RESEARCH PAPERS IN MANAGEMENT STUDIES

INTRADAY FX TRADING: REINFORCEMENT VS EVOLUTIONARY LEARNING

M A H Dempster, T W Payne and Y S Romahi

WP 23/2001

The Judge Institute of Management Studies
Trumpington Street
Cambridge CB2 1AG

Intraday FX Trading: Reinforcement vs Evolutionary Learning

M A H Dempster, T W Payne and Y S Romahi

WP 23/2001

Professor M A H Dempster & Y S Romahi
Centre for Financial Research
The Judge Institute of Management
University of Cambridge
Trumpington Street
Cambridge
CB2 1AG
UK
Tel.: +44 1223 339652
Email: {mahd2,yr206}@cam.ac.uk

INTRADAY FX TRADING: REINFORCEMENT VS EVOLUTIONARY LEARNING

M.A.H. DEMPSTER, T.W. PAYNE, AND Y.S. ROMAHI
CENTRE FOR FINANCIAL RESEARCH
JUDGE INSTITUTE OF MANAGEMENT
UNIVERSITY OF CAMBRIDGE

Academic investigation has recently provided reliable evidence that technical analysis, as used by traders in the foreign exchange markets, has predictive value regarding future movements of foreign exchange prices. In this paper, we examine the performance of two important computational learning algorithms in technical intraday foreign exchange trading. We diverge from traditional academic literature in two important directions. First of all, we examine performance of *intraday* trading at different frequencies. This is especially important considering that technical analysis is most commonly used in high frequency foreign exchange trading. We incorporate a credit line in modelling a trading framework, whereas previous work has generally used an investment model. This paper contrasts the performance of reinforcement learning and genetic algorithms as methods to infer optimal trading rules from a range of popular technical indicators. We examine several currencies (USDCHF, GBPUSD, USDJPY) at various trading frequencies and we also examine the effect of optimising the underlying technical indicator parameters rather than using standard parameters. We find that annual average returns range from 10-20% at low transaction costs (1-2bp) but are attentuated at lower trading frequencies. We demonstrate however that artificially constraining trading frequency has negative effects on the performance of the system. Instead, the system should be fed with the highest frequency data available and the computational learning algorithm left with the task of adapting the trading frequency based on the slippage. With regard to optimising indicator parameters, we show that this increases the correlation between the indicator recommendations themselves and thus has adverse effects on the information content of the data for reinforcement learning.

1 Introduction

Following the breakdown of the postwar Bretton Woods system of fixed exchange rates, the early 1970s saw explosive growth in the use of technical trading in the foreign exchange markets. Recently there has been growing evidence in academic analyses that technical analysis *does* have predictive value regarding future movements of foreign exchange prices [5, 17, 19]. However, academic investigation of technical trading has mainly concerned itself in the past with asset allocation frameworks [22, 23, 10, 29]. Here, following [5], we aim to investigate its effectiveness within the framework of a trader (issued with a credit line) operating within a high frequency foreign exchange setting (where most technical traders operate [33]). The trader's ultimate aim is to maximize some measure of performance which can be profit, risk-adjusted return or some other economic utility.

The relative performance of various optimization techniques in high frequency (intraday) foreign exchange trading is examined in this paper. We compare the performance of a *genetic algorithm* (GA) and a *reinforcement learning* (RL) system and demonstrate that the GA is able to outperform the RL system out-of-sample, despite superior performance of the RL in-sample. We also consider whether optimizing the indicator parameters as a pre-processing step leads to improved performance.

In Section 2 we give a brief review of earlier work in technical analysis. Sections 3 and 4 then introduce the GA and RL. The *stochastic optimization* problem to be solved by the compared methods [9] is defined in Section 5, while the next sections, 6 and 7, describe in more detail how each approach can be applied to solve this optimization problem approximately. The computational experiments performed are outlined and their results given in Section 8. Section 9 concludes with a discussion of these results and further avenues for research.

2 Technical Analysis

Despite a history of over 100 years amongst investment professionals, technical analysis has been marginalised in academic literature for much of this period due to academic belief in the *random walk* or *efficient market* hypothesis [30] and to the fact that until recently little effort was made by its proponents to test the predictability of its techniques. There has however been a recent resurgence of academic interest in the claims of technical analysis as potential evidence that markets are less efficient than previously believed. Lo and McKinley [19] state that

> ... financial markets *are* predictable to some degree, but far from being a symptom of inefficiency or irrationality, predictibility is the oil that lubricates the gears of capitalism.

In particular, they point out that price discovery is neither instantaneous nor costless, and frictions play a major role in determining the nature of competition and the function of markets.

Although the majority of the work that has examined technical analysis in the past has focused on equities [29, 10, 4], this approach has been widely adopted over the past three decades by foreign currency traders [24]. A survey by Taylor and Allen [33] found that in intraday currency trading 90% of respondents reported the use of technical analysis, with 60% stating that they regarded such information as at least as important as economic fundamentals. Neely *et al.* [24] argue that this can be partly explained by the unsatisfactory performance of exchange rate models based on economic fundamentals.

Net returns due to various trading rules in the foreign exchange markets have been examined [18, 27] to conclude that trading rules *are* able to earn significant returns net of transaction costs and that this phenomenon cannot be easily explained as compensation for bearing risk. Neely and Weller [22] note however that academic investigation of technical trading has not in fact been consistent with the practice of technical analysis. In the foreign exchange markets the majority of traders trade throughout the day using high-frequency data but aim to end the day with a net open position of zero. The academic literature has mainly tended to take much longer horizons into account and to consider only daily closing prices [35]. A survey of past work investigating the statistical properties of high frequency trading data is provided by Goodhart and O'Hara [13] and Goodhart and Curcio [12] examine the performance of the resistance levels published by Reuters and also of various filter rules identified by practitioners. Dempster and Jones [6, 7] examine profitability of the *systematic* application of the popular channel and head-and-shoulders patterns to intraday FX trading at various frequencies, including application with an overlay of statistically derived filtering rules. In subsequent work [15, 5], they apply a variety of technical trading rules to trade such data (see also Tan [32]) and study a genetic program which trades combinations of these rules on the same data [5]. With the exception of [5] none of the above studies report any evidence of *significant* profit opportunities, but by focusing on relatively narrow classes of rules their results do not necessarily exclude the possibility that a search over a broader class would reveal profitable strategies. Gencay *et al.* [11] in fact assert that simple trading models are able to earn significant returns after transaction costs in various foreign exchange markets using high frequency data.

3 Evolutionary Learning

The application of *computational learning* techniques to technical trading and finance in general has experienced significant growth in the recent past. *Neural networks* have received the most attention but have shown varying degrees of success and are difficult to interpret.

A shift in favour of user-transparent, non-black box evolutionary methods such as genetic algorithms and genetic programming has been recently witnessed. Such approaches have been used for option pricing [2, 3] and as an optimization tool in technical trading applications [5, 15, 24].

Evolutionary learning encompasses sets of algorithms that are inspired by Darwinian evolution. *Genetic algorithms* (GAs) are population based optimization algorithms first proposed by Holland [14]. They have since become an active research area within the artificial intelligence community and have been successfully applied to a broad range of hard problems. Their success is due in part to their several control parameters which allow them to be highly tuned to the specific problem at hand. *Genetic programming* (GP) is an extension proposed by Koza [16] which originally attempted to evolve computer programs.

GAs have been used to optimize a class of exponentially weighted moving average rules, but in the study of Pictet *et al.* [28] ran into serious overfitting and poor out-of-sample performance. They report 3.6% to 9.6% annual excess returns net of transaction costs, but as the underlying models are not publicly available their results are difficult to evaluate. Neely and Weller [22] report for their GA approach strong evidence of predictability in the data measured out-of-sample when transaction costs are set to zero, but no evidence of profitable trading opportunities when realistic transaction costs are applied and trading is restricted to times of high market activity.

4 Reinforcement Learning

Reinforcement learning has to date received limited but growing attention in financial applications. The reinforcement learning technique is strongly influenced by the theory of *Markov decision processes* (MDPs) which evolved from attempts to understand the problem of making a sequence of decisions under uncertainty when each decision can depend on the previous decisions and their outcomes.

Watkins [34] brought about the integration of reinforcement learning with the theory of MDPs by devising the method of *Q-learning* for estimating action-value functions. The nature of reinforcement learning makes it possible to approximate optimal policies in ways that put more effort into learning to make good decisions for frequently encountered situations at the expense of less effort for less frequently encountered situations [31]. This is a key property that distinguishes reinforcement learning from other approaches for approximate solution of

MDPs.

As fundamental research in reinforcement learning advances, applications to finance have started to emerge. Neuneier [26, 25] has demonstrated Q-Learning in an asset allocation framework, applying it to the German DAX stock index. Moody *et al.* [21] examine a recurrent reinforcement learning algorithm that seeks to optimize an online estimate of the Sharpe ratio. They also compare the *recurrent RL* approach to that of Q-learning. Dempster et al. [9] similarly compare GAs and RLs, as well as the exact solution of an appropriate Markov decision problem and a simple heuristic, for intraday foreign exchange in an asset allocation framework

The main shortcoming in this work (see also [22, 23, 24]) is that most technical analysts active in the foreign exchange market are *traders* who operate at the high frequency level and are not well modelled as investors. Further, even technical traders who look for patterns in daily data alone often use tick data for confirmatory trade entry signals. In this work we therefore depart from most past experience by developing a *trading system* [5] rather than an asset allocation system.

5 Foundations

5.1 Modelling trading

As noted above, trading strategies have been traditionally evaluated as asset allocation strategies in which the agent has a lump sum of money and must choose at each timestep whether to allocate this money to be held in the home currency or the foreign currency (possibly earning the overnight interest rate in the chosen currency). Any profit or loss made is added to or subtracted from the lump sum to be allocated in the next timestep, *i.e.* reinvested.

High frequency traders however are typically able to draw on a fixed credit line from which they may borrow in either the home or the foreign currency. The money borrowed is then converted to the other currency at the current market rate to hold cash in one currency and a debt in the other. When the trader wishes to close out his position he converts his cash at the new (hopefully advantageous) exchange rate and pays any profit into or shortfall from his account. Thus he places a series of fix-sized bets.

More formally, a trade with proportional transaction cost c, exchange rates (expressed per unit of home currency) of F_t at trade entry and $F_{t'}$ at trade exit, drawing on a credit line of C units of home currency and taking a long position in the foreign currency will yield a percent return per unit of home currency of

$$100\left[\frac{F_t}{F_{t'}}(1-c)^2-1\right]. \tag{1}$$

If a short position is taken in the foreign currency then C/F_t units of foreign currency are drawn from the credit line and the percent return per unit of home currency is

$$100\left[(1-c)-\frac{F_{t'}}{F_t}\frac{1}{(1-c)}\right]. \tag{2}$$

The asymmetry of the equations is apparent and arises from the result of closing out the short position in the foreign currency being credited or debited to the home currency account. Both types of possible trades involve transaction costs or *slippage* c being paid per unit on two currency conversions. (See [5] for a discussion of the composition of these proportional transaction costs.)

5.2 Technical indicators

We consider two sets of technical indicators which are used as input for our trading strategies. First, as in [9], we use eight commonly used indicators with the parameters suggested in [1]. These are Price Channel Breakout, Adaptive Moving Average, Relative Strength Index, Stochastics, Moving Average Convergence/Divergence, Moving Average Crossover, Momentum Oscillator and Commodity Channel Index. Each indicator produces two signals: buy (long) or not buy, and sell (short) or not sell. These sixteen binary signals together define the market state $\mathbf{s}_t \in \mathcal{S} = \{0,1\}^{16}$. We refer to this set of indicators as the *unoptimised* set.

5.3 Optimizing Indicator Parameters

Second we use a *restricted* set of four indicators whose parameters have been chosen by exhaustive search on a fine grid (described below) to maximize the return over the in-sample period: Moving Average Crossover, Momentum Oscillator, Price Channel Breakout and Relative Strength Index were chosen because of their limited number of parameters (no more than three) and simple formulation. This indicator set is termed the *optimized* set and corresponds to a market state space $\mathcal{S} = \{0,1\}^{8}$.

In order to examine the effects of optimization of the indicators, the inputs to the learning systems were manually restricted to the same restricted set of indicators and the results are presented in §8.2.

Due to the large combination of currencies, data frequencies and transaction costs, the objective of the optimisation was to maximise the *break-even transaction cost.* This is defined as the slippage cost c at which the strategy would break even - ie. make 0 return - over a *trading horizon* T involving N_T trades.

It is computed empirically over the in-sample period as the solution of equation (3) and provides a good estimate of the quality of a strategy that is independent of the actual transaction cost. Therefore, parameters need only be chosen for each currency/frequency pair.

$$\sum_{i=1}^{N_T} r_i(F_{t_i}, F_{t'_i}) = 0, \tag{3}$$

where $r_i(\mathbf{F}_{t_i}, \mathbf{F}_{t'_i})$ denotes the return on the i^{th} trade given by (1) or (2) as appropriate and t_i and t'_i denote the entry and exit times of the trade.

Momentum Oscillator and Price Channel Breakout each require a single parameter representing a lookback period. Values of 1 to 250 multiples of the data frequency were tested using exhaustive search over each in-sample period and the parameter maximising in-sample break even transaction cost was chosen to be used in the out-of-sample period. Moving Average Crossover requires two lookback parameters, and each parameter was allowed to vary between 1 and 250 multiples of the underlying frequency. Relative Strength Index is a function of three parameters: a lookback period (again values between 1 and 250 were tested), and an upper and a lower threshold. The upper threshold was tested using integer values between 50 and 100, and the lower thresold was tested with values 1 to 50.

5.4 Trading strategies

If we consider the indicator signals (market state) over time $\mathbf{s}$ to be a *stochastic process* driven by the exchange rate process $\mathbf{F}$ we could make the required trading decisions by solving the *stochastic optimization problem* defined by the maximisation of expected return over the trading horizon net of transactions costs

$$\mathbb{E} \sum_{i=1}^{\mathbf{N}_T} r_i(\mathbf{F}_{t_i}, \mathbf{F}_{t'_i}) \tag{4}$$

involving a random number $\mathbf{N}_T$ of trades to the horizon.

However, the statistics of the processes $\mathbf{F}$ and $\mathbf{s}$ are entirely unknown. The genetic algorithm and reinforcement learning approaches we consider therefore attempt to find approximate solutions to this problem by discovering a (feedback) *trading strategy* $\phi : \mathcal{S} \times \{l, s\} \rightarrow$

$\{l, s\}$ that maps the current market state $\mathbf{s}_t$ and current position (long or short) to a new position (long or short). It should be noted that although our trading strategies ϕ are formally *Markovian* (feedback rules), the technical indicators require a number of periods of previous values of $\mathbf{F}$ to compute the corresponding $0-1$ entries in $\mathbf{s}_t$.

5.5 Evaluation

Since we do not have an explicit probabilistic model of how exchange rates evolve, we adopt the familiar approach of dividing out data series into an *in-sample* region, over which we optimize the the performance of a candidate trading strategy, and an *out-of-sample* region where the strategy is ultimately tested. This is illustrated in Figure 1 for the 36 months of out-of-sample tests reported in Section 7.

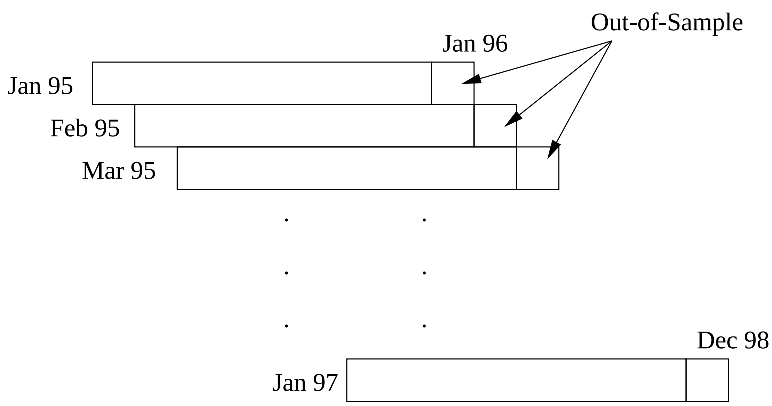

Figure 1: In-sample vs out-of-sample regions

6 Applying RL to the Technical Trading Problem

The ultimate goal of a reinforcement learning based trading system is to optimize some relevant measure of trading system performance such as profit, economic utility or risk-adjusted return. This paper follows the approach of [9] which is summarised here.

Reinforcement learning systems consist of an *agent* interacting with an *environment*. At each time step t the agent *perceives* the state of the environment $s_t \in \mathcal{S}$ and chooses an *action* $a_t \in \mathcal{A}$ from the set of available actions in state s_t. As a consequence of this action the agent observes the new state of the environment s_{t+1} and receives a *reward* r_t. This can be defined as a *dynamic programming* problem where the objective is to find the *policy* π (state to action mapping) that maximises the *optimal value function* V^* given by

$$V^*(s) = \max_a \mathbb{E}\{\boldsymbol{r}_{t+1} + \gamma V^*(\boldsymbol{s}_{t+1}) | \boldsymbol{s}_t = s\}, \tag{5}$$

where γ is the *discount factor* representing the preference for immediate over future rewards.

The value of state s can be considered in terms of the values of each action a that can be taken from that state assuming that policy π is followed subsequently. This value Q^* is referred to as the *Q-value* and is given by

$$Q^*(s,a) = \mathbb{E}\{\boldsymbol{r}_{t+1} + \gamma \max_{a'} Q^*(\boldsymbol{s}_{t+1}, a') | \boldsymbol{s}_t = s, \boldsymbol{a}_t = a\}. \tag{6}$$

The optimal value function expresses the obvious fact that the value of a state under an optimal policy must equal the expected return for the best action from that state, *i.e.*

$$V^*(s) = \max_a Q^*(s,a).$$

The functions Q^* and V^* provide the basis for learning algorithms expressed as solutions of *Markov decision problems.*

We use Watkins's *Q-learning algorithm* [34] that estimates the *Q-value* function using data from the previous learning episode. The Q-learning *update* is given by the backward recursion

$$Q(s_t, a_t) \leftarrow Q(s_{t_c}, a_{t_c}) + \alpha[r_{t+1} + \gamma \max_a Q(s_{t+1}, a) - Q(s_{t_c}, a_{t_c})], \tag{7}$$

where the current *state-action pair* $(s_t, a_t) := (s_{t_c}, a_{t_c})$, that from the previous learning episode. At each iteration (episode) of the learning algorithm, the action-value pairs associated with all the states are updated and over a large number of iterations their values converge to those optimal for (6) [31].

For our trader the state s_t is the market state defined by the technical indicators and the set of actions $\mathcal{A}$ is whether to take a long or short position in the foreign currency (and is not state dependent). The *reward function* r_{t+1} is the differential change in value of the agent's portfolio from time t to $t+1$ following Maes and Brookes' [20] suggestion that immediate rewards are most effective.

7 Applying the Genetic Algorithm to the Technical Trading Problem

This section again summarises the approach in [9]. The genetic algorithm also searches for a function that directly maps the market state into an action. So-called *trading rules* are represented as binary trees where the terminals are the individual indicators defining the market state and the non-terminals are Boolean functions `AND`, `OR` and `XOR`. To determine the action to be taken at each time step the rule is evaluated recursively. The value of a terminal is the value of a particular indicator and the value of a non-terminal is the value of the Boolean function it represents applied to its sub-trees. The action taken is the value of the root node, where `TRUE` is taken as representing a long position in the foreign currency and `FALSE` represents a short position.

Initially, a population of 100 rules is generated randomly. Evolution then proceeds by generating a new population from this. Rules are ranked on their in-sample performance and are chosen in proportion to their rank to undergo either crossover or mutation and are then inserted into the new population. *Crossover* combines two rules to form two "children" by swapping sub-trees; *mutation* makes a random change to a single rule to create a new rule. This process continues for 100 generations, after which the single best in-sample performing rule found during the entire run is chosen to be the output of the genetic algorithm [5].

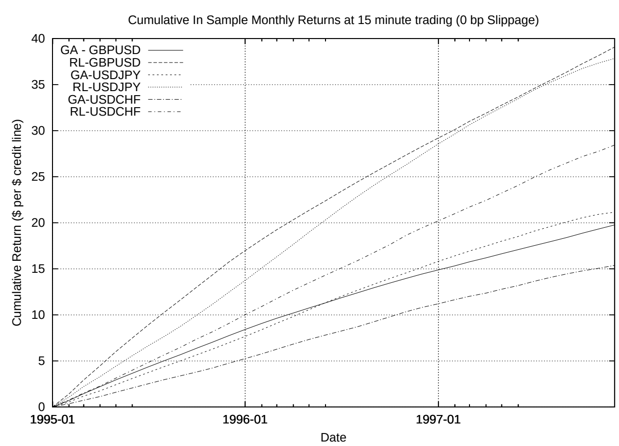

Figure 2: Insample Results: RL vs GA at 0bp (15 minute trading)

8 Numerical experiments

The results reported below were obtained by applying the above approaches to GBPUSD, USDCHF and USDJPY mid-point exchange rate data at 1 minute frequency from January 1994 until January 1998, using a moving window of 1 year for training (fitting) followed by 1 month out-of-sample testing.

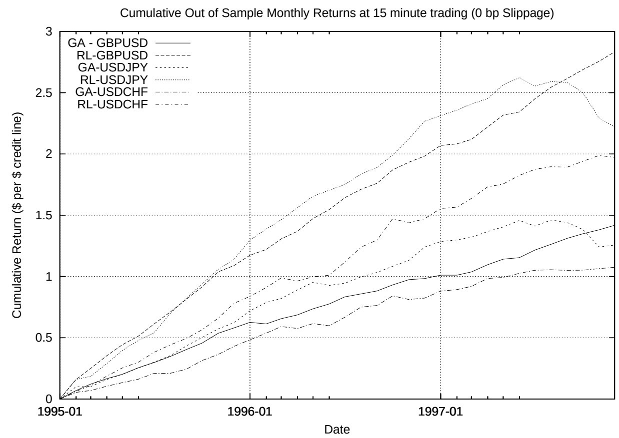

Figure 3: Out-of-sample Monthly Returns: RL vs GA at 0bp (15 minute trading)

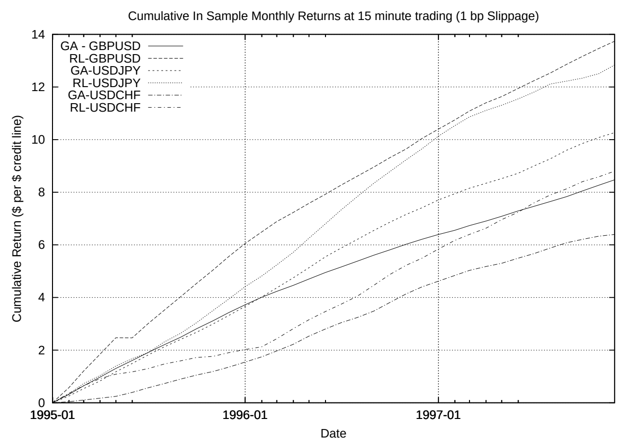

Figure 4: Insample Results: RL vs GA at 1bp (15 minute trading)

There are several questions that we seek to answer here:

- How do the methods compare?
- Does optimising the input indicator parameters improve performance?
- What are the effects of enforcing different trading frequencies?
- Are the results significant?

8.1 How do the methods compare?

The methods have been applied to several different currency crosses to ensure that the results obtained indicate general performance which is not particular to a specific currency pair.

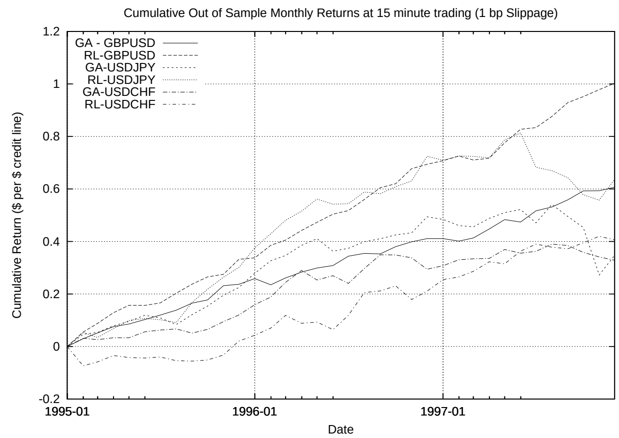

Figure 5: Out-of-sample Monthly Returns: RL vs GA at 1bp (15 minute trading)

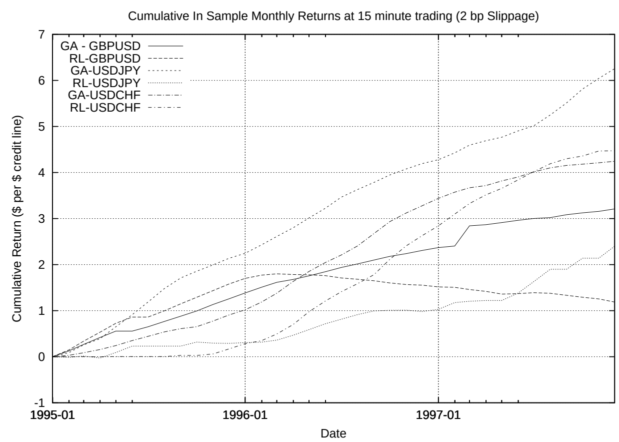

Figure 6: Insample Results: RL vs GA at 2bp (15 minute trading)

Several issues are immediately evident in the results.

In line with previous work [9] it is noted that at 15 minute trading frequency when no slippage is charged the RL consistently outperforms the GA both in-sample and out-of-sample. This is illustrated in Figures 2 and 3 (see also Figure 14).

As slippage is increased (Figures 4, 5, 6, 7) however we note that the GA starts to outperform the RL, which is the case at higher slippage values across the different frequencies. In the out-of-sample results for 15-minute trading (Figures 7 and 9), as well as for 8-hour trading (Figures 10 and 11) at slippage values of 2bp and 4bp, the GA consistently outperforms the RL.

The GA is constrained in depth and thus by design cannot overfit the insample period, whereas this is not the case for the RL system which is given free reign with the input

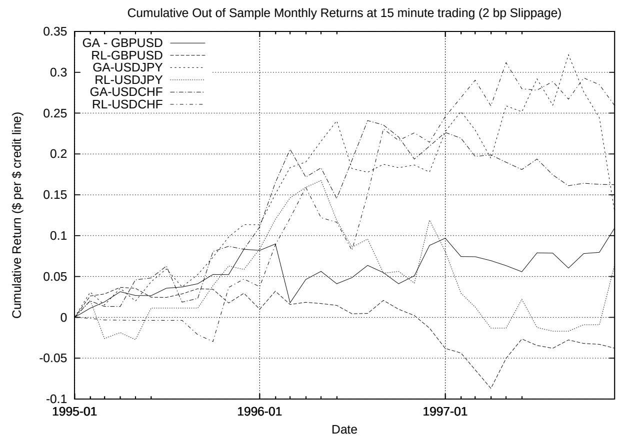

Figure 7: Out-of-sample Monthly Returns: RL vs GA at 2bp (15 minute trading)

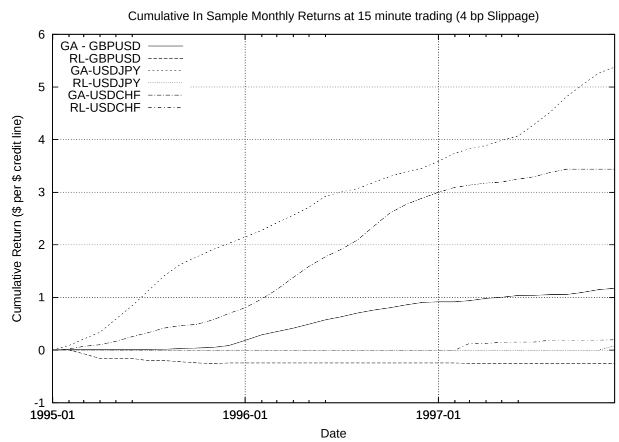

Figure 8: Insample Results: RL vs GA at 4bp (15 minute trading)

indicators. This leads to the view that the RL performance could be improved by introducing a method of constraining the inputs fed into the system. To remedy this, we are currently examining an evolutionary RL system where a GA is utilised to optimise the input sets passed into the RL.

In order to further evaluate the relative performance of the trading models we now define several risk measures found in the financial literature and adjust all returns for risk. A measure commonly used to evaluate portfolio models is the *Sharpe ratio* given by

$$\frac{\hat{\mu}_{R_{\text{month}}}}{\hat{\sigma}_{R_{\text{month}}}}, \tag{8}$$

where $\hat{\mu}_{R_{\text{month}}}$ and $\hat{\sigma}_{R_{\text{month}}}$ denote respectively the mean and standard deviation of monthly out-of-sample returns over the test period of 36 months (since no risk free rate is applicable

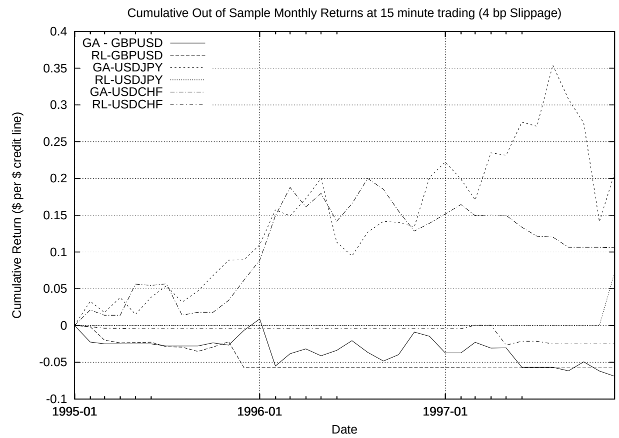

Figure 9: Out-of-sample Monthly Returns: RL vs GA at 4bp (15 minute trading)

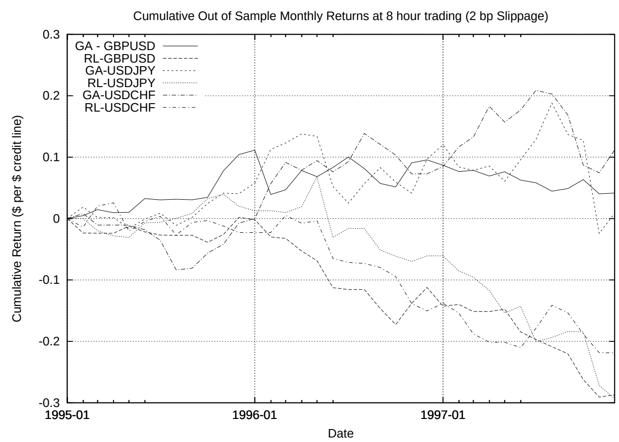

Figure 10: Out-of-sample Results: RL vs GA at 2bp (8 hour trading)

in the trading context).

The Sharpe ratio as shown in Table 1 demonstrates that on the dataset used we are able to gain quite acceptable returns up to and including a slippage value of 2bp. Olsen [11] notes however that the Sharpe ratio is numerically unstable for small variance of returns and cannot consider the clustering of profit and loss trades. Jones [15] adds that the Sharpe ratio penalizes strategies for upside volatility and that no path dependence means that a successful strategy can be reported that in fact suffers from significant drawdown. An improvement therefore is the *Sortino ratio* which considers downside variance in the denominator of (6) and thus only penalizes strategies for downside volatility. We quote the corresponding Sortino ratios in Table 2.

These results further confirm what was illustrated earlier: whereas at low slippage values

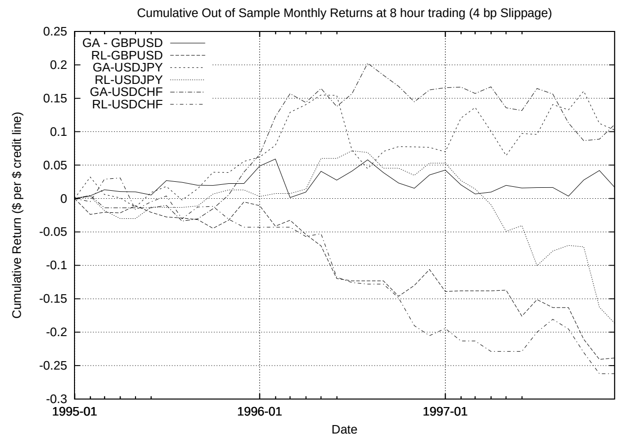

Figure 11: Out-of-sample Monthly Returns: RL vs GA at 4bp (8 hour trading)

the performance of the RL was superior to the GA, as slippage increased the performance disparity narrowed and the GA surpassed the RL at 2bp. Beyond 2bp we find that both methods - as applied here - are unable to make positive returns.

Table 1: Out-of-Sample annualized Sharpe ratios - 15 minute trading

	GBPUSD	USDCHF	USDJPY
RL - 0bp	9.78	4.45	2.84
GA - 0bp	7.59	3.58	2.32
RL - 1bp	5.62	1.14	1.16
GA - 1bp	3.86	1.14	0.55
RL - 2bp	-0.16	0.82	-0.03
GA - 2bp	0.48	0.57	0.30

8.2 Does optimizing the input indicators improve performance?

As mentioned in §5.3, the indicator parameters were individually optimized by exhaustive search over the in-sample period. The results illustrate that for both computational learning techniques performance using the optimized indicators as input was inferior to the case of the unoptimised indicators. Performance with the optimized indicators is shown in Figures 12 and 13. For comparison purposes the unoptimized RL illustrated was fed with only the 4

Table 2: Out-of-sample Sortino ratios - 15 minute trading

	GBPUSD	USDCHF	USDJPY
RL - 0bp	N/A[1]	7.03	1.50
GA - 0bp	19.33	4.55	1.15
RL - 1bp	11.83	0.58	0.54
GA - 1bp	3.74	0.63	0.20
RL - 2bp	-0.07	0.52	-0.01
GA - 2bp	0.19	0.29	0.12

indicators (8 signals) used in the optimization set. It is interesting to note that this version of the RL consistently outperformed the RL that was fed with the full 8 indicator (16 signal) set. This is further evidence of the overfitting problem faced by the RL.

This is not entirely surprising however when we consider that the indicators themselves attempt to capture similar dynamics: trends or trend reversals. With optimization the correlation between the indicators is substantially increased and thus less independent information is being fed into the computational learning techniques.

8.3 What are the effects of forcing different trading frequencies?

The general trend of these results is that trading is more profitable at the higher frequencies. This is in line with what we might expect because both computational learning techniques should be able to learn to trade less as slippage is increased and might therefore be expected to be better than artificially constraining trading frequency as is effectively happening when we input pre-processed data at different frequencies.

In the no slippage case (Figure 14) it is obvious why we expect the results shown. Since no slippage is charged we expect profits to be lower if we artificially limit trading ability. However the same conclusions are found when we consider performance at higher slippage values. This is somewhat surprising at first glance but implies that both computational learning methods are themselves better able to limit trading as slippage is increased than by having an external constraint imposed. The implication is that in order to maximize profits the highest frequency data possible should be input and thereafter the algorithms should adapt their trading frequencies automatically.

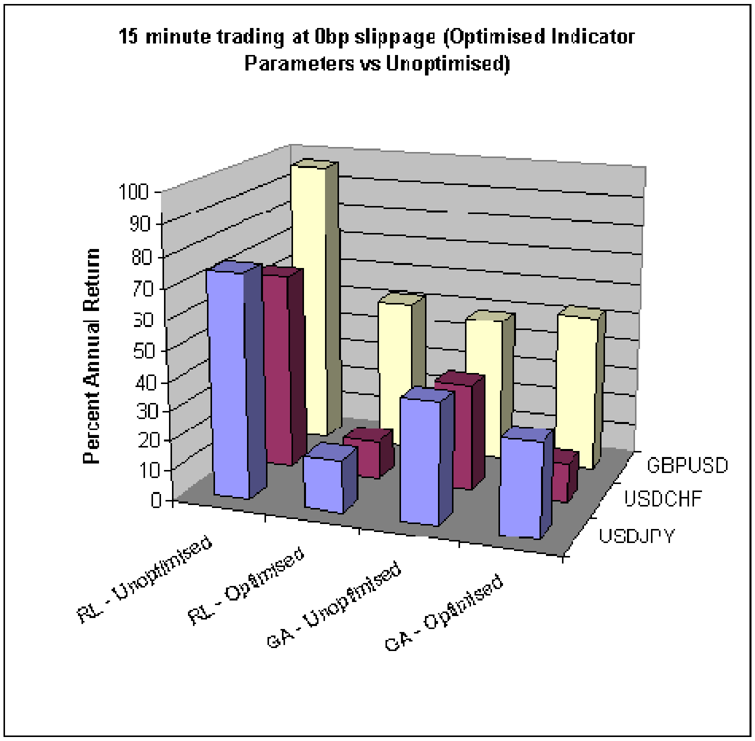

Figure 12: Performance with Optimised Indicators vs Unoptimised: GBPUSD 15 minute trading at 0bp slippage

Tables 3 to 8 in fact illustrate that the average number of monthly trades automatically decreases as slippage is increased. The tables clearly show that at the lower slippage values and trading frequencies both systems are unable to trade as often as at the higher frequencies and this restricts their ability to exploit profitable opportunities. As the slippage is increased the higher frequency variants automatically trade less frequently and the average numbers of monthly trades converge to similar numbers regardless of the frequency of the data we input. Thus we might expect that if effective trading frequencies are similar regardless of the input trading frequency, arbitrarily placing trading points in time by feeding the algorithm with lower frequency data must adversely affect the results.

8.4 Are the results significant?

In this section the results are examined more closely to demonstrate their statistical significance.

In order to determine the statistical significance of our trading results, we utilize the fol-

Table 3: Reinforcement Learning - Out of Sample Average Monthly Trading Frequency

Slippage	0 bp	1 bp	2 bp	4 bp	8 bp
GBPUSD - 15 minute	324.52	135.14	112.44	0.36	0
GBPUSD - 1 hour	71.61	37.78	29.53	4.64	0.39
GBPUSD - 2 hour	38.61	22.33	8.97	3.17	2.58
GBPUSD - 4 hour	16.31	11.92	10.56	7.22	4.61
GBPUSD - 8 hour	11.19	8.17	6.92	6.14	1.89

Table 4: Reinforcement Learning: Out of Sample Average Monthly Trading Frequency

Slippage	0 bp	1 bp	2 bp	4 bp	8 bp
USDJPY - 15 minute	492.25	86.81	58.53	5.69	0.39
USDJPY - 1 hour	75.86	31.11	13.30	4.67	3.33
USDJPY - 2 hour	37.42	22.11	17.31	15.61	9.67
USDJPY - 4 hour	21.39	11.72	10.13	8.56	4.83
USDJPY - 8 hour	11.03	9.19	8.67	7.78	5.25

Table 5: Reinforcement Learning: Out of Sample Average Monthly Trading Frequency

Slippage	0 bp	1 bp	2 bp	4 bp	8 bp
USDCHF - 15 minute	286.72	106.67	22.69	1.81	0.11
USDCHF - 1 hour	67	35.69	18.03	3.89	0.47
USDCHF - 2 hour	47.86	22.19	11.83	4.64	0.58
USDCHF - 4 hour	14.72	13.58	10.11	6.28	1.36
USDCHF - 8 hour	6.11	5.08	4.33	3.64	2.61

Table 6: Genetic Algorithm: Out of Sample Average Monthly Trading Frequency - GBPUSD

Slippage	0 bp	1 bp	2 bp	4 bp	8 bp
GBPUSD - 15 minute	373.13	148	84.22	2.58	0.3
GBPUSD - 1 hour	60.52	36.83	25.44	12.63	1.33
GBPUSD - 2 hour	24.11	14.3	11.13	4.19	1.63
GBPUSD - 4 hour	15.58	11.3	9.63	7.25	4.16
GBPUSD - 8 hour	4.8	4.66	4.52	2.8	1.61

Table 7: Genetic Algorithm: Out of Sample Average Monthly Trading Frequency - USDJPY

Slippage	0 bp	1 bp	2 bp	4 bp	8 bp
USDJPY - 15 minute	380.86	126.66	70.44	3.97	1.11
USDJPY - 1 hour	46.3	18.08	10.19	6.05	1.8
USDJPY - 2 hour	20.08	13.58	8.3	4	4.27
USDJPY - 4 hour	8	6.75	4.91	5.41	3.11
USDJPY - 8 hour	3.86	4.27	3.55	2.13	1.8

Table 8: Genetic Algorithm: Out of Sample Average Monthly Trading Frequency - USDCHF

Slippage	0 bp	1 bp	2 bp	4 bp	8 bp
USDCHF - 15 minute	389.91	105.94	21.25	1.36	0.52
USDCHF - 1 hour	40.08	13.3	9	3.5	0.77
USDCHF - 2 hour	22.27	15.66	6.94	5.16	1.86
USDCHF - 4 hour	13.5	9.38	7.5	4.69	2.52
USDCHF - 8 hour	5.19	4.69	3	1.27	1.08

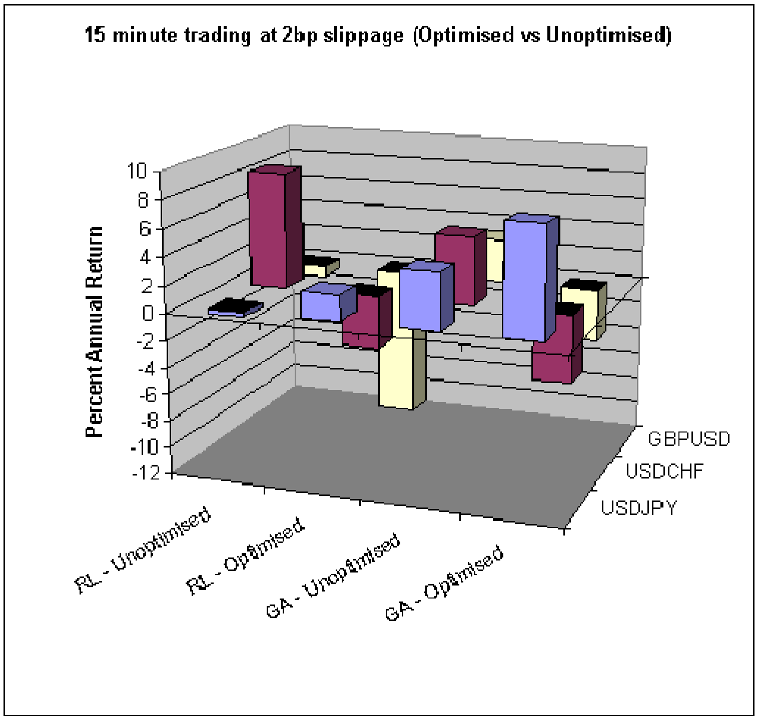

Figure 13: Performance with Optimised Indicators vs Unoptimised: GBPUSD 15 minute trading at 2bp slippage

lowing simple non-parametric binomial test [5]. Taking as the null hypothesis that cumulative trading profits and losses are periodically sampled from a continuous time stationary ergodic process with state distribution having median zero, profits and losses over an out-of-sample trading period (here one month) are equally likely with probability 1/2. It follows that over n monthly periods the *number of profitable months* n_+ is binomially distributed with parameters n and $\frac{1}{2}$. We therefore test the two-tailed alternative hypothesis that median profit and loss is non-zero with the statistic n_+.

The returns illustrated in Figures 3 and 5 are seen to be significantly profitable at the 10% level across both methods (see the p-levels given in Tables 9 to 11). Looking more closely at the USDCHF 15 minute trading performance at 1bp slippage for illustrative purposes, where monthly out-of-sample returns were shown in Figure 5, we find that with n_+=22 for the RL and n_+=25 for the GA with $n = 36$ months, the p-value is 90.87% and 99.02% respectively, giving us a significance at the 10% level for the RL and at the 1% level for the GA.

The significance tables illustrate that at the two highest frequencies and at zero slippage, the results are almost all significant at the 5% level. At 1bp slippage, the GA results remain

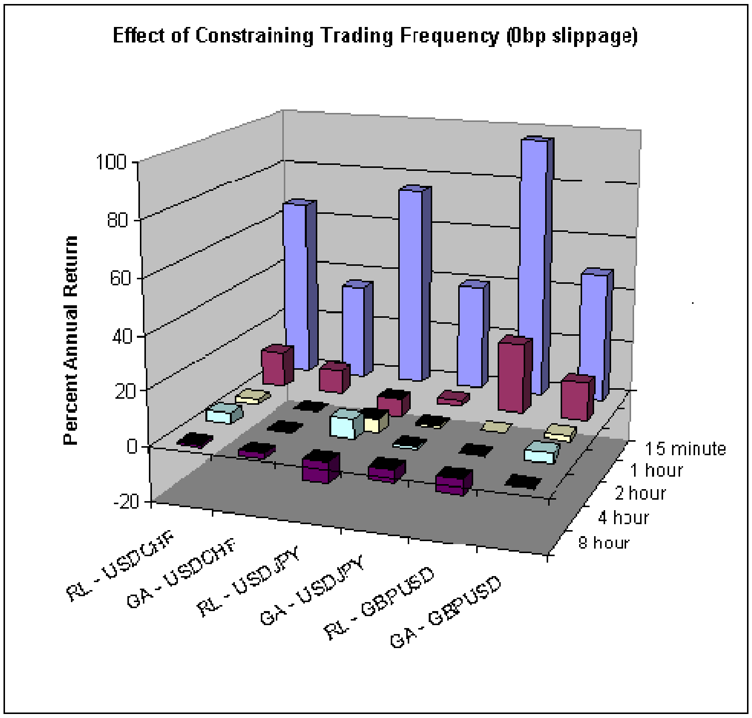

Figure 14: Effect of Trading Frequency (0bp slippage)

largely significant, though not the RL. Similarly at lower frequencies, the results are more volatile and significance is uniformly not present. It was argued earlier however that artificially constraining trading proved to adversely affect profitability and this will have affected these results.

9 Discussion & Future Work

We have shown that significant promise lies in the use of computational learning techniques in high frequency foreign exchange trading. The fact that the techniques investigated here return significantly positive results out-of-sample implies that there is useful information in the technical indicators that can be exploited. This is consistent with the tenets of technical analysis and contradictory to the Efficient Market Hypothesis. Furthermore, the relatively good out-of-sample performance obtained demonstrates that using a combination of technical indicators leads to better performance than using the individual indicators themselves. In fact, Dempster & Jones [5, 8] demonstrate on different data that these indicators are largely

Table 9: p-values (GA vs RL) at 0bp slippage

Trading Frequency	15 minute	1 hour	2 hour	4 hour	8 hour
RL - GBPUSD	**100**	**99.99**	50	63.06	25.25
GA - GBPUSD	**100**	**99.99**	84.13	84.13	30.25
RL - USDCHF	**99.99**	**99.62**	63.05	63.06	24.01
GA - USDCHF	**99.99**	**95.22**	88.49	36.94	50.00
RL - USDJPY	**99.99**	63.06	63.06	90.88	56.81
GA - USDJPY	**99.99**	**99.02**	63.06	84.13	50.00

Table 10: p-values (GA vs RL) at 1bp slippage

Trading Frequency	15 minute	1 hour	2 hour	4 hour	8 hour
RL - GBPUSD	**100**	**99.87**	63.06	74.75	0.98
GA - GBPUSD	**99.99**	**99.62**	74.75	63.06	63.06
RL - USDCHF	**95.22**	74.75	9.12	50	24.01
GA - USDCHF	**95.22**	84.13	**97.03**	63.79	50
RL - USDJPY	90.98	36.94	24.01	74.75	43.19
GA - USDJPY	**99.02**	**97.72**	90.88	93.86	75.99

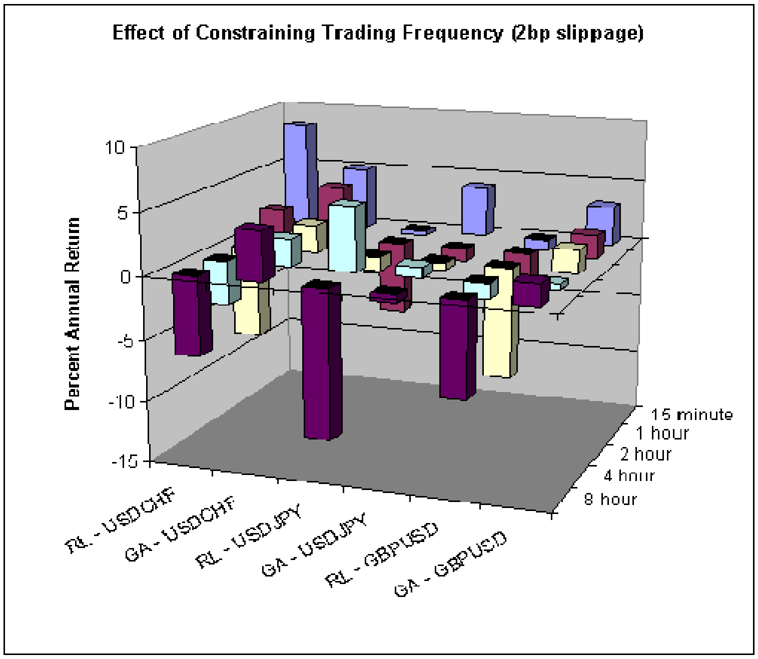

Figure 15: Effect of Trading Frequency (2bp slippage)

unprofitable when considered in isolation.

At low slippage values, annual returns of 10-20% are not uncommon. However these slippage values are only typically available to market makers. Investment managers for example, who more typically face slippage of up to 10bp, would be unable to utilize the methods outlined here in the manner described.

Demonstrated overfitting the data by the RL system leads us to suggest a potential improvement. In order to limit the ability of the RL to overfit the in-sample data, a GA can be used to optimize the choice of underlying indicators. This approach is currently being investigated.

When it came to considering frequency effects though, it was shown that in most cases, it is best to let the computational learning technique determine the trading frequency itself by feeding it the highest frequency market data available.

Optimization of indicator parameters was also considered to convincingly demonstrate that the system performance degenerates due to the increased correlation between the indicators. However the system could also be fed with several inputs of the same indicator with varying parameters. This is a variation of indicator optimization that has not been considered here but which overcomes the issue of increased correlation between the input indicators and

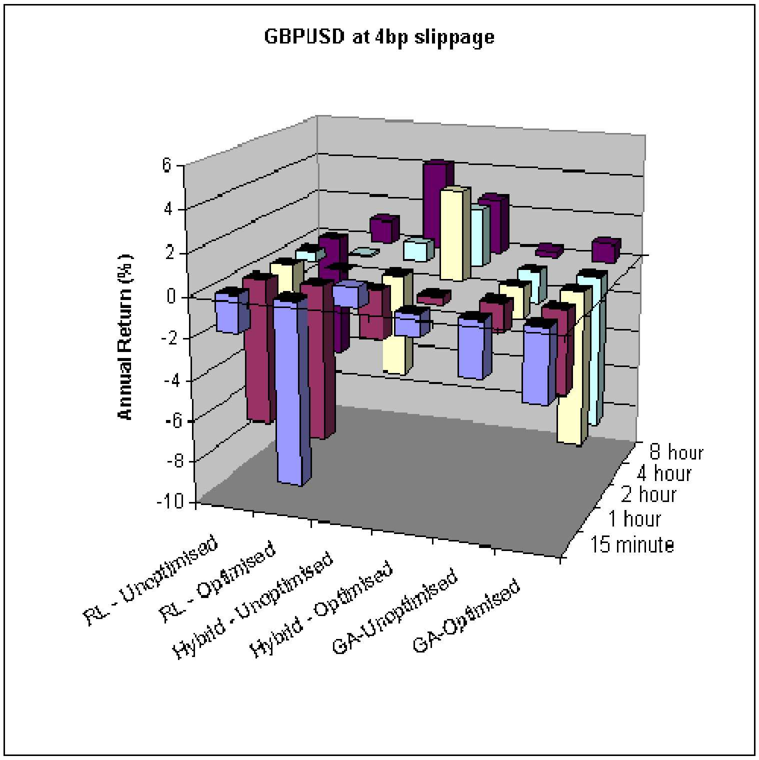

Figure 16: Effect of Trading Frequency (4bp slippage)

is a topic for further research.

A natural way forward however is to consider different trading objective functions [5, 15]. Exploring *risk adjusted* return rather than raw return and overlaying the systems with cash management such as *stop losses* requires them to also consider *neutral* - ie. out-of-the-market - states which they currently do not. Another avenue of current research is the use of alternative reinforcement learning approaches such as the *recurrent reinforcement learning* approach described by Moody [21].

We are currently also exploring the incorporation of trade volume data into the learning algorithms.

References

[1] S. Achelis, *Technical Analysis from A to Z*, McGraw-Hill, New York, 2001.

[2] S. Chen and W. Lee, *Option pricing with gas: A second report*, IEEE International Conference on Neural Networks, 1 (1997), pp. 21–25.

Table 11: Significances (GA vs RL) at 2bp slippage

Trading Frequency	15 minute	1 hour	2 hour	4 hour	8 hour
RL - GBPUSD	9.12	6.14	3.03	84.13	0.67
GA - GBPUSD	90.88	63.06	84.13	63.06	84.13
RL - USDCHF	29.27	90.07	6.68	25.25	5.09
GA - USDCHF	50	80.43	80.43	75.99	81.83
RL - USDJPY	65.54	58.20	8.77	74.75	11.51
GA - USDJPY	84.13	90.88	84.13	84.13	69.65

[3] N. Chidambaran, C. Jevons Lee, and J. Trigueros, *An adaptive evolutionary approach to option pricing via genetic programming*, in Proceedings of the 6th International Conference on Computational Finance 1999, Y. S. Abu-Mostafa, A. S. Weigend, B. LeBaron, and A. W. Lo, eds., vol. 1813, Springer-Verlag, 2000, pp. 583–598.

[4] G. Deboeck, *Trading on the Edge : Neural, Genetic, and Fuzzy Systems for Chaotic Financial Markets*, Wiley, New York, 1994.

[5] M. A. H. Dempster and C. Jones, *A real-time adaptive trading system using genetic programming*, Quantitative Finance, 1 (2001), pp. 397–413.

[6] M. A. H. Dempster and C. M. Jones, *Can technical pattern trading be profitably automated: 1. The channel.* Working Paper No. 11/99, Judge Institute of Management, University of Cambridge, 1999. To appear in European Journal of Finance, (2002).

[7] ——, *Can technical pattern trading be profitably automated: 2. The head and shoulders.* Working Paper No. 12/99, Judge Institute of Management, University of Cambridge, 1999.

[8] ——, *The profitability of intra-day FX trading using technical indicators.* Working Paper No. 35/00, Judge Institute of Management, University of Cambridge, 2000.

[9] M. A. H. Dempster, T. W. Payne, Y. Romahi, and G. Thompson, *Computational learning techniques for intraday FX trading using popular technical indicators*, IEEE Transactions on Neural Networks, Special Issue on Computational Finance, 12 (2001), pp. 744–754.

[10] B. Flower, T. Cripps, M. Jabri, and A. White, *An artificial neural network based trade forecasting system for capital markets*, Proceedings of the 4th International Conference on Neural Networks in the Capital Markets, (1995).

[11] R. Gencay, M. Dacorogna, U. Muller, R. Olsen, and O. Pictet, *An Introduction to High-Frequency Finance*, Academic Press, 2001.

[12] C. Goodhart and R. Curcio, *When support/resistance levels are broken, can profits be made? evidence from the foreign exchange market.* Discussion Paper 142, London School of Economics, July 1992.

[13] C. Goodhart and M. O'Hara, *High frequency data in financial markets: Issues and applications*,

Journal of Empirical Finance, 4 (1997), pp. 73–114.

[14] J. H. Holland, *Adaptation in natural and artificial systems*, University of Michigan Press, Ann Arbour, MI, 1975.

[15] C. Jones, *Automated technical foreign exchange trading with high frequency data*, PhD thesis, Centre for Financial Research, Judge Institute of Management Studies, University of Cambridge, June 1999.

[16] J. R. Koza, *Genetic programming: On the programming of computers by means of natural selection*, MIT Press, Cambridge, MA, 1992.

[17] B. LeBaron, *Technical trading rule profitability and foreign exchange intervention*, Journal of International Economics, 49 (1999), pp. 124–143.

[18] R. Levich and L. Thomas, *The significance of technical trading rule profits in the foreign exchange market: A bootstrap approach*, Journal of International Money and Finance, 12 (1993), pp. 451–474.

[19] A. W. Lo and A. C. MacKinlay, *A Non-Random Walk Down Wall Street*, Princeton University Press, Princeton, NJ, 1999.

[20] P. Maes and R. Brooks, *Learning to coordinate behaviors*, Proceedings of the 8th National Conference on Artificial Intelligence, (1990), pp. 796–802.

[21] J. Moody, L. Wu, Y. Liao, and M. Saffell, *Performance functions and reinforcement learning for trading systems and portfolios*, The Journal of Forecasting, 17 (1998), pp. 441–470.

[22] C. Neely and P. Weller, *Intraday technical trading in the foreign exchange market.* Working Paper 99-016A, Federal Reserve Bank of St. Louis, November 1999.

[23] C. J. Neely, *Technical analysis in the foreign exchange market: A layman's guide*, in Federal Reserve Bank of St. Louis Review, September/October 1997, pp. 23–38.

[24] C. J. Neely, P. A. Weller, and R. Dittmar, *Is technical analysis in the foreign exchange market profitable? A genetic programming approach*, Journal of Financial and Quantitative Analysis, 32 (1997), pp. 405–426. Also available as Federal Reserve Bank of St. Louis Working Paper 96-006C.

[25] R. Neuneier, *Optimal asset allocation using adaptive dynamic programming*, in Advances in Neural Information Processing Systems, D. S. Touretzky, M. C. Mozer, and M. E. Hasselmo, eds., vol. 8, The MIT Press, 1996, pp. 952–958.

[26] ———, *Enhancing Q-learning for optimal asset allocation*, in Advances in Neural Information Processing Systems, M. I. Jordan, M. J. Kearns, and S. A. Solla, eds., vol. 10, The MIT Press, 1998.

[27] C. L. Osler and P. H. K. Chang, *Methodical madness: Technical analysis and the irrationality of exchange-rate forecasts*, Economic Journal, 109 (1999), pp. 636–661.

[28] O. V. Pictet, M. M. Dacorogna, B. Chopard, M. Oudsaidene, R. Schirru, and M. Tomassini, *Using genetic algorithm for robust optimization in financial applications*, Neural Network World, 5 (1995), pp. 573–587.

[29] P. Refenes, A. Burgess, and Y. Bentz, *Neural networks for financial engineering: A study on methodology*, IEEE Transactions on Neural Networks, 8 (1997).

[30] P. Samuelson, *Proof that properly anticipated prices fluctuate randomly*, Industrial Management Review, 6 (1965), pp. 41–49.

[31] R. S. Sutton and A. G. Barto, *Reinforcement learning: an introduction*, The MIT Press, 1998.

[32] G. W. Tan, *Topics in foreign exchange trading systems.* MPhil Thesis, Centre for Financial Research, University of Cambridge, June 1999.

[33] M. P. Taylor and H. Allen, *The use of technical analysis in the foreign exchange market*, Journal of International Money and Finance, 11 (1992), pp. 304–314.

[34] C. Watkins, *Learning from Delayed Reward*, PhD thesis, Kings College, University of Cambridge, 1989.

[35] H. Zimmermann and R. Neuneier, *Active portfolio-management with neural networks*, in Proc. of Computational Finance 1999, Y. S. Abu-Mostafa, A. S. Weigend, B. LeBaron, and A. W. Lo, eds., vol. 1813, Springer-Verlag, 2000.