Title	The syntax of stock selection : grammatical evolution of a stock picking model
Authors(s)	McGee, Richard; O'Neill, Michael; Brabazon, Anthony
Publication date	2010-07
Publication information	2010 IEEE Congress on Evolutionary Computation (CEC) [proceedings]
Conference details	Congress on Evolutionary Computation, IEEE World Congress on Computational Intelligence, Barcelona, Spain, 18-23 July
Publisher	IEEE Press
Link to online version	http://dx.doi.org/10.1109/CEC.2010.5586001
Item record/more information	http://hdl.handle.net/10197/2732
Publisher's statement	Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.
Publisher's version (DOI)	10.1109/CEC.2010.5586001

The Syntax of Stock Selection: Grammatical Evolution of a Stock Picking Model

Richard McGee, Michael O'Neill, Anthony Brabazon

***Abstract*— A significant problem in the area of stock selection is that of identifying the factors that affect a security's return. While modern portfolio theory suggests a linear multi-factor model in the form of Arbitrage Pricing Theory it does not suggest the identity, or even the number, of risk factors in the model. Candidate factors for inclusion in a fundamental model can include hundreds of data points for each firm and with thousands of firms in the fund manager's selection universe the model specification problem encompasses a large, computationally intense search space. Grammatical Evolution (GE) is a form of evolutionary computing that has been used successfully in model induction problems involving large search spaces. GE is applied to evolve a stock selection model with a customized mapping process developed specifically to enhance the performance of evolutionary operators for this problem. Stock selection models are rated using fitness functions commonly employed in asset management; the information coefficient and the inter-quantile return spread. The findings of the paper indicate that evolutionary computing is an excellent tool for the development of stock picking models.**

I. Introduction

Stock selection models have traditionally been constructed as a linear combination of a fixed number of factors. This structure has been motivated by the most widely adopted models developed by modern financial theory to explain the contemporaneous cross section of stock returns. These include the Capital Asset Pricing Model (CAPM), introduced in [19], [11] and [13], and Arbitrage Pricing Theory (APT), see [18]. Both theories pose estimation problems in that for CAPM the sole factor, the market portfolio, is not measurable and for APT the identity and number of the risk factors are not specified. Specification of a relevant set of factors is of interest to asset managers, both in risk management and in the forecasting of expected returns, and many different approaches to the estimation of risk factors have been put forward in the finance literature. Three different types of multi-factor model are classified in [3], macroeconomic models, using macroeconomic indicators such as inflation and term spread; fundamental models, using a firm's fundamental accounting variables such as price to earnings and dividend yields; and statistical models using approaches such as principal component analysis. The best results were obtained using a fundamental model with the author suggesting that this success may be due to the fact that macroeconomic data is already embedded in firm fundamentals. A seminal operationalisation of a multi-factor model is described in [17], Rosenberg later went on to form BARRA (now MSCI BARRA) a company that provides factor models to many of the world's leading asset managers.

Other research has cast doubt on whether firm characteristics lead to increased return via a mechanism of compensation for exposure to a given risk factor, as proposed in APT, suggesting instead that the characteristics themselves identify mispricing based on the behavioral biases of investors. In [4] and [10] the authors find that returns are attributable to firm characteristics levels such as book to market, and are not in reward to exposure to non-diversifiable risk factors. High book to market firms for example may be underpriced by investors extrapolating past poor performance into the future. In [9] it is also noted that the risk factors supposedly captured by the famous size and book to market sorted portfolios of Fama and French [6] have not yet been identified leaving the rational pricing theory somewhat incomplete.

In this paper a non regression based forecasting model structure using firm characteristics is adopted. In the following section it will be expounded that (in the linear case) this model is capable of capturing returns due to compensation for exposure to a risk factor, returns due to a mispricing effect or both.

Given a model structure and an estimation approach we are still left with the specification problem of selecting the relevant factors to include. The goal of extracting these raw components from the huge volume of information available in the financial markets is a daunting one but the large search space can be tamed considerably by the application of financial theory, experience and heuristic optimization techniques. In this paper an evolutionary computing technique called Grammatical Evolution is used to select risk factors from a large candidate pool of fundamental variables for a universe of US stocks. As both the model structure and specification can be explored by GE, model induction is also applied allowing a departure from the traditional format of a linear model. In [2] and [1] the use of Genetic Programming (GP) is found to be beneficial in allowing non-linear, multiplicative interaction terms into the model. In this paper variants of Grammatical Evolution (GE) are applied in the place of GP. The specification of a custom grammar, incorporating domain specific knowledge, enables search space reduction over a GP implementation.

This paper is concerned primarily with determining whether or not GE techniques can be used to evolve a good stock picking model and with analysing the performance of the resulting model. It also contributes to the evolutionary

Richard McGee is with University College Dublin, email: Richard.McGee@ucdconnect.ie

Michael O'Neill is with University College Dublin, email: m.oneill@ucd.ie

Anthony Brabazon is with University College Dublin, email: anthony.brabazon@ucd.ie

computing literature in examining the performance of a customized GE mapping process, where mapping refers to the process of converting a genotype into a phenotype (in this case a stock model) via a user defined grammar. The utility of crossover operations has long been questioned in Genetic Programming (GP) and Grammatical Evolution (GE), [15] demonstrates that crossover in GE is productive and even required in order to obtain solutions to some types of problems. In this paper we find that a customized mapping procedure outperforms a variant of the standard GE mapping when applied to selecting a stock picking model.

II. Data

Fundamental and price data are obtained from the Compustat North American database for the period October 1979 to October 2009, all active and research data are initally included for all stocks in the database during that period. Companies in the financial and utilities sectors are filtered out as we are evolving a model using fundamental data and the different business models in these sectors can lead to a different interpretation of the fundamental data warranting separate models. This leaves a total of 16,976 company members of our investment universe. Companies with a market capitalisation of under 500 million dollars are also filtered out of the database as stocks with a smaller capitalisation can have reduced liquidity leading to increased trading costs and the quality of information reported for these companies can be less reliable.

The fundamental variables obtained from Compustat include an array of accounting and market variables such as price to earnings (PE), price to book value (PB), earnings before interest and taxes (EBIT), return on capital employed (ROCE) etc. Any company without a complete set of return and fundamental variable information on any given month is removed from the database for that month and both active returns and benchmark returns will be calculated for the corresponding month without the company included.

The goal of the process is to extract the information content from the data to differentiate between the stocks in the investment universe. To that end the data is cleaned, clipped and winsorized to within two standard deviations to remove the influence of outliers, the output is a set of factor scores in the range [-2,2]. Histograms are plotted of all the resulting factor information to ensure that the scores are nicely distributed. Some scores may be discarded later if a company is missing data for another factor on that date, but this is performed after the data cleaning so that all available factor data is included in generating the scores. All quarterly reported data is lagged to prevent look ahead bias in the results, quaterly results are delayed by a period of three months to allow for delays in the data becoming available.

The resulting database is filtered to ensure a minimum of 300 companies with a complete set of factor and return data are present in any given month, months not meeting this requirement are not included in our analysis. The average number of companies in the final database used is 914 per month over 229 valid months. Table I below shows the mean return, volatility and exposures to the Fama French factors of the resulting database (BMK) compared with the same measures for the market benchmark (FF MKT) as calculated by Fama & French [1](see [6]). As expected the BMK has decreased exposure to the size factor β_{smb} due to our market cap filter, there is also decreased exposure to the growth factor β_{hml}. The benchmark is 97.88% correlated with the FF MKT, has almost identical return volatility and an increased mean return.

III. The Selection Model

A. Model Structure

As discussed in the introduction a characteristic based model similar to that described in [4] is adopted:

$$E\left[r_t\right] = a + \sum_{i=1}^{N} b_i.\theta_i$$

Where r_t represents the stock price return at time t; b_i represents an array of weightings; θ_i represents a vector of slowly changing firm characteristics e.g. price to book and a is a constant risk free return (which for the purposes of stock picking can be removed from the model as it is common to all stock returns). In our implementation we do not try to estimate the r_t values directly, we are using the $\sum_{i=1}^{N} b_i.\theta_i$ term to generate a relative score to rank a particular stock versus the other securities with the b_i weightings evolved using GE. Although this model is not estimated as a traditional risk factor model using regression analysis it does have potential to capture returns due to a risk premium. If the firm characteristic is taken to represent a beta (as it represents the level of exposure to a risk factor) the weighting vector b_i can then be interpreted as representing the expected return or premium for each risk factor.

In our implementation the value of N is limited to a maximum of five factors and these members of the vector θ_i are selected from a pool of over thirty variables by an evolutionary algorithm. Characteristics in the pool include fundamental accounting variables for each firm available in Compustat such as price to earnings (PE), return on investment (ROI), return on capital employed (ROCE), cash flow etc.

As described in the Data section all of the candidate fundamental data for our model are normalized and winsorized to reduce the impact of outliers above two standard deviations from the mean. The weights and polarity given to the selected characteristics in the model, b_i, are also determined during the optimization stage.

B. Fitness Function

Two fitness functions are considered to evaluate the performance of a selected model: the Information Coefficient and the Inter-Quantile Spread.

[1]data was downloaded from Kenneth French's website: `http://mba.tuck.dartmouth.edu/pages/faculty/ken.french`

TABLE I
PORTFOLIO ANALYSIS (P VALUES IN BRACKETS)

Portfolio	Return Mean	Return Volatility	β_{rm-rf}	β_{smb}	β_{hml}	Residuals Mean
FF MKT	0.0069	0.0434	1.0143(< 0.0001)	-0.0061(< 0.0001)	.0167(< 0.0001)	0.0032(< 0.0001)
BMK	0.0096	0.0432	0.9783(< 0.0001)	-0.0989(< 0.0001)	-0.0802(< 0.0001)	0.0064(< 0.0001)
Long Only	0.0211	0.0498	1.0644(< 0.0001)	-0.1585(< 0.0001)	-0.1825(< 0.0001)	0.0179(< 0.0001)
Long-Short	0.0268	0.0386	0.0242(< 0.0001)	-0.2887(< 0.0001)	-0.4275(< 0.0001)	0.0284(< 0.0001)

1) Information Coefficient (IC): The information coefficient measures the Spearman rank correlation of the forecast from the evolved model with the following period's outcome. In our implementation we measure the correlation of the ranking that the model assigns to stocks with the actual ranking of the stocks sorted by their return for the following time periods:

$$IC_t = Correlation(E\left[rank(r_{i,t})\right]_{t-1}, rank(r_{i,t}))$$

Where $E\left[rank(r_{i,t})\right]_{t-1}$ is the expected ranking at time t-1 of the return for stock i at time t, and $rank(r_{i,t})$ is the realized ranking of the stock return at time t. For the task of comparitively ranking and classifying the stocks in the investment universe the IC is an appropriate fitness function, however it should be noted that it does not include any measurement of risk for a portfolio resulting from the selected model. It is assumed here that the risk management is performed in a second portfolio construction phase. For more discussion on the interpretation of information coefficients see [7]. As a general guideline an acceptable IC value for a factor in investment management is around 5 % (with a higher value obviously being better).

2) Inter-Quantile Spread (IQS): Stocks are split using percentiles into three quantiles representing a buy, a sell and a neutral classification. The Inter-Quantile Spread is calculated as the difference in returns between the market weighted portfolio of top quantile ranked (buy) stocks and the market weighted portfolio of the bottom quantile ranked (sell) stocks:

$$IQS_t = \sum_{i=1}^{N} \frac{rtop_{i,t}.MC_{i,t}}{\sum_{i=1}^{N} MC_{i,t}} - \sum_{i=1}^{M} \frac{rbottom_{i,t}.MC_{i,t}}{\sum_{i=1}^{N} MC_{i,t}}$$

Where $MC_{i,t}$ is the market capitalisation of the stock i at time t, $rtop_{i,t}$ is the return of stock i in the top quantile portfolio at time t and $rbottom_{i,t}$ is the return of stock i in the bottom quantile portfolio (where there are N stocks in the top(buy) portfolio and M in the bottom(sell) portfolio).

If the model is effective at ranking the securities a portfolio made up of the top ranked securities should significantly outperform the benchmark and the bottom ranked portfolio should underperform the benchmark (where the benchmark is taken as the average return of all stocks in the database available for inclusion in our portfolio).

C. Overfitting Risk

The process of factor subset selection is prone to overfitting of the final model to the data set used. If too many candidate factors are included the number of degrees of freedom of the model is increased to the point whereby the model may be fitted to random noise in the historical data, e.g. see [16]. This results in poor out of sample performance when the model is tried on new data. Measures are taken both to reduce overfitting to the data set and to enable an evaluation of the model on a separate data set not used to fit the model. A randomly drawn sample period is removed and held aside for out of sample evaluation of the selected model and the model is fit to the remaining in-sample data. A k-fold cross validation approach is used in evaluating the fitness function on this in-sample data. The in-sample data is split into k sub-periods and sub-samples are created by removing the ith sub-period for each i in the range 1 to k. Fitness is then evaluated for each of the resulting k sub-samples of length k-1. The minimum fitness value for the model over all of these sample periods is then returned as the fitness value for the selected model during the evolutionary process. The final evolved model is then run on out-of-sample data as an independent validation that performance is not due solely to overfitting of the data. The number of factors in candidate models was also restricted to five to limit the potential for overfitting.

IV. EXPERIMENTS

Experiments were performed to develop a mapping procedure best suited to the selection model problem. The performance of two mapping procedures, the first a tree depth constrained standard GE mapping, and the second a custom mapping developed specifically for this problem (described below), were compared. A sample stock selection problem was set up with the goal of evolving a model using GE with the Information Coefficient as the fitness function. The metrics described in section IV-D below were then collated over thirty runs for each mapping. The model size is fixed at 8 factors for these experiments, this is reduced to 5 factors when applying the chosen mapping to evolve a model in order to limit potential overfitting.

A. Grammatical Evolution

Grammatical evolution is an evolutionary computation system that can evolve computer programs, rule sets or more

generally sentences in any language, see [5], [14]. A Backus-Naur Form grammar is defined and this provides production rules used in generating the output language from the system. In the implementation considered in this paper the output language will be the specification of a stock selection model but more generally GE can be used to generate a program in any language. Two different mapping procedures are tested here: a constrained standard GE mapping limited to a tree depth of 2 and the custom mapping procedure described below.

1) Constrained Standard GE Mapping: A Backus Naur Form (BNF) Grammar consists of the tuple $< T, N, P, S >$, for our implementation these are defined as follows:

S, Start Symbol:

```
<EXPR> OP <EXPR>
```

P, Production Rules Set:

```
OP ::= + (1)
     | - (2)
     | * (3)

EXP ::= <EXP> <OP> <EXP> (1)
       | C*F (2)

C ::= 0.05 (1)
    | 0.1 (2)
    | 0.15 (3)
    .....
    | 0.95 (19)
    | 1.0 (20)

F ::= PE (1)
    | PB (2)
    | ROI (3)
    .....
    | ROCE (29)
    | MktVal (30)
```

N, Non-terminals:

```
<EXP>
```

T, Terminals:

```
C, F, OP
```

2) Example Mapping: The maximum tree depth for the model representation is set to three to limit the maximum number of factors in the model to eight. This restriction is motivated by domain specific knowledge and practice. The number of genes or codons in the chromosome is set to 29, this amount is sufficient to map up to an eight factor model. An example mapping of genotype to phenotype is given in Figure 1. The chromosome used is:

```
chrom = [1,2,19,8,1,2,11,29,1,2,10,103,
         1,1,2,7,16,1,2,16,74,2,2,8,31]
```

Giving the Final Example Model:

```
C(19)*F(8)+C(11)*F(29)-C(10)*F(103)+
C(7)*F(16) + C(10)*F(74)-C(8)*F(31)
```

3) Custom Mapping: The customized mapping developed for this application of GE is designed to preserve the schemata or building blocks that drive the evolutionary process, see [8]. To achieve this, locus points in the same position on the genes of two potential parents always contains the same structural component when mapped in a final model, whether that structural component is a terminal or non-terminal operator value. In other words, if a gene at a locus in parent 1 determines an operator type the gene at the same locus in parent 2 should also determine an operator type and if it determines a factor or coefficient value in one parent it must do so in both parents. This is achieved by fixing the structure so that there is a repetitive alternation between operators and non-operator terminals. In practice this leads to a fixed structure made up of building blocks repeating every three codons.

```
Model ::= C*F

while OP != Terminate
    Model ::= Model OP C*F
Endwhile

OP ::= '+'
      |'-'
      |'*'
      |'+(' / ')+'
      |'-(' / ')-'
      |'*(' / ')*'
      |'Terminate'
```

The 'Terminate' operator allows variable length in the model but a maximum length of eight factors is set to allow comparison with the standard GE implementation described above (also limited at eight factors). The operator terms including brackets allow for nesting analogous to tree depth in the standard mapping. There is a choice of two bracketed operators in each case for each operator. If an odd number of open brackets were mapped previously then the closed bracket term is selected otherwise a new nested expression is started with an operator and an open bracket.

4) Example Mapping: An example of a custom mapping is given in Figure 2.

B. Implementation

A genetic algorithm handling real number valued chromosomes was implemented in Matlab. The GA uses stochastic uniform selection, single point crossover and mutation, and elitism was implemented to keep five of the best solutions from each population in its replacement population. The GA produces a population of genotypes and the mapping of genotype to phenotype is handled by an implementation of either

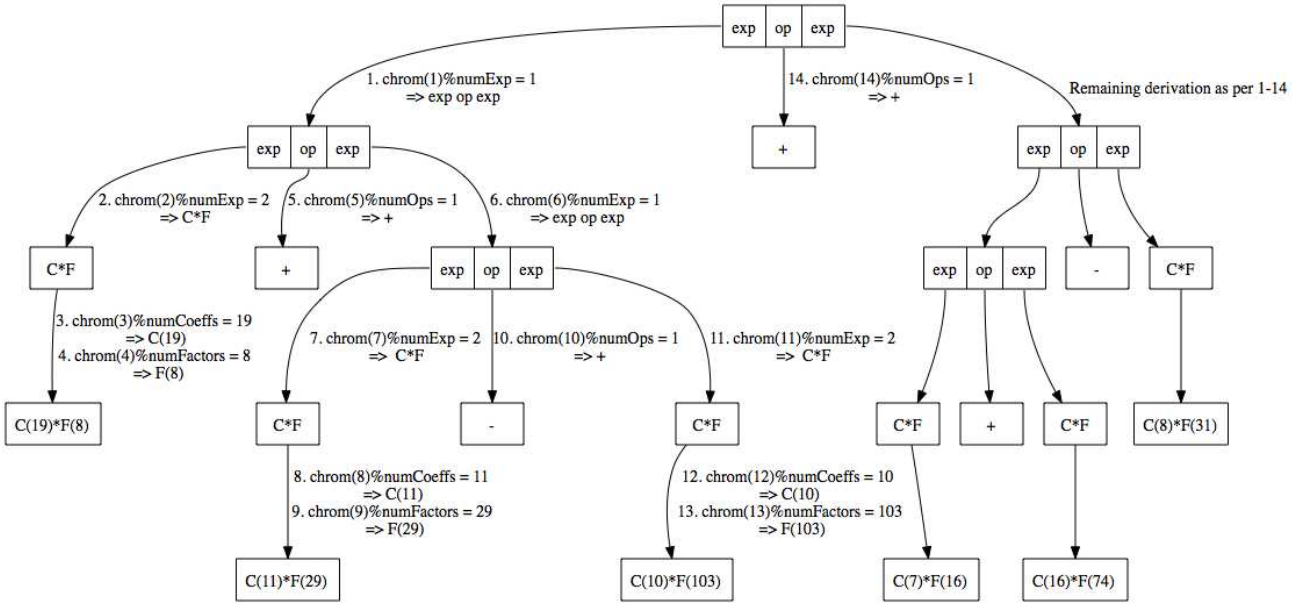

Fig. 1. Example of a constrained standard GE mapping

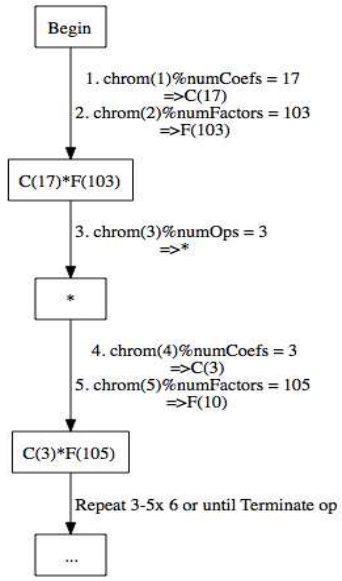

Fig. 2. Custom GE mapping example.

one of the mapping functions described above(depending on which is being tested). A final Matlab function is called to evaluate the fitness calculation (either IC or IQS) for the obtained phenotypes and this feeds back into the GA process as the fitness value of a phenotype.

C. Settings

The GE settings used in all experiments and in the final model evolution are listed in table II below.

TABLE II
GE SETTINGS

Mutation Probability	.05
Crossover Probability	.8
Crossover Method	Single Point
Selection Method	Stochastic Uniform
Num Generations	50
Population size	500
Number of Runs	30
Elitism Number	5
Fitness function	IC
Standard Map Chrom. Length	29
Custom Map Chrom. Length	23

D. Metrics

The performance metrics used in experiments are listed below:

- Best Fitness: Fitness of the fittest individual in a generation.
- Average Fitness: Mean fitness of a generation.
- Std Deviation Of Fitness: Standard deviation of solutions in a generation about the population mean.
- Percentage Crossover & Mutation Success: Number of crossed over and mutated children with fitness greater than that of both parents, divided by number of crossed over and mutated children in a generation (expressed as a percentage).
- Average Fitness Jump: The average improvement in fitness of a child over the best fitness of its two parents when crossover and mutation are successful (as defined in the previous metric).

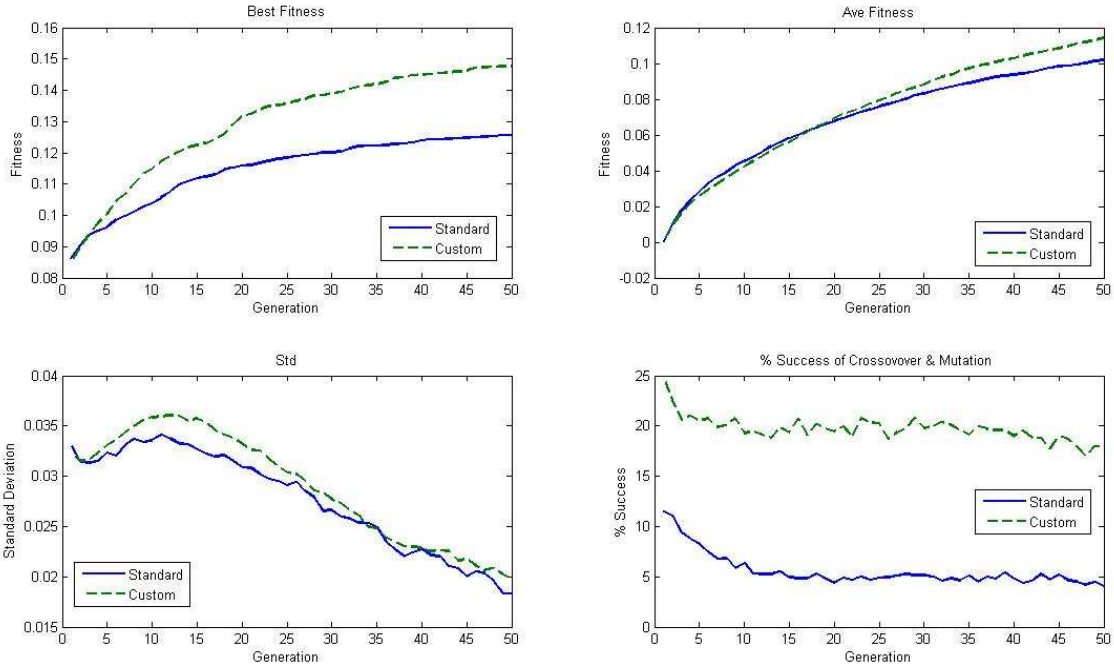

Fig. 3. Metrics (per generation over 30 runs)

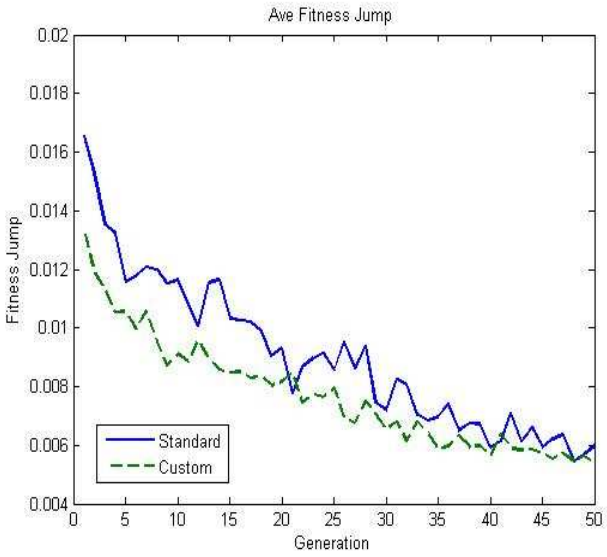

Fig. 4. Average Fitness Jump (per generation over 30 runs)

V. Results

A. Mapping Performance Results

The custom mapping outperforms the standard mapping in terms of the best fitness and average fitness metrics (see figure 3). The average fitness jump figure suggests that the reason for this success is the increased number of successful crossover and mutation operations for the custom mapping with approximately 15% more of the operations resulting in success for the custom mapping on average. This also leads to a more diverse population for the custom mapping procedure, as shown in figure 3, as fitness based selection will result in the same parents being selected more often using the standard mapping as destructive operations reduce the pool of fit parents. Figure 4 shows that the resulting fitness jump for successful operations is larger on average for the standard mapping process. This benefit seems not to be exploited however as successful operations occur too infrequently, as shown by the low percentage success of crossover & mutation illustrated in Figure 3, to prevent the convergence in the population on a sub optimal solution reflected in the inferior best fitness result.

B. Model Performance Results

1) IC Fitness: The fittest model evolved using the IC fitness function was a four factor model:

$$0.1 * PE + 0.35 * ROE + 0.2 * EBITDA - 0.35 * MktVal$$

Positive weightings were given to price to earnings (PE), return on equity (ROE) and earnings before interest, taxes, depreciation and amortization (EBITDA) and negative weighting to large cap stocks. The positive weighting on PE is surprising as value stocks are known to outperform over this period, the weighting is small however relative to that of the other factors. The negative weighting on market value is consistent with research showing that small stocks outperform over the period (e.g. see [6]).

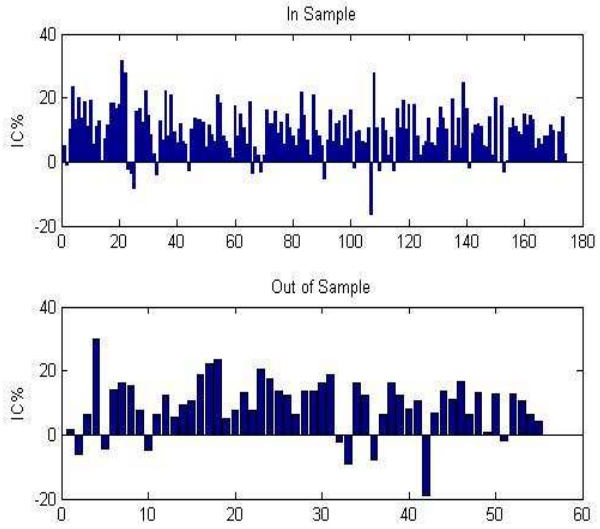

Fig. 5. Monthly Information Coefficients for the IC fitness evolved model.

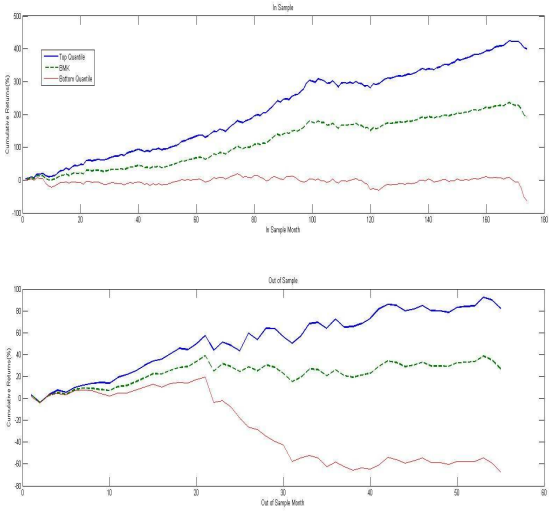

Fig. 6. Quantile Portfolio Monthly Returns for the IQS fitness evolved model vs the benchmark portfolio .

The IC value obtained for the model was a strong 8.2% in sample, cross validated, 9.73% over the whole in sample period and 9.27% out of sample. The performance of in and out of sample periods are consistent indicating that overfitting is not an obvious issue. Figure 5 below shows the monthly IC values across both in sample and out of sample periods. The result over the whole period is highly statistically significant with a p value of $< .0001$. Some periodicity appears to be present in performance suggesting that there may be market regimes when the model underperforms; the handling of this issue will be considered in future work (see section VI).

2) IQS Fitness: The fittest model evolved using the IQS fitness function was a three factor value model:

$$-0.73 * PE - 0.23 * EPS + 0.04 * ROI$$

Negative weightings were given to price to earnings (PE), earnings per share (EPS) and a small positive weighting to return on investment (ROI).

The spread between top and bottom quantile portfolios is 2.66% on average per month in sample and 2.72% out of sample. The performance in and out of sample are consistent indicating that the cross validation process has worked well in limiting overfitting to the data. Figure 6 shows the performance of the quantile portfolios versus the benchmark portfolio across both in sample and out of sample periods, the ranking is clearly visible with returns for the first quantile above the benchmark and returns for the final quantile below it.

In figure 7 the returns to two strategies are illustrated. The first is a long minus short strategy where the investor goes long (purchases) the upper quantile portfolio and shorts (or borrows) the lower portfolio to mimic the spread returns obtained above with an average 2.68 % return (it should be noted that this does not include portfolio rebalancing costs and leveraging costs incurred in shorting the stocks).

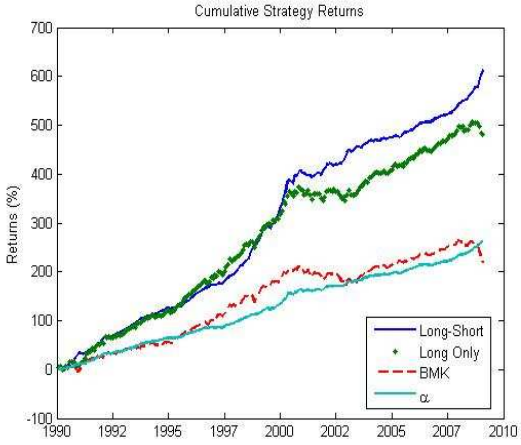

Fig. 7. Strategy Returns for the IQS fitness evolved model.

The second strategy is a long only strategy, investing only in the upper quantile portfolio. The alpha plot in figure 7 represents the gains for active management over a passive index fund, returns over and above the benchmark (an average of all of the stocks available in our investment universe) for this strategy average 1.15% per month for the period. Regressing the returns of both strategies against the three Fama French factors we get statistically significant residuals with a mean of 1.8 % for the long only and 2.8 % for the long-short portfolio per month, see table I. Both of these values are highly statistically significant indicating that the performance cannot be explained by the Fama & French three factor model.

VI. Future Work

It should be noted that the fitness functions maximised during our analysis were selected to rank stocks in terms of

returns only as this was considered a clear metric to evaluate the methodology. For a practical investment strategy a risk adjusted measure such as the Sharpe, Sortino, MAR or omega ratio or a multi-objective fitness function would be used to balance risk with return in evolving an investment model.

The results in this paper indicate that there is scope for the development of stock selection models utilising evolutionary based techniques, which employ grammatical evolution. For any developed financial model to be useful in practice however, care would need to be taken to ensure a number of further steps were taken. The application of domain knowledge from the finance industry would enable pre-screening of accounting factors to a set of practical metrics, comparable across sectors/regions.

The model specification in this paper assumed the prevalence of a static environment, and in particular the existence of static risk or mispricing premia. There is an extended literature however on the dynamic nature of financial markets and time varying risk premia. The Adaptive Market Hypothesis [12] proposes a competitive market with strategies constantly evolving, competing and interacting with market conditions. A natural extension of the work in this paper would be to incorporate these dynamics in evolving models for different market conditions. EC techniques provide a natural framework for this research and the EC literature already contains interesting research in this area, see [5].

VII. Conclusions

The stock selection model specification problem encompasses a large, computationally intense search space ideal for the application of an evolutionary algorithm. Strong performance was obtained when applying GE to the problem for both fitness functions considered. A high average information coefficient of over 9% and an average inter-quantile return spread of 2.6% per month were obtained and these were consistent out of sample indicating that the obtained models were successful at ranking members of the target universe in terms of future expected returns. The fittest models obtained in the process did not include multiplicative interaction terms, in both cases these models were rejected over the standard linear format of models traditionally used in stock selection. It is noted that non linear models were selected in the absence of cross validation suggesting that the inclusion of these terms can lead to overfitting.

This paper also contributes to the EC literature as results obtained in the mapping experiments point strongly in favor of using a customized mapping procedure that preserves the structural reference of genes in the evolved chromosome. The customized mapping outperformed in terms of best fitness, average fitness and population diversity. This is consistent with theory on the importance of schemata preservation in evolution [8] and suggests that the power of GE in mapping via a grammar may be better harnessed by embedding structure in the mapping process.

Acknowledgements

This publication has emanated from research conducted with the financial support of Science Foundation Ireland under Grant No. 08/SRC/FM1389.

References

[1] Y. Becker, P. Fei, and A. Lester. Stock selection: An innovative application of genetic programming methodology. *Genetic Programming Theory and Practice IV, Springer, Ann Arbor*, 5:315–334, 2006.

[2] Y. Becker and U. O'Reilly. Genetic programming for quantitative stock selection. *Proceedings of the first ACM/SIGEVO Summit on Genetic and Evolutionary Computation*, pages 9–16, 2009.

[3] G. Connor. The three types of factor models: A comparison of their explanatory power. *Financial Analysts Journal*, 51(3):42, 1995.

[4] K. Daniel and S. Titman. Evidence on the characteristics of cross sectional variation in stock returns. *Journal of Finance*, 52(1):1–33, 1997.

[5] I. Dempsey, M. O'Neill, and A. Brabazon. Foundations in grammatical evolution for dynamic environments. *Springer Verlag*, 2009.

[6] E. F. Fama and K. R. French. Common risk-factors in the returns on stocks and bonds. *Journal of Financial Economics*, 33(1):3–56, 1993.

[7] R. Grinold and R. Kahn. Active portfolio management. *McGraw Hill*, 2000.

[8] J. H. Holland. Adaptation in natural and artificial systems. *Ann Arbor, MI: University of Michigan Press*, 1992.

[9] J. Lewellen. The time-series relations among expected return, risk, and book-to-market. *Journal of Financial Economics*, 54(1):5–43, 1999.

[10] J. Lewellen. Predicting returns with financial ratios. *Journal of Financial Economics*, 74(2):209–235, 2004.

[11] J. Lintner. The valuation of risk assets and the selection of risky investments in stock portfolios and capital budgets. *Review of Economics and Statistics*, 47(1):13–37, 1965.

[12] A. W. Lo. The adaptive markets hypothesis. *Journal of Portfolio Management*, page 15, 2004.

[13] J. Mossin. Equilibrium in a capital asset market. *Econometrica*, 34(4):768, 1966.

[14] M. O'Neill and C. Ryan. Grammatical evolution. evolutionary automatic programming in an arbitrary language. *Kluwer Academic Publishers*, 2003.

[15] M. O'Neill and C. M. Ryan, C.and Keijzer M. Crossover in grammatical evolution. *Genetic Programming and Evolvable Machines*, 4(1):67–93, 2003.

[16] J. Reunanen. Overfitting in making comparisons between variable selection methods. *Journal of Machine Learning Research*, 2003.

[17] B. Rosenberg. Extra-market components of covariance in security returns. *Journal of Financial and Quantitative Analysis*, 9(2):263–274, 1974.

[18] S. A. Ross. Arbitrage theory of capital asset pricing. *Journal of Economic Theory*, 13(3):341–360, 1976.

[19] W. F. Sharpe. Capital-asset prices - a theory of market equilibrium under conditions of risk. *Journal of Finance*, 19(3):425–442, 1964.