A Genetic Programming Approach to the Dynamic Portfolio Rebalancing Problem

Vijay Karunamurthy
E*trade Financial
4500 Bohnannon Dr.
Menlo Park, CA 94024
vj@imsa.edu

ABSTRACT

Modern portfolio theory holds that the set of efficient portfolios are those that minimize mean variance for a given return; however, the question of how portfolios should be rebalanced over time, given changing correlations among asset class returns and transaction costs, is an open one. Genetic programming enables the discovery of rebalancing methodologies that can generate excess returns over a passive portfolio, while taking into account significant transaction costs and uncertain values for asset class correlations.

1 INTRODUCTION

Financial applications offer many possibilities for optimization by means of genetic programming. One of these is in the field of modern portfolio theory. Portfolio theory is concerned with asset allocation and the diversification of portfolios to meet the risk and return objectives of individual or institutional investors. Previous applications of genetic programming to finance have focused on technical analysis, such as in Neely (1997), or financial forecasting, as in Dempster (2001). These applications all have an underlying methodology of using historical data and trends to predict the future performance of assets and outperform the market. In contrast, portfolio rebalancing in portfolio theory is an application predicated on the "efficient market" hypothesis. Instead of attempting to pick assets with outperforming potential, portfolio rebalancing seeks to diversify a group of assets to maintain an optimized portfolio, without predictions on the future performance of those assets. Dynamic portfolio rebalancing extends this diversification problem to multiple points in time. The constraints on the dynamic portfolio rebalancing problem are both linear (bid-ask spreads) and non-linear (tax consequences) in the magnitude of transactions, and the problem deals with uncertain parameters computed from historical data. As a consequence, no general method of solving the dynamic portfolio rebalancing problem has been demonstrated. This paper will discuss a genetic programming approach that generates a trading system capable of beneficially rebalancing a portfolio in the face of transaction costs.

2 PORTFOLIO OPTIMIZATION

The field of finance known as portfolio theory is concerned with the allocation of funds by an investor to a set of assets whose expected returns and risk will achieve the investment objectives of the investor. The foundations of modern portfolio theory were laid down by Harry Markowitz in 1952, in an approach known as mean-variance optimization (for which he would be later awarded the Nobel Prize.) Mean-variance optimization breaks down the expected performance of a portfolio of assets into 2 variables: expected risk and expected return.

Over a given time period, the expected return of a portfolio of assets is the price-weighted average of the expected return of all the assets in the portfolio:

$$E[r] = \sum w_i r_i$$

where w_i is the weight of asset i in the portfolio, such that the sum of all weights equals 1, and r_i is the expected return of an asset i. The expected return includes capital appreciation or depreciation, as well as any dividends or interest paid. Many formulations of mean-variance optimization do not take into account tax effects, although ideally after-tax returns should be utilized. The expected risk of a portfolio is defined as the variance of the expected return.

For a portfolio of 2 assets, asset A and asset B, with individual expected variances σ_A and σ_B respectively, the portfolio variance is

$$\sigma_P^2 = w_A^2\sigma_A^2 + w_B^2\sigma_B^2 + 2w_A w_B \cdot \sigma_A \cdot \sigma_B \cdot \rho_{AB}$$

where ρ_{AB} is the expected covariance between the assets. The portfolio variance is often computed as

$$\sigma^2_P = \boldsymbol{x}\sum\boldsymbol{x}'$$

where $\boldsymbol{x}$ is a vector of the weights of the assets in the portfolio, and $\sum$ is a matrix of the expected covariances of the asset returns (this matrix is guaranteed to be positive definite, so that the resulting variance is positive.) Mean-variance optimization asserts that, when choosing between 2 possible portfolios with equal expected returns, all investors would choose the portfolio with the least expected variance. Conversely, when choosing between portfolios with equal expected variances, all investors would choose the portfolio with the greatest expected return. The set of portfolios that offer the best return for a given level of risk, or the least risk for a given level of return, are known as the efficient frontier of portfolios. For stock portfolios, in a world with a few arbitrage opportunities, the efficient frontier is comprised of portfolios with a high degree of diversification. Diversification has a positive effect in that variations in the return of one asset can be partially offset by variations in the return of an uncorrelated or a negatively uncorrelated second asset in the portfolio, potentially reducing portfolio variance without affecting return.

There are several commonly acknowledged problems with mean-variance optimization. One is that under normal conditions for risky assets, both expected returns and expected variances exist under uncertainty. These are both forward-looking variables which must be estimated from historical data, yet no guidelines exist for what form the estimation should take or what historical data should be utilized. Another problem is that mean-variance optimization assumes an initial instantaneous allocation period with no guidelines for how long the resulting portfolio should be maintained. A real-world portfolio must be rebalanced over time to continue to meet an investor's objectives, even if those objectives remain unchanged, as expected returns and expected variances often change drastically with time. At the same time, rebalancing requires the buying and selling of assets, which are actions with associated costs. These so-called "transaction costs" include capital gains taxes that must be paid on profits, as well as typical market costs, such as those reflected in the spread between quoted bid prices and ask prices on stock exchanges. Transaction costs for most financial assets are significant, such that many asset managers recommend that assets be held for a several years before selling. In addition, Jobst, et. al. (2001) have noted that the addition of discrete trading constraints (the buying or selling of integral numbers of shares) prevents an easy formulation of the optimization problem, and makes the construction of an optimum solution an NP-hard problem.

The dynamic portfolio rebalancing problem can be expressed in terms of a desired trading strategy. An investor wishes to know a set of criteria by which assets can be traded to maintain an optimum portfolio over time. The result of applying the strategy to a data set corresponding to a point in time should be a set of trades. Financial data sets can contain any number of variables, including economic data, moving averages of prices or geopolitical events. However, for portfolio rebalancing under mean-variance optimization, the data set can be restricted to those variables derived from historical returns of assets. This restriction is justified by the grounding of portfolio theory in the "efficient-market hypothesis", which in a typical form suggests that individual asset prices quickly reflect any available data on future risk and returns. Modern portfolio theory is not interested in picking individual assets which will "outperform the market", since any such assets would create arbitrage opportunities, and quickly update their prices accordingly. Instead it is interested in the diversification of the assets in a portfolio so that the risks and returns meet investors' objectives.

The benchmark portfolio utilized in this paper is the basis of the Dow Jones Industrial Average (DJIA) index, as in Li (2001). This is a portfolio of 30 large-cap stocks (securities) whose prices are divided by constant to obtain the DJIA, which is widely reported on financial sites. The portfolio is almost completely passive, with the last major change in composition occurring in November of 1999, when stocks such as Microsoft and Intel were added to the index and Texaco and Woolworth were dropped. The securities in the portfolio come from a wide variety of market sectors, and changes in the weights of portfolio components often reflect the performances of one sector over another, for example high-tech's outperformance of retail stocks during the tech boom.

The most widely used tool for tracking the performance of an active portfolio is based on the work of Stanford professor and Nobel laureate William Sharpe. The Sharpe ratio is the ratio of the expected return of a portfolio to the expected standard deviation of the returns. Portfolios with higher Sharpe ratios are considered superior in a mean-variance optimization sense; thus investors are willing to take a higher level of risk in return for a proportionately higher level of return. In comparing the returns of an actively managed portfolio (the result of a trading strategy) with that of a benchmark (the DJIA portfolio) the Sharpe ratio becomes the information ratio, which is the ratio of excess returns relative to a benchmark to the standard deviation of those excess returns. Among American large-cap mutual funds actively managed mutual funds for 2002, the median information ratio was –0.04, and the top quartile of funds achieved an information ratio of .41. There is a direct relationship between the information ratio and the statistical significance of excess returns: the t-statistic for the hypothesis that excess returns are positive and statistically significant is the information ratio times the square root of the number of samples, with a degree of freedom equal the number of samples.

3 METHODS

Each individual trading strategy evaluated by the genetic programming method was represented by 2 separate tree structures corresponding to LISP S-expressions. These tree structures were easily coded and evaluated in parallel in the ECJ system by Sean Luke. This section explains the setup and evaluation of the trading strategies. The standard genetic programming tableau appears in Table 1.

Objective	Find a rebalancing trading strategy capable of maintaining a high-information ratio portfolio relative to the DJIA benchmark portfolio
Terminal set for rebalancing-decision branch:	`PRC, ret` and `std`
Function set for rebalancing-decision branch:	+, -, *, /, `RLOG` and `EXP`
Terminal set for trades-producing branch:	`PRC, scov, wght` and `tax`
Function set for trades-producing branch:	+, -, *, /, `RLOG` and `EXP`
Fitness cases:	Performance of the strategy over market data from three years: 2000, 2001 and 2002
Raw fitness:	The average of the information ratio of the trading strategy over the three years
Standardized fitness:	If raw fitness is negative: 2 Else: 1 – raw fitness
Wrapper:	Implementation of trading strategy
Parameters	Population size M = 1,500, number of generations run G= 200
Success predicate	A trading strategy achieves an average information ratio of 1 with positive excess returns in each year (standardized fitness = 0)

Table 1: Genetic programming tableau for portfolio rebalancing problem

3.1 Trading Implementation

The trading strategy represented by each individual's S-expressions was encoded by means of constrained syntactic structures. Each individual was assigned a portfolio for 2000, 2001 and 2002 whose assets in January of each year matched those of a DJIA portfolio worth $163203.07 on January 1st of 2000. The trading strategy was then implemented on the first trading of each month in the year. Each trading date involved evaluating the S-expression of the rebalancing-decision branch, and then optionally the S-expression of the trades-producing branch 30 times (once for each stock in the portfolio.)

The rebalancing-decision branch in each individual produced a value indicating whether a portfolio rebalancing should occur or not. A positive value from the S-expression allowed trading to occur at that time point, while 0 and negative numbers precluded any trades for that time point. If a portfolio rebalancing was to occur, the trades-producing branch was evaluated for each security in the portfolio in turn. Positive-valued results from the S-expression induced buy trades, while negative-valued results induced sell trades. The magnitude of the value indicated the proportion of current assets to be bought or sold; all values greater than 0.5 were treated as 0.5, while all values smaller than –0.5 were treated as –0.5. Only an integral number of shares could be bought or sold at each time point, and all share amounts were rounded down to reach integral numbers. No security in the portfolio was allowed to fall below 1 share as a result of a trade. Trades were resolved as follows:

1) All sell trades were executed immediately, and the proceeds (minus taxes) were added to cash reserves
2) Buy trade requests were held until all requests had been received. Buy trades were then resolved by repeatedly executing the following, until either all trades had been filled or cash reserves were insufficient to fill 1 share of any outstanding trades:
    a. Trade requests were prioritized by the dollar value of outstanding shares to be bought
    b. 1 share of the highest priority trade request was filled

Taxes on the sale of shares were computed as a 28% capital gains rate times the profit on the sale of shares, or 0 if the profit is negative. To determine the profit on a sale, the weighted-average cost basis of each security in the portfolio was maintained separately. Buying of shares was subjected to 0.5% price premium over the prices received on the selling of shares, to reflect the bid-ask spread on the highly liquid shares of large-cap stocks.

3.2 Functions and Terminals

Both branches were given the arithmetic functions {+,-,*,/}, each taking 2 arguments. Note that / is an implementation of protected division, where a result of 1 is returned if division by 0 is attempted. In addition, both branches were given the mathematical functions `RLOG` and `EXP`, each taking 1 argument. `RLOG` is an implementation of protected logarithm, where 0 is returned if 0 is passed as an argument, and the natural logarithm of the absolute value of the argument is returned otherwise. `EXP` is the natural exponent function.

Both branches were given perturbable random constants as terminals. Each perturbable random constant was assigned a random number uniformly distributed between –5.0 and +5.0 in generation 0. In succeeding generations, the reproduction operation was modified to perturb the value of the random constants by a Gaussian variable with a mean of 0 and standard deviation of 1.0. The perturbation is applied with 50% probability to each random constant in a tree selected by the reproduction operator. This has the advantage over the traditional method of ephemeral random constants in that individuals can explore a search space of constant values in a Gaussian fashion in later generations.

The rebalancing-decision branch was also given the terminals {`ret,std`}. Each is a function that takes the current portfolio and date as implicit arguments. `ret` is set at each monthly time step to the absolute return of the managed portfolio over the past month. `std` is set at each monthly time step to the standard deviation of the portfolio's daily returns over the past month, multiplied by the square root of the number of trading days in the month (to extend the standard deviation from a daily to a monthly time horizon).

The trades-producing decision branch was given the terminals {`wght,scov,tax`}. Each is a function that takes the current portfolio, date, and one of the 30 securities in the portfolio as an implicit argument. `wght` is the weight of the security in the portfolio; it is computed as the ratio of the net asset value of the security to the net asset value of the portfolio as a whole. `scov` is a security-portfolio covariance function. It is set at each monthly timestep, for each security, to the rate of change of the portfolio variance function σ^2_P relative to the weight of that security in the portfolio:

$$\delta \sigma^2_P / \delta w_i = \delta(\boldsymbol{x} \sum \boldsymbol{x}') / \delta x_i$$

The covariances in the covariance matrix $\sum$ are computed from the daily returns over the past month. `tax` gives the proportion of the proceeds from a sale of a security that would be lost to taxes if a share of the security is sold. It takes as an additional implicit argument the weighted-average cost basis of the security in the portfolio.

Note that the trades-producing branch has limited information about the past performance of a security, other than what can be inferred from the tax consequences of selling a security and the weight of the stock in the portfolio. This is a deliberate choice to prevent the trading strategy from picking certain stocks based on trends in stock performance.

3.3 Sample Trees

A simple tree for a human-produced set of LISP S-expressions is shown in Figure 1. The LISP S-expression for the rebalancing decision branch is `(- ret std)`, while that of the trades producing branch is `(- (+ wght –0.0600000) (- scov tax))`.

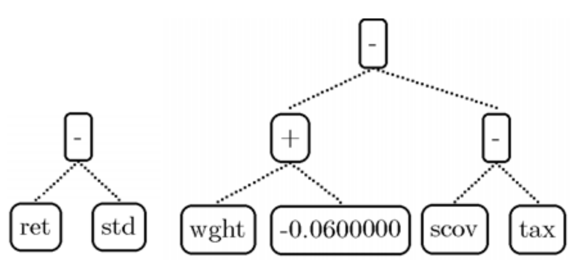

Figure 1. Rebalancing-decision branch (left) and trades-producing branch (right) of sample tree

This sample individual would make a decision to rebalance the underlying portfolio whenever the returns on the portfolio were greater than the standard deviation of those returns. If the decision to rebalance was made, the trades-producing branch would be evaluated for each of the 30 securities in the portfolio. For this sample individual, a buy trade would be made if the rate of change of portfolio variance with respect to asset weight of a security was less than the weight of the asset in the portfolio, plus the portion of sale proceeds that would loss from taxes on a sale, minus .06. If this condition was not met, a sale would be made. The amount of shares to buy or sell would be found by taking the magnitude of the difference, multiplying by the asset value of the security holdings in the portfolio, dividing by the price of 1 share of that security, and rounding down to the next integer.

3.4 Evaluation of fitness

After each individual had conducted the encoded trading strategy over each of the three years, the mean monthly excess returns of the strategy over the benchmark portfolio and the standard deviation of the excess returns were computed. The ratio of these two numbers produced the information ratio for each year. The mean information ratio was taken as the raw fitness of the individual.

To compute the standardized fitness, a check was made if the trading strategy underperformed the benchmark portfolio over the three years; this was determined if the mean information ratio was negative. Since comparisons among negative information ratios are not very informative, a standardized fitness of 2 was assigned for these individuals. For individuals who outperformed the benchmark over the three years, the standardized fitness was computed as 1 – raw fitness.

3.5 Parameters

Experiments were staged with population sizes of 3,000, and the maximum number of generations set to 200. Initial trees were generated half the time with a full method with a depth of 5, and half the time with a grow method with a depth ramp from 2 to 5.

Three breeding phases were used: crossover at a any node (90%), reproduction (9%), and mutation (1%). Reproduction also involved the perturbation of perturbable random constants by a Gaussian value. The maximum depth of evolved trees was set to 17.

Experiments were run on a 4-CPU Intel P4 2.4 Ghz machine, running a symmetric multiprocessing Linux kernel (2.4.7-10smp).

4 DISCUSSION OF RESULTS

The standardized fitness of best-of-run individuals from typical runs was between .5 and .4. In no run was a perfectly fit individual (standardized fitness 0) evolved. Runs typically lasted 20 minutes.

A representative individual that was best-of-generation 4 on a sample run is shown in Figure2. This individual achieved annualized mean monthly returns in excess of the DJIA of 1.08%, 0.72%, and 4.42% in the years 2000, 2001 and 2002, respectively. The standard deviations of the excess returns, on an annualized basis, were 3.48%, 4.97%, and 5.78% in the years 2000,2001 and 2002. The mean information ratio, on a monthly basis, was 0.12. This individual rebalanced the portfolio on just 6 out of 36 months. 7 buy trades were made on those days, with an average of 20 shares purchased per trade. In comparison, 46 sell trades were made, with an average of 21.7 shares sold per trade. Total asset turnover (the percentage of assets sold in a year) averaged 8.8% per year.

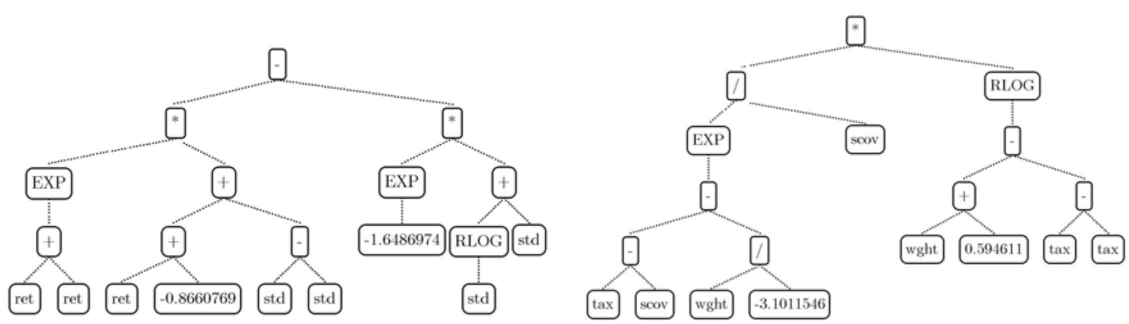

Figure 2 Rebalancing-decision branch (left) and trades-producing branch (right) from best-of-generation 4 individual

This individual is representative of all best-of-run individuals in that it made use of all offered parameters in both the rebalancing-decision branch and the trades-producing branch. This representative individual made the decision to rebalance based in manner proportional to the absolute return of the portfolio over the past month, provided returns were in excess of 0.866%. As the standard deviation of daily returns increased, however, this expression became negative and the opposite decision, not to rebalance, was made.

The trades-producing branch of the representative individual preferred to buy shares that would have high tax consequences if a sale was made. At the same time, securities which were highly correlated with the overall variance of the portfolio were sold, due to the subtraction of `scov` from `tax` on the left hand side of the tree. This trade-off between minimizing variance and the tax consequences of selling were seen in many best-of-generation individuals. Tax consequence also become more prevalent if the marginal tax rate was increased for a sample run.

The LISP S-expression for the rebalancing-decision branch of a best-of-run individual was the following:

```
(* (EXP (+ (+ (EXP (* (* (* (+ (+ (+ (- (*
     std (+ ret -1.3794677)) -1.7768257) -0.8660769)
     -0.8660769) std) (+ (EXP (+ (EXP -2.361492)
     ret)) (+ (- (* std (+ ret -1.3794677)) (+
     ret ret)) -0.8660769))) (* (EXP (+ (EXP -3.0683744)
     (- (- (EXP (+ ret -0.6501038)) (* (EXP -2.1628654)
         (+ std std))) (+ ret ret)))) (* (EXP -0.5191791)
     (+ (- (* std (+ ret -1.3794677)) (+ ret ret))
         (- (EXP -2.1628654) (* (- (- (EXP (+ ret
             -0.6501038)) (* (EXP -2.1628654) (+ std std)))
             (+ ret ret)) (EXP (EXP (+ (EXP -0.5191791)
             ret))))))))) (* std (+ ret -1.3794677))))
     (- (* ret (+ (* (EXP (+ (+ (+ (EXP (+ ret
         ret)) -0.8660769) (- (* (EXP (+ ret ret))
         (+ (+ std std) -1.7768257)) std)) ret)) (+
         (+ (EXP (EXP (+ (+ ret ret) ret))) (* (EXP
             std) (+ (+ ret -0.5608599) (- std std))))
         (- std std))) (+ ret ret))) std)) (+ std
     std))) (- (* (EXP (+ ret ret)) (+ ret -0.6501038))
     (* (EXP -0.5191791) (+ (+ ret -0.8660769)
         (- (* std (+ ret -1.3794677)) (+ ret ret))))))
```

The LISP S-expression for the trades-producing branch of this individual was as follows:

```
(* (/ (EXP (- (RLOG (/ (/ (EXP wght) (RLOG
     (* (* (+ (* (/ (/ (EXP wght) (RLOG scov))
         (* (RLOG tax) tax)) (RLOG (/ (/ (EXP wght)
         (RLOG scov)) (* (RLOG scov) tax)))) (* (EXP
         wght) tax)) scov) tax))) (* (* (* (* tax
     -4.523504) (/ -4.523504 (* (RLOG (/ (/ (EXP
     wght) (RLOG scov)) (* (RLOG tax) tax))) scov)))
     (RLOG (RLOG (/ (/ (EXP wght) (RLOG scov))
         (* (RLOG scov) tax))))) tax))) (/ scov wght)))
     scov) (- (/ wght (* (* (* (* tax -4.523504)
     (/ -3.0230055 wght)) (RLOG (/ wght (* (*
     (EXP scov) (RLOG (/ (/ (EXP wght) (RLOG scov))
     (* (RLOG scov) tax)))) tax)))) tax)) (+ (+
     (EXP wght) scov) (* (RLOG scov) tax))))
```

This individual produced annualized mean monthly returns in excess of the DJIA of 1.46%, 5.3%, and 11.75% in the years 2000, 2001 and 2002, respectively. The standard deviations of the excess returns, on an annualized basis, were 2.09%, 1.48%, and 5.85% in the years 2000,2001 and 2002. The mean information ratio, on a monthly basis, was 0.60. The one-tailed t-statistic with 36 degrees of freedom for the statistical significance of monthly excess returns was 3.6. This is significant at the 99% confidence level. The individual trading strategy rebalanced the portfolio on 8 out of 36 months, or approximately 3 times a year. 43 buy trades were made on those days, with an average of 22.3 shares purchased per trade, and 87 sell trades were made, with an average of 17.8 shares sold per trade. Total asset turnover (the percentage of assets sold in a year) averaged 17.2% per year.

A sense of how this individual traded can be obtained from the chart in Figure 3. Solid lines on this chart track the asset weighting of IBM shares, while dashed lines track the asset weighting of Proctor+Gamble shares. Lightly shaded lines track the weightings of these stocks in the benchmark DJIA basis portfolio, while the darker lines track the weightings of these stocks in the portfolio generated by the best-of-run trading strategy. Since the trading strategy started with a copy of the DJIA sample portfolio in January of each sample year, the weights in this portfolio normalize back to the DJIA weights on these dates. As there are 30 securities in each portfolio, an evenly-weighted portfolio would assign assets weights of .033 to all stocks in the portfolio.

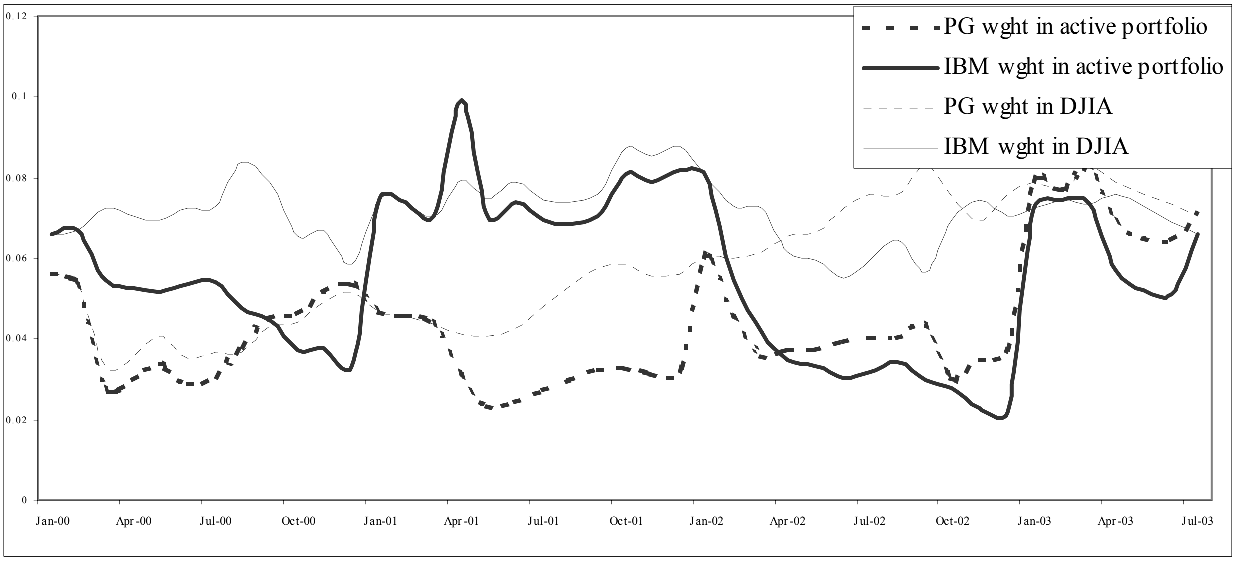

Figure 3. Asset weighting of 2 sample stocks in best-of-run trading strategy and DJIA portfolios

The 6 months beginning in January 2003 were not included in the training data set, and were thus used to validate the trading strategy on non-trained data. The mean monthly return of the Dow Jones Industrial Average over this period was 1.35%, with a standard deviation of mean monthly returns of 4.53%. This substantially outperformed the period included in the data set (mean monthly returns of the DJIA –1.02%, with standard deviation 5.3%.) Figure 4 demonstrates the outperformance of the DJIA over this period.

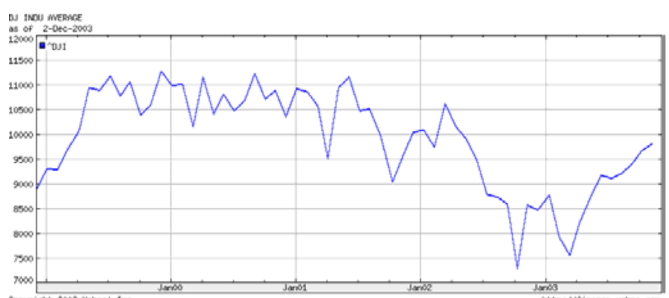

Figure 4. Value of DJIA index from Jan 1999 to November 2003

For the best-of-run trading strategy, annualized mean monthly excess returns of 0.41% were achieved over this 6-month period, with a standard deviation of annualized mean monthly excess returns of 1.64. An information ratio of 0.25 was obtained for the trading strategy over the 6-month period. The one-tailed t-statistic with 6 degrees of freedom for the statistical significance of monthly excess returns over the 6-month period was 0.61. This is not significant at the 99% confidence level. The trading strategy rebalanced on 2 dates throughout the 6 months.

5 CONCLUSIONS

This paper demonstrated that a genetic programming approach to dynamic portfolio rebalancing under transaction costs and integral trade amounts was able to produce statistically significant beneficial trading strategies. These trading strategies were represented as paired S-expressions, with one S-expression devoted to rebalancing decisions and another devoted to individual trading decisions. This methodology allowed decisions on individual trades to be made solely on the basis of the effect of securities on the risk of the portfolio and the tax consequences of trades. The computational complexity of achieving these results is not excessive in the 30-asset model, and may be feasibly extended to 2000 asset or greater markets.

Analysis of Figure 3 gives a broad sense of how the best-of-run trading strategy worked. Both assets had a weighting well in excess of an even weighting of .033 throughout this period. The trading strategy consequently displayed a tendency to reduce holdings in these assets. However, in certain periods the returns on these assets were not significantly correlated with that of the DJIA portfolio. IBM during the 2001 year had returns that significantly outperformed the broader market and a relatively low volatility. The trading strategy consequently evolved a classifier to avoid the sell-off of assets that matched the correlation and weighting profile of IBM during the year. Since the holdings of other assets such as PG were reduced over the same period, jumps in the price of IBM stock, as in April of that year, had a greater impact on the trading strategy portfolio versus the benchmark portfolio.

The fact that so few dates throughout the year were used for rebalancing purposes validates the buy-and-hold strategy preferred by many asset-managers; in fact, the best trading strategies rebalanced even fewer than the once-in-a-quarter approach. However, the rebalancing did not occur on a regular basis and was triggered by market conditions classified by the rebalancing-decision branch, pointing to the value of discontinuous rebalancing decisions. It is also relevant that different runs seemed to produce qualitatively different trading strategies. This is not unusual given the problem area even for highly optimized strategies, as there are a near-infinite number of portfolios on the efficient frontier, and consequently a near-infinite number of trading strategies to generate them. The relative underperformance of the evolved strategy on the 6 month untrained period versus the 36 months trained indicates the values of larger datasets to prevent overfitting of the evolved strategy. The performance of the DJIA and the broader market was quantitatively different over the 6 month period and the conservative trading strategy learned over a depressed market period was not as relevant when certain cyclical stocks outperformed the market, increasing the benefit of taking greater risk.

The use of perturbable random constants as opposed to traditional ephemeral random constants played in fitness throughout each run. A fit individual would often reach higher levels of fitness through a Gaussian-directed search of the random constants in the search space – even though, in the case of the trade-producing branch, the results of evaluations were subjected to integral constraints. The perturbation of random constants played an increasingly key role towards the ends of runs, as highly-fit individuals made incremental improvements to performance, often by changes in a relatively small number of trades.

6 FUTURE WORK

The experiments performed should be expanded to train over a wider variety of assets and asset classes, to realize opportunities throughout the market for diversification. It is also important to train over a longer time period, although we note that the lack of long-term datasets is a universal problem in computational finance. In a real world trading scenario, however, trading decisions would be made on a per-second basis and not a monthly basis, and using intraday return and volatility information as inputs to the model would increase applicability. A greater variety of parameters, such as volume information, option volatilities and futures prices, may also be beneficial for detecting changes in market structure.

Another area to consider is the use of more complicated tax schedules. Even though the weighted-average cost basis used in the experiment offers a model of simplicity, other tax schedules may offer greater returns. These have the problem that marginal tax-rates are non-linear in the number of shares sold. For example, using a last-in-first-out cost basis approach, where 5 shares are bought at $10, then 5 at $5 and then 5 again at $10, the first 5 shares sold for $10 would have no tax consequences, while the next 5 shares would have significant tax consequences, and the last 5 shares sold would again have no tax consequences. The best way to parameterize such information may be to use an additional branch devoted to loading tax information into a memory structure. The trades-producing branch may then reference points in memory to take tax consequences as inputs.

Acknowledgments

I would like to thank John Koza for his advice and encouragement.

References

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

Fabozzi, Frank J. and Markowitz, Harry M. 2002. *The Theory and Practice of Investment Management*. Hoboken, NJ: John Wiley & Sons, Inc.

Jobst, N. J., Horniman, M. D., Lucas, C. A., and Mitra G. 2001. Computational aspects of alternative portfolio choice models in the presence of discrete asset choice constraints. *Quantitative Finance*, Volume 1 (2001) 1-13.

Jorion, Phillipe. 2003. *Financial Risk Manager Handbook, Second Edition*. Hoboken, NJ: John Wiley & Sons, Inc.

Koza, John R. 1991. A genetic approach to econometric modeling. In Bourgine, Paul and Walliser, Bernard (editors.) *Economics and Cognitive Science*. Oxford: Pergamon Press. Pages 57-75.

Koza, John R. 1992. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. Cambridge, MA: The MIT Press.

Li, Jin and Tsang, Edward P. K. 2001. Reducing failures in investment recommendations using genetic programming. *Quantitative Finance.* 1 (July 2001) 397-413.

Neely, Christopher J., Weller, Paul A. and Dittmar, Rob. 1997. Is technical analysis in the foreign exchange market profitable? A genetic programming approach. *Computing in Economics and Finance*. 2000 332-346.

Sharpe, William F. 1992. Asset allocation: management style and performance measurement. *Journal of Portfolio Management*, Winter 1992, pp. 7-19.