Chapter 8

ALGORITHMIC TRADING WITH DEVELOPMENTAL AND LINEAR GENETIC PROGRAMMING

Garnett Wilson[1] and Wolfgang Banzhaf[1]
[1]*Memorial University of Newfoundland, St. John's, NL, Canada.*

Abstract A developmental co-evolutionary genetic programming approach (PAM DGP) and a standard linear genetic programming (LGP) stock trading system are applied to a number of stocks across market sectors. Both GP techniques were found to be robust to market fluctuations and reactive to opportunities associated with stock price rise and fall, with PAM DGP generating notably greater profit in some stock trend scenarios. Both algorithms were very accurate at buying to achieve profit and selling to protect assets, while exhibiting both moderate trading activity and the ability to maximize or minimize investment as appropriate. The content of the trading rules produced by both algorithms are also examined in relation to stock price trend scenarios.

Keywords: Developmental Genetic Programming, Linear Genetic Programming, Computational Finance

1. Introduction

Algorithmic trading examines a stock's past price movements in order to anticipate what effect they will have on its future price. Such analysis uses technical indicators like price fluctuations and trading volume to identify these changes in an asset's price. Evolutionary Computation techniques, such as genetic programming (GP), have been applied to the analysis of financial markets with a reassuring degree of success (Brabazon and O'Neill, 2006). This chapter explores the application of a developmental GP system, Probabilistic Adaptive Mapping Developmental Genetic Programming (PAM DGP), and linear genetic programming (LGP), to interday stock trading. PAM DGP uses co-operative co-evolution of genotype solutions and genotype-phenotype map-

R. Riolo et al. (eds.), *Genetic Programming Theory and Practice VII*,
Genetic and Evolutionary Computation, DOI 10.1007/978-1-4419-1626-6_8,

pings and permits emphasis of certain functions over others, while LGP uses a single genotype population and the encoding of functions is static.

The following section discusses previous GP-related approaches to stock market analysis. Section 3 describes the stock trading implementations of LGP and PAM DGP, including function set and interpretation of trading rules as genotype individuals. The ability of both algorithms to generate profit when applied to a number of stocks across market sectors is examined in Section 4. Section 5 examines the general trading activity and its profitability for the two algorithms. Section 6 presents an analysis of the actual function set content of the trading rules. Conclusions and future work follow in Section 7.

2. Related Approaches to Stock Prediction

Genetic programming is pervasive in the field of financial analysis , and a number of implementations are described in the literature. The system described here was first introduced in (Wilson and Banzhaf, 2009). In this work, we examine a much more substantial variety of stocks and price trends, and also the trading rules generated by the different implementations. The first implementation we consider is traditional linear GP (Brameier and Banzhaf, 2007). LGP has been applied to stock market analysis previously by (Grosan and Abraham, 2006), where a LGP hybrid (with multi-expression systems) outperformed neural networks and neuro-fuzzy systems for interday prediction of stock prices for NASDAQ and Nifty indices. The second implementation we examine is PAM DGP, a co-evolutionary developmental approach. While the authors are not aware of any previous approaches to stock market analysis using developmental approaches, a co-evolutionary process has been applied to the evolution of trading rules by (Drezewski and Sepielak, 2008). In their co-evolutionary system, one species represented entry strategies and one species represented exit strategies. Both a multi-agent version of the co-evolutionary algorithm and an evolutionary algorithm were created, where the multi-agent co-evolutionary approach generated the most profit.

In terms of the application of the GP algorithms to interday trading, some modified elements of the grammatical evolution (GE) approach of (Brabazon and O'Neill, 2006) were adopted. In their approach, after initial evolution during a training period, the best rules in the population traded live for a window of n days. The training window then shifted and the current population was retrained using the data in the window on which it was previously trading live. The algorithm then traded live on the following n days, and so on. The authors compared two versions of the GE system, one where the final population from the last window was used as the starting population for the current window, and one that re-initialized the population with each window shift. The authors found that maintaining the population, rather than re-initializing it, provided more profitable performance (and better rules). Similarly, our populations were not re-initialized with each window shift. Our technique uses a shifting window of length 5 days, but shifts only in increments of 1 day.

3. LGP and PAM DPG Algorithm for Stock Analysis

LGP is a very popular form of genetic programming, where instead of the most traditionl form of trees being used as individuals, genotypes consist of binary strings and registers to store subresults. These binary strings are interpreted as instructions of a program, where a unique binary sequence encodes for only one member of the function set. Throughout program execution in standard LGP, the mapping of binary sequence to instruction does not change.

In PAM DGP (Wilson and Heywood, 2007), there is a population of genotypes that cooperatively coevolves with a separate population of mappings. A probability table is updated throughout algorithm execution with entries corresponding to each pairing of individual genotype and mapping from both populations. The table entries represent frequencies that dictate the probability that roulette selection in a steady state tournament will choose the genotype-mapping pairing of individuals determined by the indices of the table. The genotype and mapping individual that are members of the current best genotype-mapping pairing are immune to mutation and crossover to maintain the current best solution discovered. Each tournament round involves the selection of four unique genotype-mapping pairings. Following fitness evaluation and ranking, the probability table columns associated with the winning combinations have the winning combination in that column updated using Equation (8.1) and the remaining combinations in that column updated using Equation (8.2)

$$P(g,m)_{new} = P(g,m)_{old} + \alpha(1 - P(g,m)_{old}) \tag{8.1}$$

$$P(g,m)_{new} = P(g,m)_{old} - \alpha(P(g,m)_{old}) \tag{8.2}$$

where g is the genotype individual/index, m is the mapping individual/index, α is the learning rate (corresponding to how much emphasis is placed on current values versus previous search), and P(g,m) is the probability in table element [g, m]. To prevent premature convergence, the algorithm uses a noise threshold. If an element in the table exceeds the noise threshold following a tournament round, a standard Gaussian probability in the interval [0, 1] is placed in that element and all values in its column are re-normalized so the column elements sum to unity. The PAM DGP algorithm and selection mechanism are summarized in Figure8-1.

Genotypes in PAM DGP are binary strings, with interpretation of sections of the binary string being instruction-dependent. Mappings in this work are redundant such that individuals are composed of $b \geq s$ 10-bit binary strings, where b is the minimum number of binary sequences required to represent a function

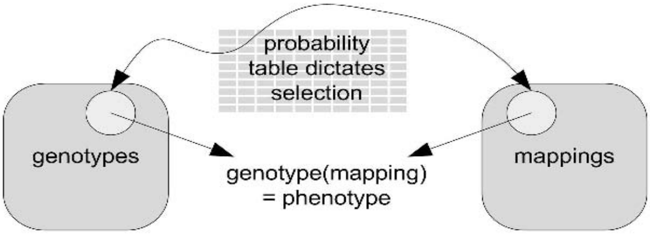

Figure 8-1. Probabilistic Adaptive Mapping Developmental Genetic Programming (PAM DGP).

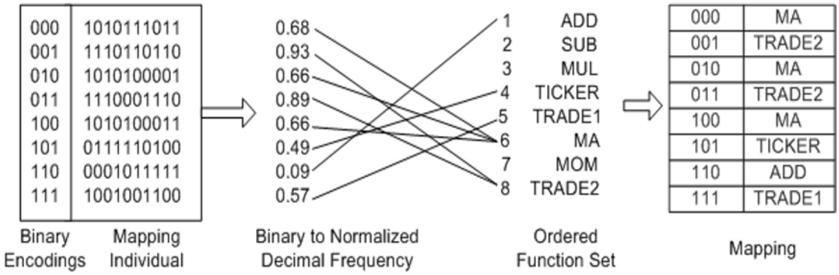

Figure 8-2. PAM DGP mapping process.

set of s symbols. Each 10 bit mapping section is interpreted as its decimal equivalent, normalized to the range [0, 1], and mapped to an ordered function set index by multiplying by s and truncating to an integer value (allowing redundant encoding of symbols). The process whereby a particular mapping is used to isolate and emphasize certain members of the function set when interpreting genotype is depicted in Figure 8-2. Using this mapping mechanism with co-evolutionary selection, PAM DGP will emphasize the most useful members of the function set, ignore members of the function set which are not pertinent, and simultaneously evolve an appropriate genotype solution.

PAM DGP is compared to the standard LGP implementation (Brameier and Banzhaf, 2007) in this study. LGP individuals are also bit strings, and there is naturally only a genotype population. The interpretation of instructions for PAM DGP is the same for LGP, where LGP here can be seen as a special case of PAM DGP that uses a static mapping and constant function set. (PAM DGP extends LGP such that members of a function set are adaptively emphasized.) Additional details of PAM DGP, along with its original motivations and comparisons to related developmental systems, are available in (Wilson and Heywood, 2007).

The PAM DGP and LGP implementations are applied to several stocks across market segments, including technology: Google Inc. (GOOG), Apple Inc. (AAPL), and Microsoft Corporation (MSFT), energy: Chevron Co. (CVX) and Ballard Power Systems (BLDP), consumer: PepsiCo Inc. (PEP), automobile: Ford Motor Co. (F), and finance: Bank of Montreal (BMO). The initial exchange portion of the ticker symbols were removed in all cases for brevity. High, low, open, and close data was provided as input for 200 day periods throughout 2007 and 2008, with different dates chosen to test the implementations' performance. The first 16 days of the 200 days were reserved as a basis on which to draw initial data for the technical indicators. After those dates, GP fitness was evaluated on data corresponding to a moving window of n days. Individuals represent sets of trading rules, based on functions in the function set. For each window of n days corresponding to trading days m to n, each of m to n - 1 days were used for calculation of a trading decision given the individual's rule set, with $m + 1$ to n being used to evaluate the recommendation based on the immediately preceding day. Days used for the calculation of a trading decision were normalized using two-phase preprocessing as in (Brabazon and O'Neill, 2006): All daily values were transformed by division by a lagged moving average, and then normalized using linear scaling into the range [0, 1] using

$$v_{scaled} = \frac{v_t - l_n}{h_n - l_n} \tag{8.3}$$

where v_{scaled} is the normalized daily trading value, v_t is the transformed daily trading value at time step t, h_n is highest transformed value in the last n time steps, l_n is the lowest transformed value in the last n time steps, and n is length of the time lag chosen for the initial transformation.

In addition to an instruction set, each individual consists of a set of four registers, a flag for storing the current value of logical operations, and a separate output (trade) register for storing a final value corresponding to a trade recommendation. Following the execution of the trading rules of a GP individual, if the value of the trade register is 0, no action is recommended. Otherwise, the final value in the trade register corresponds to a value in the range [0, 1]. This value was multiplied by a maximum dollar amount to be bought or sold per trade ($10,000 was used here based on an initial account balance of $100,000 with which to trade) to give some portion of $10,000 to be traded. For each trade conducted, there is a $10 commission penalty. The trading system is permitted to run a small deficit >= $10 to either handle a sell recommendation when maximally invested (where the deficit would be immediately recouped) or, similarly, to allow a buy in order to be maximally invested. Fitness of an individual is the value of the cash and shares held.

The best individual consisting of the best trading rule set is used by a “live” trading algorithm. That is, the live trader provides known information to the GP for days *m* to *n*. The GP algorithm returns a recommendation on which the live trading system bases its decision to trade on the following day, *n + 1*. In particular, the net number of shares bought and sold by the best evolved individual (trading rules) given the recommendation of the trade register over all the fitness cases is the buy or sell recommendation to the “live” trading system. The transactions of the live trading system are thus based on unknown data, and determine the success of the algorithms.

While PAM DGP uses co-evolution to refine function set composition, the appropriate initial function set members must be provided as a basis upon which the algorithm can select its optimum function set. In the case of standard GP, this initial function set remains constant throughout execution. The function set includes standard mathematical operators, and instructions to trade based on logical operators applied to the four internal registers. In addition, there are established financial analysis metrics used: moving average, momentum, channel breakout, and current day high, low, open, or close price. The financial technical indicator moving average (MA) is the mean of the previous *n* share prices. The momentum indicator (MOM) provides the rate of change indicator, and is the ratio of a particular time-lagged price to the current price. Channel breakout (BRK) establishes a trading range for a stock, and reflects its volatility. The most popular solution places Bollinger bands around a *n*-day moving average of the price at $\pm$ 2 standard deviations of the price movement.

4. Profit Analysis

The worth of the live trading system’s assets over 184 days of trading is examined (200 fitness cases in total were actually used, with the first 16 reserved to provide initial values for technical indicators). Fifty trials over the 184 trading days were conducted for the four stocks using an Apple iMac Intel Core 2 Duo 2.8 GHz CPU and 4GB RAM using OS X Leopard v10.5.4. With an initial account of $100,000, the mean worth (with standard error) of all assets (cash and shares) of the live trading system for PAM DGP, LGP, and naive buy-and-hold strategies is given in Figure 8-3. Naive buy-and-hold is simply the strategy of maximally investing on the first trading day and staying invested for the entire time period.

It is evident from Figure 8-3 that PAM DGP and LGP are both robust to share price fluctuations (where the buy and hold trend line is a direct indication of share price fluctuations). The evolved solutions seem to take advantage of the upward trends, although the solutions reflect a conservative strategy overall, adept at anticipating and buffering against sharp share price declines and volatility in general. In terms of specific upward trends, GOOG and AAPL exhibit moderately volatile behavior followed by fairly sharp declines (from approximately day 60 to 80), proceeded by climbing stock prices. In both of these instances, LGP and PAM DGP outperform buy-and-hold, with PAM DGP outperforming LGP. Given PEP and CVX, a general upward climbing trend can be examined. Here, the performance of LGP and PAM DGP is much closer. While the more steady upward climb of PEP does not allow LGP or PAM DGP to outperform buy-and-hold, the price drops of CVX between days 60 and 120 allow PAM DGP and LGP to outperform buy-and-hold by the end of the time period. Note that in the instance of PEP, both algorithms are naturally not invested to the (maximal) extent of buy-and-hold prior to the steep price climb, and thus have less final profit (but are still competitive and almost as profitable).

In terms of the ability of the algorithms to handle downward market trends, BLDP and MSFT show downward trending stock prices. MSFT exhibits a fairly gradual downward trending slope. PAM DGP and LGP perform relatively on par with buy-and-hold, with all implementations generating comparable losses at the end of the time period. BLDP, in contrast, features some volatility with spiking near the end. This volatility allows LGP and PAM DGP to end the time period with greater profit than buy-and-hold (although all algorithms suffer losses when investing in these downward trending stocks). While we see in all stocks in Figure 8-3 that LGP and PAM DGP are typically able to recognize steep downward trends, and sell assets to protect investments, general downward trends with consistent moderate volatility (see MSFT especially) can prevent the algorithm from pulling out assets to a large degree because there are brief episodes of profit. F and BMP exhibit an upward trend, followed by punctuated steeper downward trends. During the downward trends, LGP and PAM DGP will typically sell to protect investments (although PAM DGP does get caught in a very sudden drop at the end of the time period for BMO). Final and cumulative measures of profit are shown in Figures 8-4 and 8-5, respectively.

In the boxplots of Figures 8-4 and 8-5, each box indicates the lower quartile, median, and upper quartile values. If the notches of two boxes do not overlap, the medians of the two groups differ at the 0.95 confidence interval. Points represent outliers to whiskers of 1.5 times the interquartile range. In Figure 8-4, a comparison of final profit indicates that stocks that were well-chosen (were profitable overall during the time period) generated profit for both algorithms. Note that time period end is arbitrary and profits are a direct reflection of underlying market trend. If a stock is losing value, direct buying and selling of the stock cannot generate profit.

Figure 8-5 is more informative, as it shows the mean daily cumulative profit (%) greater than buy-and-hold for the LGP and PAM DGP live trading systems over all trading days. Both PAM DGP and LGP were generally more profitable than buy-and-hold at any given time for all stocks. Exceptions included, naturally, the case of PEP where naïve buy-and-hold is a very good strategy, and

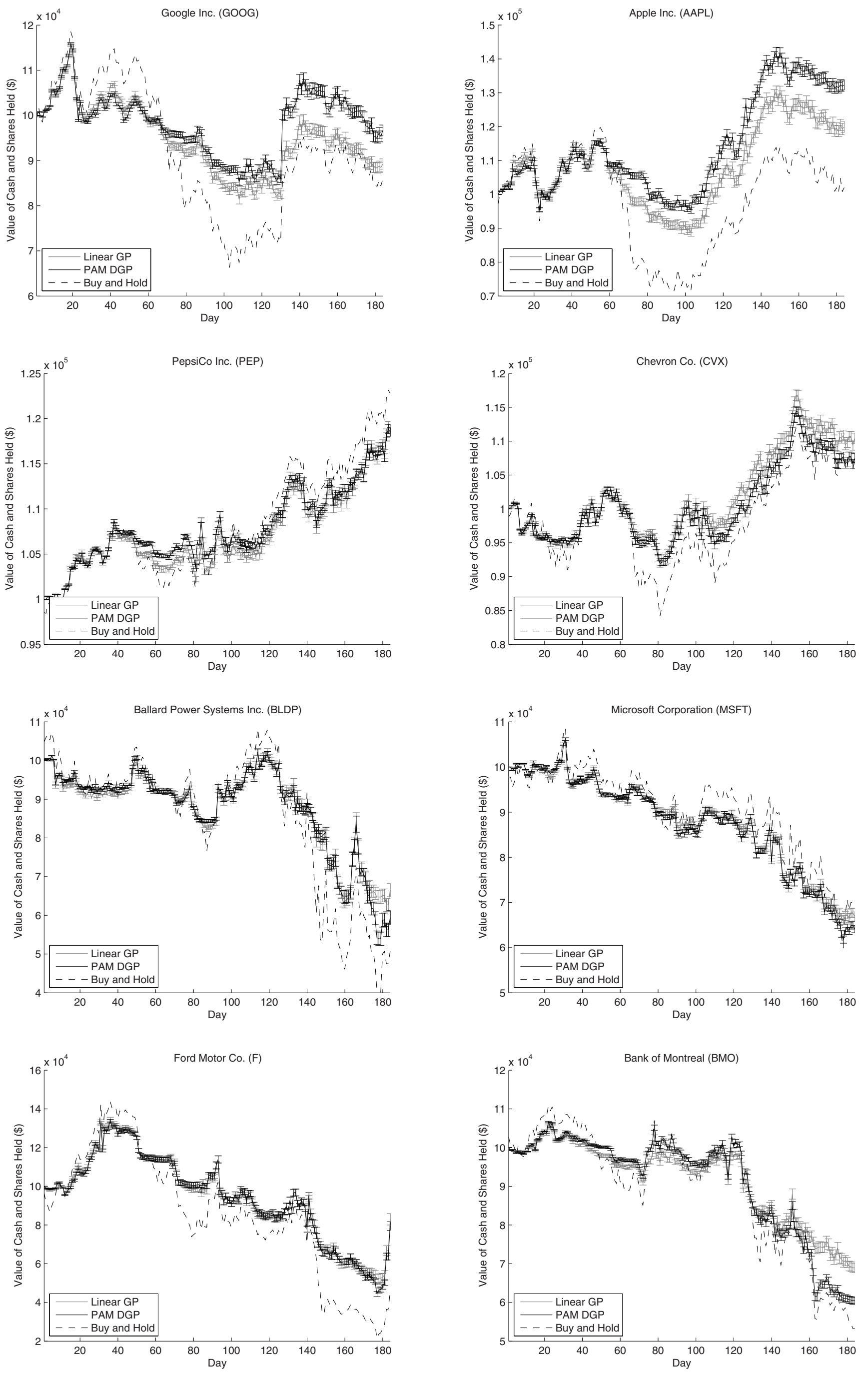

Figure 8-3. Mean total worth (value of cash and shares) for PAM DGP, LGP, and buy-and-hold strategies over 50 trials with standard error given initial $100,000 cash value.

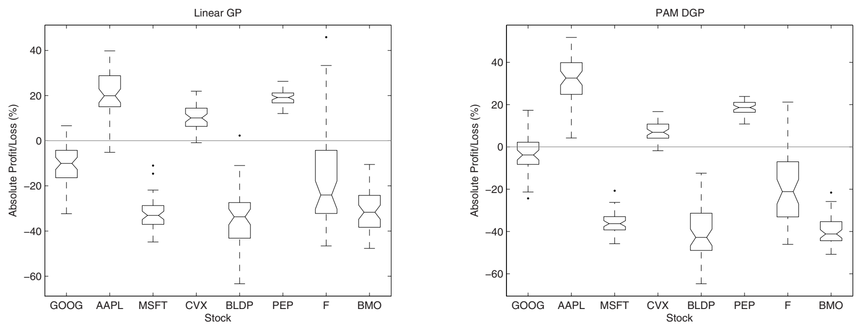

Figure 8-4. Boxplot of mean final profit (%) greater than buy-and-hold for PAM DGP and LGP over 50 trials. Value of 0 indicates the breakeven point.

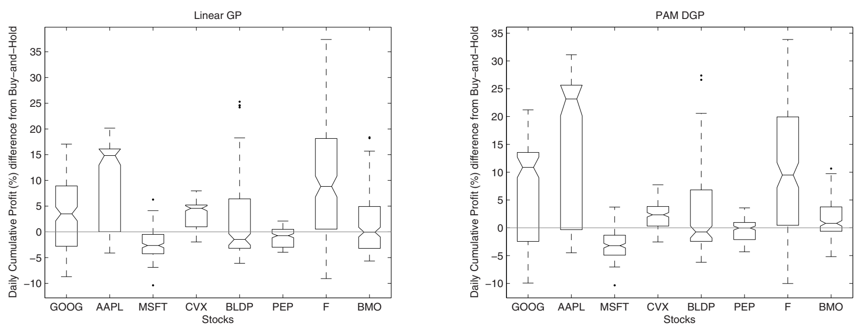

Figure 8-5. Boxplot of mean daily cumulative profit (%) difference from buy-and-hold for PAM DGP and LGP over 50 trials. Value of 0 indicates the breakeven point.

BLDP and MSFT where high volatility combined with a drawn out downward trend caused losses. In all cases where buy-and-hold was cumulatively more profitable, the performance was only lower for LGP or PAM DGP by a very slight amount (0–1% for BLDP and PEP, less than 5% for MSFT). PAM DGP was more profitable than LGP at any given time by a large margin for GOOG and AAPL and a very small margin for PEP. LGP slightly outperformed PAM DGP for CVX. Given the behavior in Figures 8-3 and 8-5, PAM DGP provides increased robustness to market downturns and quickly takes advantage of growth opportunities later in evolution. Also, we can see in Figure 8-5 that LGP slightly outperforms PAM DGP for CVX by not selling quite as much stock during a market dip immediately preceding a steady climb starting at approximately day 100 (Figure 8-3). Thus PAM DGP is slightly more reactive in its selling to prevent loss, where this benefits performance for GOOG and AAPL, but not CVX. There was no substantial statistically significant difference in cumulative profit for the other stocks.

5. Trading Activity

Trading activity is shown in Figure 8-6, expressed as the number of shares retained daily as a percentage of the live system's total worth. Comparing Figures 8-3 and 8-6, it is evident that both PAM DGP and LGP are capable of efficiently reacting to the market: they will both sell if a stock price starts to drop and buy if the stock price appears to be rising. Figures 8-3 and 8-6 collectively show that both algorithms will stay maximally invested during sustained profitable periods.

The performance of these trades can be further examined by analysis of how many trades were conducted and their benefit. Proportion of profitable trades is a common metric for evaluation of trading activity, although it can be deceptive: it does not even reflect the overall ability of an algorithm in terms of actual profit generated (Brabazon and O'Neill, 2006). Many trades are beneficial in preventing loss during market downturns, and generate no profit at all. Thus, rather than the standard measure of percentage of profitable trades, the percentage of profitable buy trades and percentage of sell trades preventing loss for each algorithm are given in Figures 8-7 and 8-8, respectively. Figure 8-9 shows the percentage of trading opportunities where a trade was actually conducted. The number of trading opportunities not taken when the system was maximally or minimally invested, out of all possible trades, is shown in Figure 8-10. Figure 8-7 reveals that both LGP and PAM DGP are very accurate when buying for profit: LGP exhibited medians of 96–100% profitable buys across all stocks, and PAM DGP exhibited 87% to 100% profitable buys across all stocks (with the vast majority above 96%). Figure 8-8 shows that LGP was extremely good at selling to prevent loss; all medians were 100%. PAM DGP did not perform quite as well, but still exhibited very impressive results by selling to prevent loss with 94–100% accuracy. Overall, both algorithms were very good at both buying for profit and selling to prevent loss. Even outliers in either buying for profit or selling to prevent loss were acceptably high percentages.

Figure 8-9 shows the trading activity behind all the performance measures we have considered so far. PAM DGP generally conducted more trades (based on spread of data) than LGP for all stocks. For all stocks with a general upward trend (GOOG, AAPL, CVX, and PEP), a lower number of trades were conducted for both LGP and PAM DGP. In particular, LGP conducted approximately 28–35% (based on median) of possible trades for (generally) rising stocks, while approximately 37–42% of possible trades were conducted for the (generally) falling stocks (MSFT, BLDP, F, BMO). PAM DGP conducted approximately 30–40% of available trades for rising stocks and 44–50% for falling

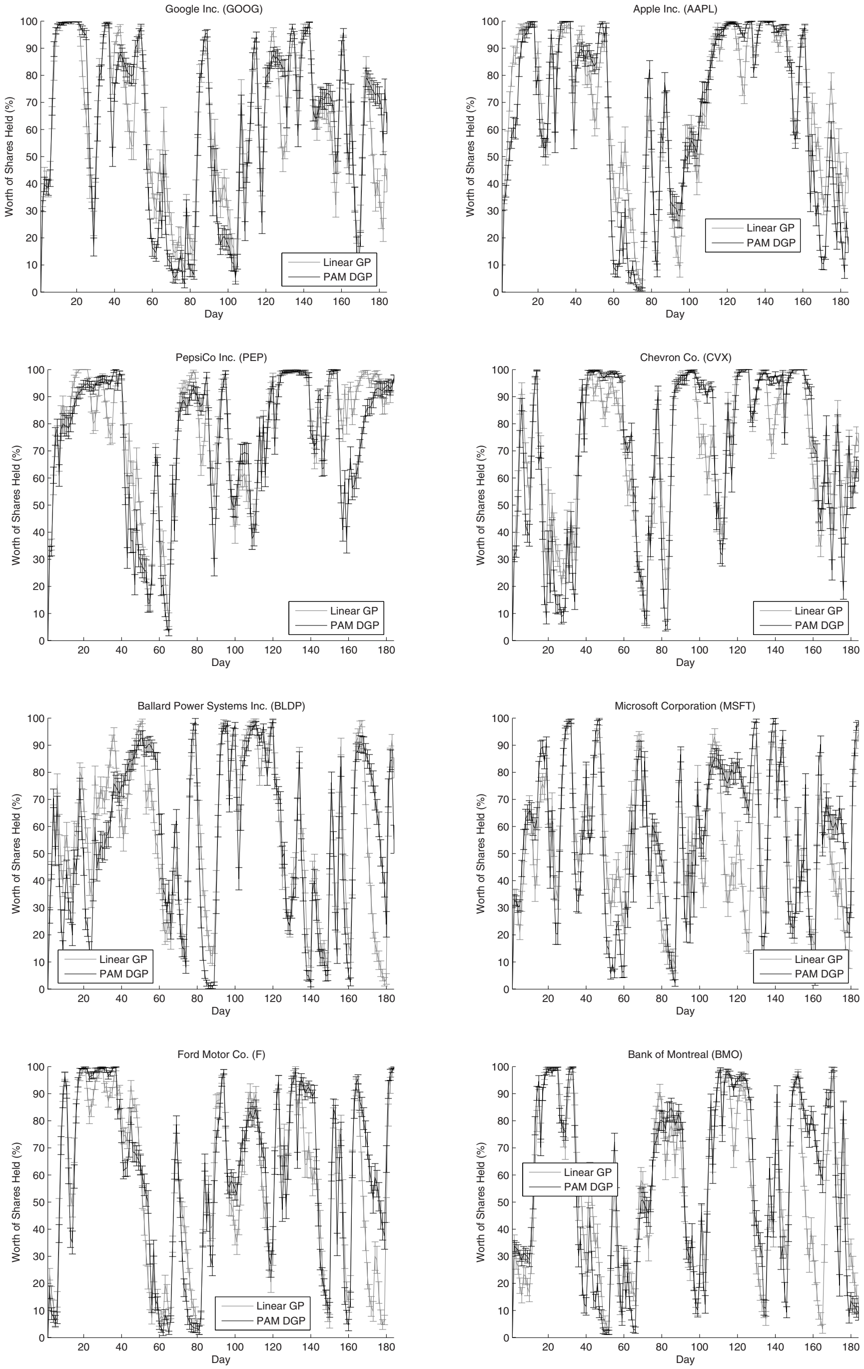

Figure 8-6. Mean shares held by PAM DGP (black line) and LGP (grey line) live trading systems as a percentage of total worth over 50 trials with standard error.

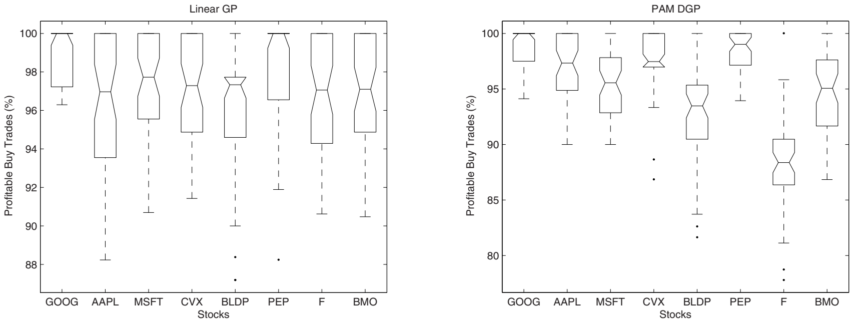

Figure 8-7. Percentage of profitable buy trades for 184 trading days over 50 trials.

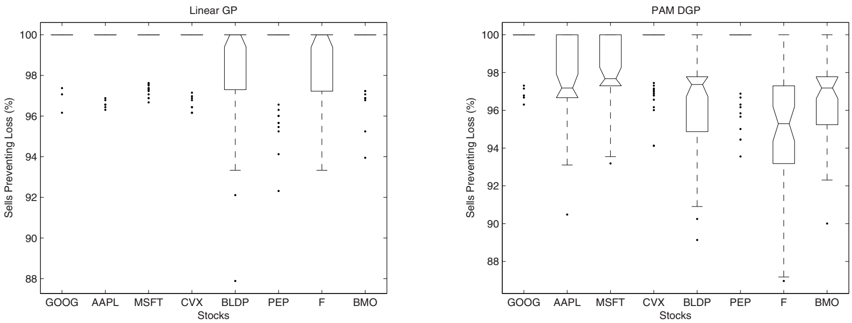

Figure 8-8. Percentage of sell trades preventing losses for 184 trading days over 50 trials.

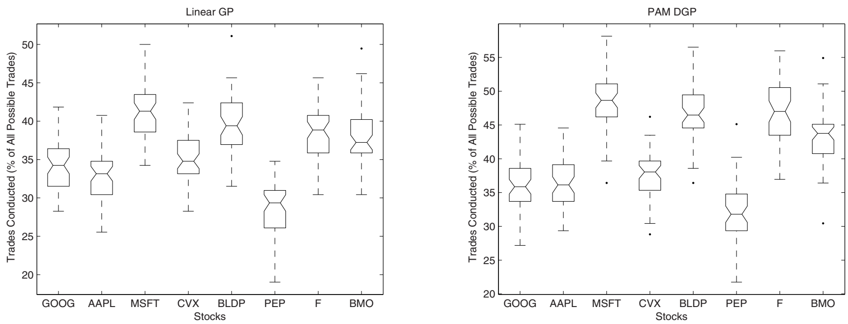

Figure 8-9. Percentage of trades executed overall for each stock for 184 trading days over 50 trials.

stocks. Overall, the groups of falling stocks caused both algorithms to trade more actively than they would for the rising stocks, where this was statistically

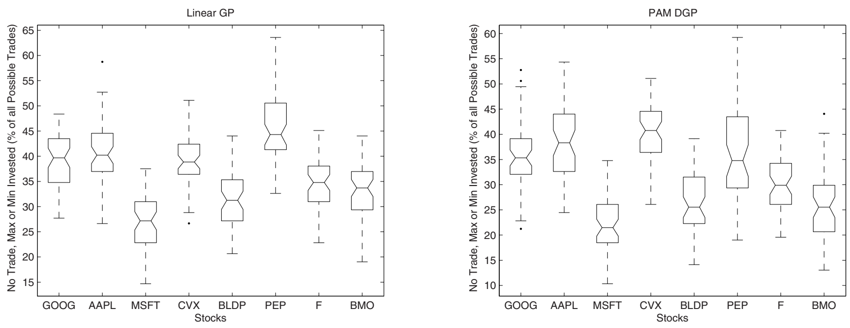

Figure 8-10. Trades not conducted while maximally or minimally invested as a percentage of all trades for 184 trading days over 50 trials.

significant for both LGP and PAM DGP. Figure 8-10 indicates the percentage of all trades where the system wished to maintain a maximally or minimally invested position. For both LGP and PAM DGP, the system would maximize (or minimize) investment for all rising stocks between approximately 35 and 45% (based on median) of the time for rising stocks. Compared with Figure 8-6, it is evident that most of these positions were maximal investment to generate profit. However, again for both algorithms, the system would maximize or minimize investment for only approximately 20 to 35% (median) of the time for falling stocks. Overall, Figures 8-7 to 8-10 indicate that the proportion of beneficial trades (generating profit or protecting the investor from further losses) was impressive, where this occurred in the context of moderate levels of trading.

6. Trading Rules

The actual content of the trading rules will vary between trading days and across general stock price trends. Since populations are kept across training windows, as recommended in (Brabazon and O'Neill, 2006) (see Section 3), the content of an individual at the arbitrary end of a time period is a reflection of trading rules for that stock built up over the entire time period. Thus, we examine the proportion of each member of the function set in the final best individual at the end of the time period over all 50 trials. The composition of individuals for two stock price trend types are examined: a rising stock (AAPL) that achieved profit and a falling stock (BMO) that suffered losses. The percentage of each function set member in the final individuals over all 50 trials is provided for AAPL and BMO in Figure 8-11 and 8-12, respectively. Standard mathematical operators, a logical operator (*logical*), moving average (*ma*), momentum (*mom*), a measure of the turbulence (*trb*), measures based on stock ticker data (*ticker*), and different trading mechanisms (*trade1* to *trade4*) are shown.

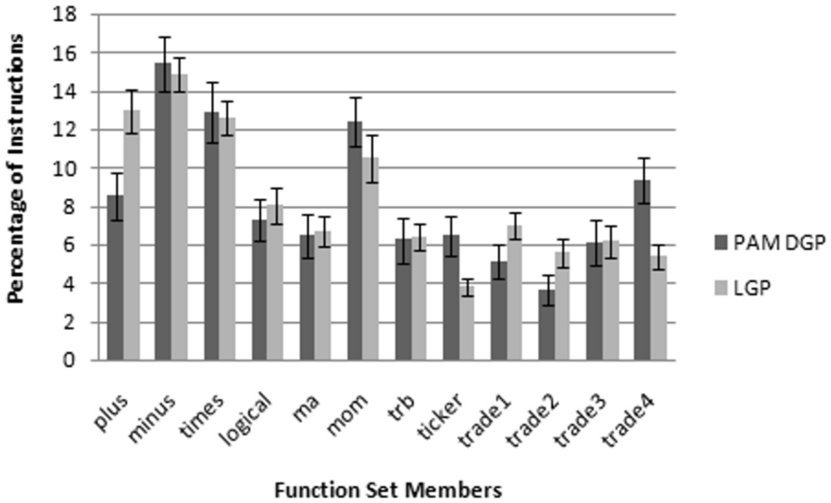

Figure 8-11. Percentage composition corresponding to function set members in final day trading rules after 184 trading days over 50 trials for AAPL.

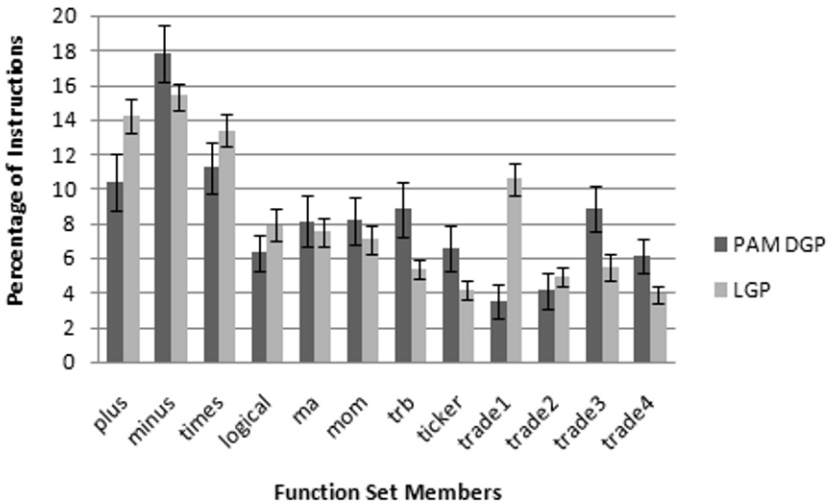

Figure 8-12. Percentage composition corresponding to function set members in final day trading rules after 184 trading days over 50 trials for BMO.

We can note from Figure 8-11 that there are only statistically significant differences in the trading rules of LGP and PAM DGP for 5 of the 12 function set members, but never by a margin of more than 5%. In terms of interesting quantitative measures, momentum analysis dominates the instruction sets for the dropping followed by rising stock (AAPL) where profits were greatest. For the mean number of instructions in the sets over all trials for the falling BMO stock in Figure 8-12, there is a more even distribution of instruction types. However, PAM DGP does provide greater emphasis on certain instructions that execute trades (*trade3* and *trade4*). As with AAPL, only 5 of the 12 function set members differ statistically for BMO between LGP and PAM DGP, but never by more than 5% (with the exception of *trade1*). Overall, there appears to be no substantial difference in proportional function set composition seen in Figures 8-11 and 8-12, averaging content within all final instructions sets.

7. Conclusions and Future Work

This work examined the trading performance of a co-evolutionary developmental GP model (PAM DGP) using a genotype-phenotype mapping and more traditional LGP on eight stocks across market sectors. Both implementations were found to be robust to stock price fluctuations, and outperformed naïve buy-and-hold strategies in almost all scenarios (with the exceptions of steady rise, where buy-and-hold cannot be beaten, and volatile moderate downturn). Even for a stock with a steady rise in price, LGP and PAM DGP are still very competitive and a less risky strategy for shorter time periods than buy-and-hold. Both algorithms evolved so that they protected investments during price drops with impressive accuracy, and they very accurately made buying decisions to generate profit. The beneficial trades by both algorithms were conducted with moderate trading activity and periods of maximal investment to capitalize on rising stock prices. Analysis of trading rules for two chosen stock trends showed that, overall, both algorithms picked similar levels for the majority of functions over all trials. Future work will examine index trading, intraday trading, incorporation of additional quantitative metrics, and extension of the algorithms for trading portfolios of multiple stocks.

Acknowledgment

We would like to thank the reviewers for their helpful comments, especially Michael Korns and Gregory Hornby.

References

Brabazon, Anthony and O'Neill, Michael (2006). *Biologically Inspired Algorithms for Financial Modelling*. Natural Computing Series. Springer.

Brameier, Markus and Banzhaf, Wolfgang (2007). *Linear Genetic Programming*. Number XVI in Genetic and Evolutionary Computation. Springer.

Drezewski, Rafal and Sepielak, Jan (2008). Evolutionary system for generating investment strategies. In Giacobini, Mario, Brabazon, Anthony, Cagnoni, Stefano, Caro, Gianni Di, Drechsler, Rolf, Ekárt, Anikó, Esparcia-Alcázar, Anna, Farooq, Muddassar, Fink, Andreas, McCormack, Jon, O'Neill, Michael, Romero, Juan, Rothlauf, Franz, Squillero, Giovanni, Uyar, Sima, and Yang, Shengxiang, editors, *EvoWorkshops*, volume 4974 of *Lecture Notes in Computer Science*, pages 83–92. Springer.

Grosan, Crina and Abraham, Ajith (2006). Stock market modeling using genetic programming ensembles. In Nedjah, Nadia, de Macedo Mourelle, Luiza, and Abraham, Ajith, editors, *Genetic Systems Programming*, volume 13 of *Studies in Computational Intelligence*, pages 131–146. Springer.

Wilson, Garnett and Banzhaf, Wolfgang (2009). Prediction of interday stock prices using developmental and linear genetic programming. In Giacobini, Mario, De Falco, Ivanoe, and Ebner, Marc, editors, *Applications of Evolutionary Computing, EvoWorkshops2009: EvoCOMNET, EvoENVIRONMENT, EvoFIN, EvoGAMES, EvoHOT, EvoIASP, EvoINTERACTION, EvoMUSART, EvoNUM, EvoPhD, EvoSTOC, EvoTRANSLOG*, LNCS, Tubingen, Germany. Springer Verlag.

Wilson, Garnett and Heywood, Malcolm (2007). Introducing probabilistic adaptive mapping developmental genetic programming with redundant mappings. *Genetic Programming and Evolvable Machines*, 8(2):187–220. Special issue on developmental systems.