hand, a mutation event that results in an integer codon value of 10 becoming 12 leaves $< variables >$ being replaced with a. This type of mutation event is referred to as neutral mutation as it has no effect on the phenotype's functionality (fitness). A standard one-point crossover event on a GE chromosome results in the right-hand sides of the parental chromosomes undergoing a simple swap in a standard GA fashion. In the case of GE, again due to the mapping process, the impact on the phenotype may not be so simple. Crossover in GE has a ripple effect on the derivation sequence after the crossover point. Figure 3 helps to illustrate this process. The mapping process in GE can be described as a sequence of derivations, which in turn can be represented as a derivation tree. From the example derivation tree in Figure 3, we can see that a number of ripple sites at different locations on the derivation tree are created once the genetic material on the right-hand side of the crossover site is removed, the result being that the codons swapped over from the second parent are used to complete the derivation sequence at these incomplete points. A full description of GE can be found in O'Neill & Ryan (2003).

In the context of this study, GE is used to evolve trade execution strategies. A population of strategies is maintained and iteratively improved via a simulated evolutionary process. The structure of these rules is governed by a choice of grammar and the utility of evolved strategies is assessed by testing them in an artificial stock market environment. Both the choice of grammar and the design of the stock market environment are discussed in the next section.

5 Experimental Approach

In this section we describe the experimental approach adopted in this study. Two separate sets of simulation experiments were undertaken. In the first set of these experiments we examine the simpler case where a trader wishes to create a quality execution rule which will allow her to decide how to amend unfilled limit orders as

Figure 3 An illustration of ripple crossover in grammatical evolution using the chromosome (represented as rule choices) (b) and its corresponding derivation tree, which is produced as a result of the grammar (a). The site of one-point crossover is indicated (b) on the chromosome and the derivation tree. The resulting derivation tree ripple sites are indicated with '?' (c)



market conditions change, limiting the option to a decision as to whether to cross the spread immediately or whether to leave the limit order unchanged. Critically, the decision is determined by information concerning current market conditions at the time of order amendment. In the second set of experiments, we examine the case where the trader can create a rule which allows her to amend unfilled limit orders by altering their limit price to varying degrees. Again, the decision as to how to amend the limit price is determined by current market conditions.

Initially we describe the experimental design and the grammars used in each set of experiments. Then we describe the artificial stock market environment which is used to test the evolved execution strategies in both sets of experiments. A critical aspect of real-world design and testing of execution strategies, is that they cannot be easily backtested. Apart from this issue, another practical problem is that historical information from order books represents a single sample path through time. By implementing an artificial stock market environment, it is possible to create a closed world which allows the testing of new execution strategies over multiple runs, potentially allowing us to develop robust execution strategies.

5.1 Experiment One

In the first set of experiments we consider a large order of 10% of ADV of the artificial market, which is to be traded over one day. This order is equally divided into ten child orders. In all trade execution strategies, any uncompleted orders are crossed over the spread at the end of trading day in order to ensure order completion.

In our evolved strategies, the timing of order aggression is determined by an execution rule evolved using GE. At each amendment time (an integral multiple of Δt minutes after submission), if the market condition satisfies the condition of the execution rule, order aggression happens (the spread is crossed), otherwise, the uncompleted order is amended to the current best price as per the order book. An amendment frequency of 10 minutes is adopted in all limit order strategies. The grammar adopted in our first set of experiments is defined in Figure 4.

Figure 4 Grammars in Experiment One

```
<lc> ::= if (<stamt>)
             class = "CrossingSpread"
         else
             class = "NotCrossingSpread"
<stamt> ::= <cond1><op><cond2><op><cond3><op><cond4>
            <op><cond5><op><cond6>
<op> ::= and | or
<cond1> ::= (BidDepth>AvgBidDepth) is <boolean>
<cond2> ::= (AskDepth>AvgAskDepth) is <boolean>
<cond3> ::= (RelativeDepth>AvgRelativeDepth) is <boolean>
<cond4> ::= (Spread>AvgSpread) is <boolean>
<cond5> ::= (Volatility>AvgVolatility) is <boolean>
<cond6> ::= (PriceChange>AvgPriceChange) is <boolean>
<boolean> ::= True | False
```

In this grammar (Figure 4), we include the market information which GE can use in evolving execution strategies. Specifically, *AvgBidDepth* represents the average bid depth of the market, *AvgAskDepth* represents the average ask depth of the market, *AvgRelativeDepth* represents the average relative depth of the market, *AvgSpread* represents the average spread of the market, *AvgVolatility* represents the average volatility of the market and *AvgPriceChange* represents the average price change of the market. The six financial variables are observed at the time of order submission or at the time of subsequent order amendment.

The evolved rule consists of a Boolean expression which when evaluated using real-time information from the market outputs either a '0' or a '1'. In the latter case, *class* = "*CrossingSpread*", and the uncompleted limit order will be crossed over the bid-ask spread. Otherwise, its limit price will be amended to the current best price.

In order to evaluate the results from our evolved trading strategies we compare them against those of two benchmark execution strategies, which were used as benchmarks of trade execution strategies in previous work (Lim & Coggins, 2005b). One benchmark trade execution strategy is a pure market order strategy in which each child order is submitted as a market order every half hour. This strategy takes market liquidity immediately by crossing the bid-ask spread. The other benchmark trade execution strategy is a pure limit order strategy. Traders submit each child order as a limit order placed at the best price, and amend its price to best price at a fixed frequency until this order is fully executed or until the trading period expires. At the end of trading day, any unexecuted orders are traded by crossing the bid-ask spread in order to ensure order completion. For instance, a buy order s_n may be submitted to the market as a limit order placed at the best bid price with an amendment frequency of Δt minutes. If Δt minutes after submission, this limit order is not fully executed, it will be amended to the new best bid price. This amendment process continues in Δt intervals up to the end of trading day, at which time the uncompleted order(s) are traded as market orders by crossing the bid-ask spread.

In the market order strategy, *order aggression* (crossing the bid-ask spread) happens immediately after order submission which guarantees execution, at the cost of market impact. In the limit order strategy, order aggression happens at the end of trading period aiming to reduce market impact, at the risk of opportunity cost. A more sophisticated limit order strategy would allow for order aggression between these two extreme cases. A general limit order strategy is to cross the uncompleted limit order over the spread after submission but before the end of trading day.

5.2 Experiment Two

In the second set of experiments we again consider a large order of 10% of ADV of the artificial market, which is to be traded over one day (5 hours in the artificial market). This order is equally divided into ten child orders which are submitted to the market at intervals of thirty minutes over the trading day. Each child order is submitted as a limit order with an amendment frequency of ten minutes. We adopt two different order lifetimes, short (half an hour) and long (up to the end of the trading day). In all trade execution strategies, any uncompleted orders are crossed over the spread at the end of trading day in order to ensure order completion. GE is used to evolve efficient trade execution strategies which determine the aggressiveness level of each limit order at submission time and at amendment time. The grammar adopted in our second set of experiments is defined in Figure 5.
In this grammar (Figure 5), we include two additional variables $PercOfTradedVolume$ and $PercOfPastTime$ (used in experiment two) which represent the percentage of the traded volume accounting for the total volume V shares and the percentage of the past time accounting for the whole trading period at the observed time respectively.

As for experiment one, GE outputs a rule which produces a 'decision' depending on the current real-time state of the market. This determines the degree of aggressiveness of the re-pricing of currently unfilled limit orders. If the output is $class =$ "*AggressiveLimitPrice*", the limit orders to buy (sell) will be placed at one tick size above (below) the best bid (ask) price; if the output is $class =$ "*PassiveLimitPrice*", the limit orders to buy (sell) will be placed at one tick size below (above) the best bid (ask) price; if the output is $class =$ "*ModestLimitPrice*", the limit orders to buy (sell) will be placed at the best bid (ask) price.

As for experiment one, we benchmark the results from the GE strategies. Here we employ three benchmark limit order strategies based on benchmark limit order

Figure 5 Grammars in Experiment Two

```
<lc> ::= if (<stamt>)
        class = "AggressiveLimitPrice"
      else {
        if (<stamt>)
          class = "PassiveLimitPrice"
        else
          class = "ModestLimitPrice"
          }
<stamt> ::= (<stamt><op><stamt>)|<cond1>|<cond2>|
      <cond3>|<cond4>|<cond5>|<cond6>|<cond7>|<cond8>
<op> ::= and
<cond1> ::= (BidDepth<comp>AvgBidDepth)
<cond2> ::= (AskDepth<comp>AvgAskDepth)
<cond3> ::= (RelativeDepth<comp>AvgRelativeDepth)
<cond4> ::= (Spread<comp>AvgSpread)
<cond5> ::= (Volatility<comp>AvgVolatility)
<cond6> ::= (PriceChange<comp>AvgPriceChange)
<cond7> ::= (PercOfTradedVolume<comp><threshold>)
<cond8> ::= (PercOfPastTime<comp><threshold>)
<comp> ::= <less>|<more>|<lessE>|<moreE>
<less> ::= "<"
<more> ::= ">"
<lessE> ::= "<="
<moreE> ::= ">="
<threshold> ::= 0.1|0.2|0.3|0.4|0.5|0.6|0.7|0.8|0.9
```

strategy in previous work (Lim & Coggins, 2005b), which are simple aggressive limit order strategy (SA), simple modest limit order strategy (SM) and simple passive limit order strategy (SP), where the aggressiveness levels of limit orders are aggressive level (one tick size above/below the best bid/ask price for buy/sell limit orders), modest level (at best available price) and passive level (one tick size below/above the best bid/ask price for buy/sel limit orders). These strategies adopt the same amendment frequency and the same lifetimes as the GE strategies.

For both sets of experiments, we employ two periods (training and test periods). In the training period, GE is used to evolve trade execution strategies. Each individual is exposed to 20 continuous trading days in the artificial market and their fitness is calculated as their average VWAP ratio over the 20 trading days. The strategies are evolved over 40 generations, using GE settings of variable-length, one-point crossover at a probability of 0.9, mutation at a probability of 0.01, roulette selection, steady-state replacement and a population size of 100. In the test period, the best evolved strategy in the training period is tested *out of sample* over 240 days in the artificial market.

5.3 Simulating an Artificial Market

The training and evaluation of all trade execution strategies are implemented in an artificial limit order market, which is simulated using an agent-based model. Agent-based modelling is a computerised simulation consisting of a number of agents. The emergent properties of an agent-based model are the result of "bottom-up" processes, where the decisions of individual and interacting agent at a microscopic level determines the macroscopic behavior of the system. For a more detailed description of agent-based modelling in finance, please refer to LeBaron (2005), Samanidou et al. (2007) and Tesfatsion (2006). In this paper, our agent-based artificial limit order market is built based on the *Zero-Intelligence* (ZI) model (Daniel, 2006) with a continuous double auction price formation mechanism. The notion of ZI agents was first introduced in Gode and Sunder (1993). These agents randomly generate buy and sell orders. The orders are then submitted to a market agent, who manages all incoming orders according to the order matching mechanism in a real limit order market. The trading process is continuous, where unmatched orders are stored in an order book.

At each time step, an agent is equally likely to generate a buy order or a sell order. This order can be a market order, or a limit order, or a cancellation of a previous order, with probabilities λ_m, λ_l, and λ_c respectively. The sum of these probabilities is one ($\lambda_m + \lambda_l + \lambda_c = 1$). For a limit buy (sell) order, it has a probability of $\lambda_{inSpread}$ falling inside the bid-ask spread, a probability of λ_{atBest} falling at the best bid (ask) price, and a probability of λ_{inBook} falling off the best bid (ask) price in the book, ($\lambda_{inSpread} + \lambda_{atBest} + \lambda_{inBook} = 1$). The limit price inside the spread follows a uniform distribution (Toth et al., 2009). The limit price off the best bid (ask) price follows a power law distribution with the exponent of $(1 + \mu_1)$. The log order size of a market order follows a power law distribution with the exponent of $(1 + \mu_2)$, while the log order size of a limit order follows a power law distribution with the exponent of $(1 + \mu_3)$.

Table 7 Initial Parameters for Artificial Limit Order Market

Explanation	**Value**
Initial Price	$price^0 = 50$
Tick Price	$\delta = 0.01$
Probability of Order Cancellation	$\lambda_c = 0.34$
Probability of Market Order	$\lambda_m = 0.16$
Probability of Limit Order	$\lambda_l = 0.50$
Probability of Limit Order in Spread	$\lambda_{inSpread} = 0.32$
Probability of Limit Order at Best Quote	$\lambda_{atBest} = 0.33$
Probability of Limit Order off the Best Quote	$\lambda_{inBook} = 0.35$
Limit Price Power Law Exponent	$1 + \mu_1 = 2.5$
Market Order Size Power Law Exponent	$1 + \mu_2 = 2.7$
Limit Order Size Power Law Exponent	$1 + \mu_3 = 2.1$

As each incoming buy (sell) market order arrives, the market agent will match it with the best ask (bid) limit order stored in the order book. If this market order is fully filled by the first limit order, the unfilled part will be matched to the next best ask (bid) limit order until it is fully filled. As each incoming limit order arrives, the market agent will store it in the order book according to price and time priority. As each incoming cancelation order arrives, the market agent will delete the relevant limit order in the order book.

In order to ensure that the order flows generated by the artificial market are economically plausible, all the parameters in our model are derived from previous literature (Chakraborti et al., 2009; Farmer et al., 2005a,b; Mike & Farmer, 2008; Toth et al., 2009). The parameters used in our simulation are presented in Table 7. Critically, in our experiments the actions of the evolved execution strategies employed by our simulated trader impact on the state of the order book facing all the other agents in the stock market and therefore impact on their actions. In turn, the actions of those agents impact on the order book facing our trader and therefore on the utility of her execution strategy. In other words, the training and testing environment is *dynamic* and allows examination of the issues of market impact and opportunity cost. on trade execution. The use of an artificial stock market environment to test the utility of execution strategies is a novel contribution of this paper, and opens up the door to a wide range of future work in this domain.

6 Results

In this section we provide the results of our two sets of experiments.

6.1 Experiment One

Table 8 Results of best evolved GE strategies and two benchmark strategies

	SM		**SL**		**GE**			
	Mean	(S.D.)	Mean	(S.D.)	Mean	(S.D.)	H_1	H_2
Buy Order	69.64	(0.42%)	42.54	(1.45%)	-1.42	(0.49%)	0.00	0.01
Sell Order	68.73	(0.36%)	13.81	(1.59%)	-23.21	(0.48%)	0.00	0.01

The results (all out of sample) of buy strategies and sell strategies are provided in Table 8. The "Mean" is the average VWAP ratio of each strategy over the 240 days, and "S.D." represents the standard deviation of the average (daily) VWAP ratio. P-values for the null hypothesis $H_1 : mean_{SM} \leq mean_{GE}$ and $H_2 : mean_{SL} \leq mean_{GE}$ are also shown in the table, to indicate the degree of statistical significance of the performance improvement of GE strategies over the two simple strategies. The figures show that the null hypotheses are rejected at the ≤ 0.01 level.

Based on the results, GE evolved strategies notably outperform the two benchmark strategies, simple market order strategy (SM) and simple limit order strategy (SL). The negative VWAP ratios of -1.42 and -23.21 show that the GE evolved strategies achieve better execution price than the average execution price of the market. The small standard deviations of 0.49 and 0.48 indicate that our GE evolved strategy is robust over the tested trading days, supporting that the applicability of GE for evolving quality dynamic trade execution strategies. The performance of SL strategies seems more volatile than those of SM strategies and GE strategies, and the performance of SM strategies is more stable than those of the other two kinds of strategies. Comparing the performance of the strategies for buy and sell orders, we observe that the performances of sell strategies are better than those of buy strategies, which means that trading costs of buys are higher than those of sells. This asymmetry is consistent with previous empirical findings as discussed in Section 2.3.

6.2 Experiment Two

Table 9 Results of best evolved GE strategies and three benchmark strategies (Buy Orders)

	SA		**SM**		**SP**		**GE**				
	Mean	(S.D.)	Mean	(S.D.)	Mean	(S.D.)	Mean	(S.D.)	H_1	H_2	H_3
S-T	14.7	(1.74%)	66.25	(2.53%)	13.19	(1.76%)	-2.35	(1.52%)	0.01	0.00	0.01
L-T	5.14	(1.69%)	60.7	(2.03%)	9.37	(1.48%)	-5.86	(1.3%)	0.01	0.00	0.00

Table 10 Results of best evolved GE strategies and three benchmark strategies (Sell Orders)

	SA		**SM**		**SP**		**GE**				
	Mean	(S.D.)	Mean	(S.D.)	Mean	(S.D.)	Mean	(S.D.)	H_1	H_2	H_3
S-T	24.7	(1.86%)	43.6	(2.18%)	6.78	(2.03%)	-8.15	(1.87%)	0.00	0.00	0.01
L-T	5.28	(1.56%)	37.23	(2.27%)	4.91	(1.39%)	-12.07	(1.53%)	0.00	0.00	0.01

The results (all out of sample) of buy strategies and sell strategies are provided in Tables 9 & 10. The “S-T” represents short-term lifetime and the “L-T” represents long-term lifetime. The “Mean” is the average VWAP ratio of each strategy over the 240 days, and “S.D.” represents the standard deviation of the average (daily) VWAP ratio. P-values for the null hypothesis $H_1 : mean_{SA} \leq mean_{GE}$, $H_2 : mean_{SM} \leq mean_{GE}$, $H_3 : mean_{SP} \leq mean_{GE}$ are also shown in the table, to indicate the degree of statistical significance of the performance improvement of GE strategies over the two simple strategies. The figures show that the null hypotheses are rejected at the ≤ 0.01 level.

Based on the results, GE evolved strategies notably outperform the three benchmark strategies, simple aggressive limit order strategy (SA), simple modest limit order strategy (SM) and simple passive limit order strategy (SP). The performances of these four kinds of strategies can be described as:

$$Performance_{GE} > Performance_{SP} > Performance_{SA} > Performance_{SM}$$

GE strategies perform the best, while SM strategies perform the worst. The negative VWAP ratios show that the GE evolved strategies achieve better execution prices than the average execution price of the market. The values of standard deviation show that the performance of GE is more stable than most of the other three kinds of strategies over the tested trading days, and the performance of SM strategies is more volatile than the other three kinds of strategies. Comparing the performance of the strategies for buy and sell orders, we observe that the performances of sell strategies are better than those of buy strategies in most cases, which means that trading costs of buys are higher than those of sells. This asymmetry is consistent with previous empirical findings as discussed in Section 2.3. And we also observe that L-T strategies all perform better than S-T strategies, which indicate that strategies with longer lifetime can achieve better execution prices than those with short lifetime.

7 Conclusions and Future Work

Trade execution is concerned with the actual mechanics of trading an order. Traders wishing to trade large orders face a tradeoff between market impact and opportunity cost. Trade execution strategies are designed to balance out these costs, thereby minimising total trading cost relative to some benchmark like VWAP. Despite the importance of optimising trade execution, there has been relatively little attention paid in the literature to the application of evolutionary methods for this task. In this paper, GE was novelly applied for the purposes of evolving dynamic trade execution strategies, and an artificial limit order market was simulated for testing the evolved trade execution strategies. GE was found to be able to evolve quality trade execution strategies which proved highly competitive against two benchmark trade execution strategies. Also the results of two experiments both demonstrate the phenomenon of trading cost asymmetry which confirms the findings in previous studies. This consistence implies that our artificial market successfully mimics the price formation process of real markets.

The methodology developed in this study can contribute to more efficient investment implementation. This work provides practical evidence that using information content of limit order book can improve trade execution efficiency. This work fits into the market microstructure (optimisation of trading costs) which minimises the market impact.

There is notable scope for further research utilising GE in this problem domain. One obvious route is to widen the number of market variables which can be included in the evolved execution strategies. Another route is to evolve the full structure of the trade execution strategy. In our approach, we focused on one aspect of trade execution strategy (order aggressiveness), and other components like the number of orders are determined in advance. Future work will embrace the evolution of the full structure of trade execution strategy. We also note the potential for further work in this domain employing an ASM approach. Whilst a significant literature on ASM has emerged over the past 20 years, little attention has been paid to using this approach to create a dynamic environment for the examination of trade execution. Clear potential exists to employ an ASM approach for this purpose.

Acknowledgements

This publication has emanated from research conducted with the financial support of Science Foundation Ireland under Grant Number 08/SRC/FM1389.

*Bibliography

Aitken, M., Berkman, H. and Mak, D. (2001). 'The use of undisclosed limit orders on the Australian Stock Exchange', *Journal of Banking and Finance*, Vol. 25, No. 8, pp. 1589–1603.

Al-Suhaibani, M. and Kryzanowski, L. (2000). 'An exploratory analysis of the order book, and order flow and execution on the Saudi stock market', *Journal of Banking and Finance*, Vol. 24, No. 5, pp. 1323–1357.

Alfonsi, A., Fruth, A. and Schied, A. (2010). 'Optimal execution strategies in limit order books with general shape functions', *Quantitative Finance*, Vol. 10, No. 2, pp. 143–157.

Almgren, R. (2003). 'Optimal execution with nonlinear impact functions and trading-enhanced risk', *Applied Mathematical Finance*, Vol. 10, No. 1, pp. 1–18.

Almgren, R. (2008). 'Execution costs', in R. Cont (ed.), *Encyclopedia of Quantitative Finance*, pp. 1–5, Wiley, U.K.

Almgren, R. and Chriss, N. (1999). 'Value under liquidation', *Risk*, Vol. 12, No. 12, pp. 61–63.

Almgren, R. and Chriss, N. (2000). 'Optimal execution of portfolio transactions', *Journal of Risk*, Vol. 3, No. 2, pp. 5–39.

Almgren, R. and Lorenz, J. (2007). 'Adaptive Arrival Price', in B. Bruce (ed.), *Algorithmic Trading III: Precision, Control, Execution*, pp. 59–66, Institutional Investor.

ASX (2010). 'Algorithmic Trading and Market Access Arrangements'. Available at: http://www.asx.com.au/about/pdf/20100211.

Berkowitz, S., Logue, D. and Noser, E. (1988). 'The total cost of transactions on the NYSE', *Journal of Finance*, Vol. 43, No. 1, pp. 97–112.

Bertsimas, D. and Lo, A. (1998). 'Optimal control of execution costs', *Journal of Financial Markets*, Vol. 1, No. 1, pp. 1–50.

Bessembinder, H., Panayides, M. and Venkataraman, K. (2009). 'Hidden liquidity: An analysis of order exposure strategies in electronic stock markets', *Journal of Financial Economics*, Vol. 94, No. 3, pp. 361–383.

Biais, B., Hillion, P. and Spatt, C. (1995). 'An empirical analysis of the limit order book and the order flow in the Paris Bourse', *Journal of Finance*, Vol. 50, No. 3, pp. 1655–1689.

Bikker, J., Spierdijk, L. and Sluis, P. (2007). 'Market Impact Costs of Institutional Equity Trades', *Journal of International Money and Finance*, Vol. 1, No. 27, pp. 974–1000.

Bongiovanni, S., Borkovec, M. and Sinclair, R. (2006). 'Let's Play Hide-and-Seek: The Location and Size of Undisclosed Limit Order Volume', *Journal of Trading*, Vol. 1, No. 3, pp. 34–46.

Borkovec, M., Domowitz, I., Kiernan, B. and Serbin, V. (2009). 'Portfolio Optimization and the Cost of Trading'. Available at: www.itg.com/papers/PortfolioOpt031609MM.PDF.

Boscuk, A. and Lasfer, M. (2005). 'The information content of institutional trades on the London stock exchange', *Journal of Financial and Quantitative Analysis*, Vol. 40, No. 3, pp. 621–644.

Brabazon, A. and O'Neill, M. (2006). *Biologically Inspired Algorithms for Financial Modeling*, Spring-Verlag, Berlin Heidelberg.

Brabazon, A., O'Neill, M. and Dempsey, I. (2008). 'An Introduction to Evolutionary Computation in Finance', *IEEE Computational Intelligence Magazine*, Vol. 3, No. 4, pp. 42–55.

Cai, T. and Sofianos, G. (2006). 'Multi-day Executions', *Journal of Trading*, Vol. 4, No. 1, pp. 25–33.

Cao, C., Hansch, O. and Wang, X. (2008). 'Order Placement Strategies In A Pure Limit Order Book Market', *Journal of Financial Research*, Vol. 26, No. 2, pp. 113–140.

Cao, C., Hansch, O. and Wang, X. (2009). 'The information content of an open limit order book', *Journal of Futures Markets*, Vol. 29, No. 1, pp. 16–41.

CFA (2009). 'Market Microstructure: The Impact of Fragmentation under the Markets in Financial Instruments Directive', Vol. 2009, No. 13, pp. 1–60.

Chakraborti, A., Toke, I., Patriarca, M. and Abergel, F. (2009). 'Econophysics: Empirical facts and agent-based models'. Available at: http://arxiv.org/abs/0909.1974.

Chan, L. and Lakonishok, J. (1995). 'The behavior of stock prices around institutional trades', *Journal of Finance*, Vol. 50, No. 5, pp. 1147–1174.

Chan, L. and Lakonishok, J. (1997). 'Institutional Equity Trading Costs: NYSE versus NASDAQ', *Journal of Finance*, Vol. 52, No. 2, pp. 713–735.

Chan, L. K. and Lakonishok, J. (1993). 'Institutional trades and intraday stock price behavior', *Journal of Financial Economics*, Vol. 33, No. 3, pp. 173–199.

Chiyachantana, C., Jain, P., Jiang, C. and Wood, R. (2004). 'International Evidence on Institutional Trading Behavior and Price Impact', *Journal of Finance*, Vol. 59, No. 2, pp. 869–898.

Cho, J. and Nelling, E. (2000). 'The probability of limit-order execution', *Financial Analysts Journal*, Vol. 56, No. 5, pp. 28–33.

Chung, K., Van Ness, B. and Van Ness, R. (1999). 'Limit orders and the bid-ask spread', *Journal of Financial Economics*, Vol. 53, No. 3, pp. 255–287.

Coggins, R., Blazejewski, A. and Aitken, M. (2003). 'Optimal trade execution of equities in a limit order market', in *Proceedings of the IEEE International Conference on Computational Intelligence for Financial Engineering*, pp. 371–378.

Daniel, G. (2006). *Asynchronous simulations of a limit order book*, PhD thesis, University of Manchester, U.K.

Darwin, C. (1859). *On The Origin of Species: By Means of Natural Selection or the Preservation of Favoured Races in the Struggle for Life (reprinted 1985)*, Penguin Books, London.

Decovny, S. (2008). 'Asian-Pacific gears up for algorithmic trading', *Market View*, Vol. 1, No. 1, p. 13.

Duong, H., Kalev, P. and Krishnamurti (2009). 'Order aggressiveness of institutional and individual investors', *Pacific-Basin Finance Journal*, Vol. 1, No. 4, pp. 1–14.

Engle, R., Ferstenberg, R. and Russell, J. (2008). 'Measuring and Modeling Execution Cost and Risk'. Available at: http://ssrn.com/abstract=1211162.

Esser, A. and Monch, B. (2007). 'The navigation of an iceberg: The optimal use of hidden orders', *Finance Research Letters*, Vol. 4, No. 1, pp. 68–81.

Farmer, D., Gillemot, L., Iori, G., Krishnamurthy, S., Smith, E. and Daniels, M. (2005a). 'A random order placement model of price formation in the continuous double auction', in L. Blume & S. Durlauf (eds.), *The economy as an evolving complex system III*, Oxford University Press.

Farmer, J. D., Patelli, P. and Zovko, I. (2005b). 'The predictive power of zero intelligence models in financial markets', in *Proceedings of the National Academy of Sciences of the United States of America*, Vol. 102, pp. 2254–2259.

Gemmill, G. (1996). 'Transparency and Liquidity: A Study of Block Trades on the London Stock Exchange under Different Publication Rules', *Journal of Finance*, Vol. 51, No. 5, pp. 1765–1790.

Gresse, C. (2006). 'The Effect of Crossing-Network Trading on Dealer Market's Bid-Ask Spreads', *European Financial Management*, Vol. 12, No. 2, pp. 143–160.

Griffiths, M., Smith, B., Turnbull, D. and White, R. (2000). 'The costs and determinants of order aggressiveness', *Journal of Financial Economics*, Vol. 56, No. 1, pp. 65–88.

Hall, A. and Hautsch, N. (2006). 'Order aggressiveness and order book dynamics', *Empirical Economics*, Vol. 30, No. 1, pp. 973–1005.

Harris, L. and Hasbrouck, J. (1996). 'Market vs. Limit Orders - The SuperDOT Evidence On Order Submission Strategy', *Journal of Financial and Quantitative Analysis*, Vol. 31, No. 2, pp. 213–231.

Holland, J. (1975). *Adaptation in Natural and Artificial Systems*, University Michigan Press, Ann Arbor, MI.

Holthausen, R., Leftwich, R. and Mayers, D. (1987). 'The efffect of large block transactions on security prices: A cross-sectional analysis', *Journal of Financial Economics*, Vol. 19, No. 2, pp. 237–267.

Holthausen, R., Leftwich, R. and Mayers, D. (1990). 'Large block transactions, the speed of response, and temporary and permanent stock-price effects', *Journal of Financial Economics*, Vol. 26, No. 1, pp. 71–95.

Hora, M. (2006). 'Tactical Liquidity Trading and Intraday Volume'. Available at: http://ssrn.com/abstract=931667.

Keim, D. (1999). 'An analysis of mutual fund design: the case of investing in small-cap stocks', *Journal of Financial Economics*, Vol. 51, No. 2, pp. 173–194.

Keim, D. and Madhavan, A. (1995). 'Execution Costs and Investment Performance: An Empirical Analysis of Institutional Equity Trades'. Available at: http://ideas.repec.org/p/fth/pennfi/09-95.html.

Keim, D. and Madhavan, A. (1996). 'The upstairs market for large-block transactions: analysis and measurement of price effects', *Review of Financial Studies*, Vol. 9, No. 1, pp. 1–36.

Keim, D. and Madhavan, A. (1997). 'Transactions costs and investment style: an inter-exchange analysis of institutional equity trades', *Journal of Financial Economics*, Vol. 46, No. 3, pp. 265–292.

Keim, D. and Madhavan, A. (1998). 'The Cost of Institutional Equity Trades', *Financial Analysts Journal*, Vol. 54, No. 4, pp. 50–52.

Kim, K. (2007). *Electronic and Algorithmic Trading Technology*, Academic Press, U.S.A.

Kissell, R. (2006). *Algorithmic Trading Strategies*, PhD thesis, Fordham University.

Kissell, R. and Glantz, M. (2003). *Optimal Trading Strategies*, Amacom, U.S.A.

Kraus, A. and Stoll, H. (1972). 'Price impacts of block trading on the New York Stock Exchange', *Journal of Finance*, Vol. 27, No. 3, pp. 569–588.

LeBaron, B. (2005). 'Agent-based Computational Finance', in L. Tesfatsion & K. Judd (eds.), *Handbook of Computational Economics, Volume 2: Agent-based Computational Economics*, pp. 134–151, Elsevier.

Lim, M. and Coggins, R. (2005a). 'The immediate price impact of trades on the Australian Stock Exchange', *Quantitative Finance*, Vol. 5, No. 4, pp. 365–377.

Lim, M. and Coggins, R. (2005b). 'Optimal trade execution: an evolutionary approach', in *Proceedings of the IEEE Congress on Evolutionary Computation*, Vol. 2, pp. 1045–1052.

Lo, A., MacKinlay, A. and Zhang, J. (2002). 'Econometric models of limit-order executions', *Journal of Financial Economics*, Vol. 65, No. 1, pp. 31–71.

Lo, I. and Sapp, S. (2010). 'Order Aggressiveness and Quantity: How Are They Determined in a Limit Order Market?', *Journal of International Financial Markets, Institutions and Money*, Vol. 20, No. 3, pp. 223–243.

Mike, S. and Farmer, J. (2008). 'An empirical behavioral model of liquidity and volatility', *Journal of Economic Dynamics and Control*, Vol. 32, No. 1, pp. 200–234.

Ng, W. (2010). 'Dynamic Order Submission And Herding Behavior In Electronic Trading', *Journal of Financial Research*, Vol. 33, No. 1, pp. 27–43.

Obizhaeva, A. and Wang, J. (2005). 'Optimal Trading Strategy and Supply/Demand Dynamics'. Available at: http://www.nber.org/papers/w11444.

Omura, K., Tanigawa, Y. and Uno, J. (2000). 'Execution Probability of Limit Orders on the Tokyo Stock Exchange'. Available at: http://ssrn.com/abstract=252588.

O'Neill, M. and Ryan, C. (2003). *Grammatical Evolution: Evolutionary Automatic Programming in an Arbitrary Language*, Kluwer Academic Publishers, Boston USA.

Pascual, R. and Verdas, D. (2009). 'What pieces of limit order book information matter in explaining order choice by patient and impatient traders?', *Quantitative Finance*, Vol. 9, No. 1, pp. 527–545.

Perold, A. (1988). 'The Implementation Shortfall: Paper vs. Reality', *Journal of Portfolio Management*, Vol. 14, No. 3, pp. 4–9.

Ranaldo, A. (2004). 'Order Aggressiveness in Limit Order Book Markets', *Journal of Financial Markets*, Vol. 7, No. 1, pp. 53–74.

Saar, G. (2001). 'Price Impact Asymmetry of Block Trades: an institutional trading explanation', *Review of Financial Studies*, Vol. 14, No. 4, pp. 1153–1181.

Samanidou, E., Zschischang, E., Stauffer, D. and Lux, T. (2007). 'Agent-based models of financial markets', *Reports on Progress in Physics*, Vol. 70, No. 3, pp. 409–450.

Spencer, H. (1864). *The Principles of Biology*, Vol. 1, Williams and Norgate, London and Edinburgh.

Sussman, A. (2008). 'Imperfect Knowledge: International Perspectives on Transaction Cost Analysis'. Available at: http://www.tabbgroup.com/PublicationDetail.aspxPublicationID=336.

Tesfatsion, L. (2006). 'Agent-based computational economics: A constructive approach to economic theory', in L. Tesfatsion & K. Judd (eds.), *Handbook of computational economics: agent-based computational economics*, pp. 52–74, Elsevier, North Holland, Amsterdam.

Toth, B., Kertesz, J. and Farmer, J. (2009). 'Studies of the limit order book around large price changes', *European Physical Journal B.*, Vol. 71, No. 1, pp. 499–510.

Verhoeven, P., Ching, S. and Ng, H. (2004). 'Determinants of the decision to submit market or limit orders on the ASX', *Pacific-Basin Finance Journal*, Vol. 12, No. 3, pp. 1–18.

Xu, Y. (2009). 'Order aggressiveness on the ASX market', *International Journal of Economics and Finance*, Vol. 1, No. 1, pp. 51–75.

Yang, J. and Jiu, B. (2006). 'Algorithm Selection: A Quantitative Approach', in *Algorithmic Trading II: Precision, Control, Execution*, pp. 26–34, Institutional Investor.