Technical Analysis and Genetic Programming: Constructing and Testing a Commodity Portfolio[*]

PRELIMINARY DRAFT.
PLEASE DO NOT CITE OR FORWARD.
COMMENTS ARE WELCOME.

Matthew C. Roberts[†]

---

[*] The helpful comments of Wade Brorsen, Scott Irwin, and participants at the 2003 NCR-134 Meetings are gratefully acknowledged. Any remaining errors are the responsibility of the author.

[†] Assistant Professor, Department of Agricultural, Environmental, and Development Economics, The Ohio State University. 320 Ag. Admin. Bldg., 2120 Fyffe Rd., Columbus, Ohio 43210-1067. Phone: (614) 688-8686. Fax: (614) 292-0078. email: roberts.628@osu.edu. First draft May 19, 2003. This manuscript dated August 13, 2003

Technical Analysis and Genetic Programming:
Constructing and Testing a Commodity Portfolio

Abstract

Although the academic research that supports the usefulness of technical analysis is mixed at best, its use remains widespread in commodity markets. Much prior research into technical analysis suffered from data-snooping biases. Using genetic programming, *ex ante* optimal technical trading strategies are identified. Because they are mechanically generated from simple arithmetic operators, they are free of the data-snooping bias common in technical analysis research. Futures prices from 24 markets are used in rule generation. Rules for only two of the markets are capable of generating statistically significant profits, lending little support for technically-based systems.

Keywords: Technical Analysis, Genetic Algorithms, Commodity Markets, Futures Markets

Technical analysis is a broad collection of methods and strategies which attempt to forecast future prices on the basis of past prices or other observable market statistics, such as volume or open interest. Based on this definition, technical analysis conflicts with weak-form market efficiency, under which "efficiency with respect to an information set . . . implies that it is impossible to make economic profits by trading on the basis of [that information set]," (Malkiel) and the information set consists of precisely the information which technical analysis purports to exploit.

Academia maintains a generally negative view of technical analysis, perhaps best typified by Malkiel, "Obviously, I am biased against the chartist. This is not only a personal predilection, but a professional one as well. Technical analysis is anathema to the academic world." Although there are some that are more charitable toward technical analysis, Campbell, Lo, and MacKinlay suggest that "perhaps some of the prejudice against technical analysis can be attributed to semantics." Nevertheless, the study of technical analysis has a long history in academia, with mixed results.

There is little dispute that technical analysis is very common among practitioners. Oberlechner surveys foreign exchange traders on their use of technical analysis, and finds that

"Only a very small minority of foreign exchange traders demonstrate an exclusively fundamental or exclusively chartist overall forecasting approach." This is consistent with the previous survey research performed by Taylor and Allen, Menkhoff, and Lui and Mole. Brorsen and Irwin report that only two of 21 large commodity fund managers surveyed used no objective technical analysis.

Early studies, such as Alexander and Fama and Blume identified and tested simple technical strategies using equity index data and found that although they may have some predictive power, they were unable to consistently generate positive profits. Over the succeeding decades, similar conclusions were reached by many researchers, especially when transactions costs were included in the analysis. There were a few articles which identified profitable technical strategies, such as Sweeney and Osler[3].

In contrast, more of the studies of technical analysis in the commodity markets have found profits. Lukac, Brorsen and Irwin found that "[s]even of the twelve systems had gross returns significantly above zero. Four of twelve trading systems generated aggregate portfolio net returns signficantly greater than zero." Similarly, Lukac and Brorsen found that "several systems did generate returns significantly above transaction costs." Both of these studies implemented multiple trading systems based upon survey evidence of the systems in actual use by futures funds.

Until Neely, Weller and Dittmar, the modal research method in the study of returns to technical analysis, or asset-price predictability in general, was to first specify one or more trading rules based upon historical usage, fit the parameters of the models based upon historical data and test the profitability of the rules using 'out of sample' data, i.e. data not used in the parameterization process. However, by using strategies that are or have been in use, researchers may be data-snooping, by evaluating historically successful trading rules using historical data.

---

[3] Spyros Skouras has compiled an exhaustive bibliography of academic studies through 1998. It is available at http://www.santafe.edu/∼spyros/tabiblio.htm

Genetic programming can be used to find *ex ante* optimal trading rules from primitive arithmetic operators, thereby avoiding any biases induced by the use of known, successful trading strategies. Using genetic programming, this paper develops optimal *ex ante* trading rules for three agricultural futures markets. Each trading rule is generated using four years of daily closing futures prices, and then tested using the next year's prices for its out-of-sample performance. These tests reveal that these trading rules are quite capable of forecasting periods of high and low returns. The trading strategies are capable of generating profits, but when transactions costs are included, these profits become negligible.

This article has four sections. The first section explains the use of genetic programs in constructing and optimizing technical trading strategies. The second section discusses the evaluation of futures trading strategies and the data used. The third section presents the results of these rules, while the fourth section offers a summary and conclusion.

1 Genetic Programs, Data-Snooping, and Technical Analysis

Genetic programming is the subdiscipline of evolutionary algorithms in which complex algorithms or programs are built from hierarchies of simple operators; they trace their origins to Koza. These programs are optimized according to a evolutionary process whereby an initial population of random rules is generated, they are evaluated according to some 'fitness' function, and then 'evolved' through random combination to form a new generation of rules.

The use of genetic programs as technical trading strategies dates to Neely, Weller, and Dittmarr, (NWD) and Allen and Karjalainen (AK). These researchers recognized that genetic programming avoids the data-snooping biases inherent in earlier technical analysis

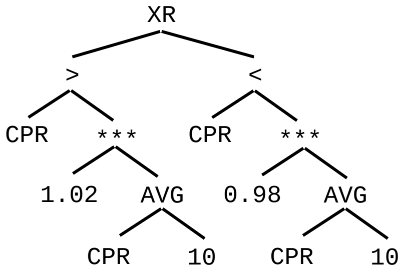

Figure 1: Simple Moving Average Trading Rule

research. In most prior technical analysis research, the performance of common trading rules is evaluated using historical data. However, the fact that these rules are common or popular is *prima facie* evidence that they have been profitable in past usage. Evaluating these rules using historical data is thus little more than *ex post* model-fitting.

Genetic programs avoid data-snooping because the rules constructed are drawn from the space populated by combinations of simple arithmetic and logical operators. These rules are mechanically optimized using historical data, and then tested using a different set of historical data. Therefore, although the specification of these rules is dependent upon their historical performance, their merit is judged using data not available during their construction.

The trading rules used in NWD and AK were binary, i.e. they could only indicate two states for the investor. These states were variously mapped to trading positions of long/short, long/neutral, or neutral/short. While binary positions may make sense in equities, they are problematic in futures markets as there is no physical asset being held, and short positions are taken as easily and as often as long positions. Therefore, this article uses trinary trading rules, in which the rule can indicate long, neutral, or short positions.

Figure 1 is an example of a simple moving average rule represented as a genetic program. `XR` is a *root node* that requires two *subnodes*, which for this rule are the inequality operators

$>$ and $<$. The real values 1.02, 0.98, and 10, as well as the data `CPR` (closing price) are *terminal nodes*, nodes which do not have subnodes. Nodes $>$, $<$, `***` (multiplication) and `AVG` (moving average) are *operators*, nodes that act as functions, and require one or more subnodes as inputs. Operator nodes may be either real-valued, such as `***` and `AVG`, or logical-valued, such as $>$ and $<$. `XR` is a trinary operator whose state is a function of the states of its subnodes (in this case, the subnodes are $>$ and $<$), as displayed in table 1, where long, neutral, and short positions are indicated by 1, 0, and -1, respectively. Rule 1 is a simple moving average rule which takes a long position if the closing price is 2% above the 10 day moving average of closing prices, and a short position if the closing price is 2% below the 10 day moving average price, and is neutral otherwise.

Table 1: State of `XR` given the subnode states

`XR`	Subnode 1	Subnode 2
0	`TRUE`	`TRUE`
1	`TRUE`	`FALSE`
-1	`FALSE`	`TRUE`
0	`FALSE`	`FALSE`

The choice of nodes in building the genetic programs is similar to those used in NWD and AK. Terminal nodes (those that take no arguments) may be real [-10,10], boolean (`TRUE`, `FALSE`), or return price data: `OPR`, `HPR`, `LPR`, `CPR`, `VOL`, and `OI` represent the opening, high, low, and closing price, and the daily volume and open interest. Function nodes can be the arithmetic operations, $+$, $-$, $\times$, and $\div$, boolean operators, `IF-THEN-ELSE`, `AND`, `OR`, `NOT`, inequalities, $<$, $>$, and the 1-norm (absolute difference). Additionally, four functions are included that operate on lagged data, each of which requires two arguments, a data series (`OPR`, `HPR`, `LPR`, or `CPR`) and a real value, k, which indicates the number of prior observations over which to operate. `LAG` returns the k-day lagged value, `MIN` and `MAX` return the minimum and maximum values over the k days periods, and `AVG` returns the k-day average. `MND` and `MXD` are similar to `MIN` and `MAX` except that they return the

Table 2: Nesting of Common Technical Indicators within Functional/Terminal Node Sets.

Technical Indicator	Nested in Node Sets	
	AK	Current
Trend Lines	-	X
Support/Resistance	X*	X
Channel Line	-	X
Percentage Retracements	X	X
Speedlines	X	X
Gaps	-	X
Head and Shoulders	-	-
Double Tops/Bottoms	-	-
Triangles	-	-
Moving Average	X	X
Envelopes	X	X
Bollinger Bands	X*	X
Momentum	X	X
RSI	X	X
Stochastics	-	X
% R	-	X
MACD	X	X
Candlesticks	-	X

* Although these indicators can be based only on closing prices, high and low prices are most commonly used.

number of days since the lowest (highest) value in the last k days. Table 2 lists the first eighteen technical trading indicators in Murphy. The set of rules that can be constructed using the operator set in this article encompasses most common technical rules, and is significantly expanded from NWD and AK.

The evolutionary process used to generate optimal trading rules is the defining characteristic of genetic programming. To start the process, a population of rules is randomly generated. Each of these rules is evaluated for its 'fitness'–such as high profitability or low risk. With a probability proportional to each rule's fitness rank in the population, rules are chosen to participate in genetic operations, such as recombination, and the resulting rules constitute the next generation of rules. This three-step process (evaluate, select,

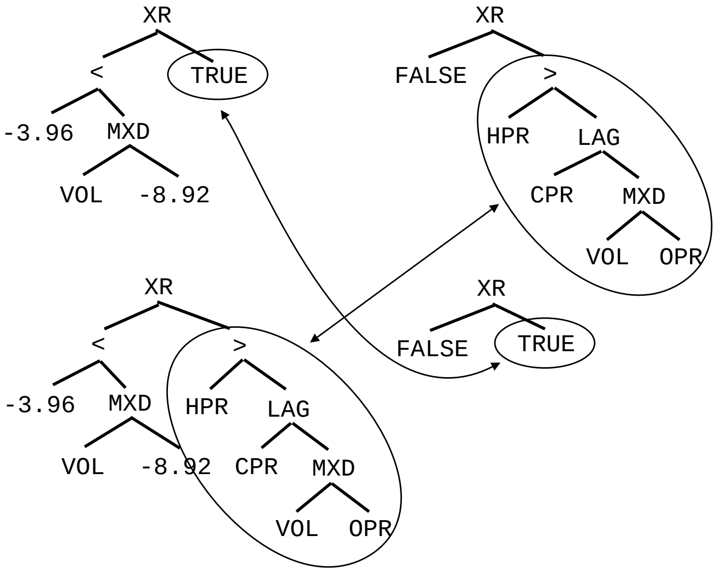

Figure 2: Recombination of Two Technical Trading Rules

operate) is repeated until convergence or the maximum number of iterations is reached.

In this study, only two genetic operators are used to create rules. In *reproduction*, rules from the parent generation are inserted into the child generation unchanged. In *recombination*, two parent rules are chosen, and sub-trees are randomly chosen from each parent rule and exchanged. Figure 2 shows the recombination of two parent rules into two child rules. While many other genetic operations have been proposed, reproduction and recombination are the two most common, and additional rules typically offer little benefit. (Koza)

Because rules are selected for operation based upon their fitness, the specification of the fitness measure is crucial for the success of genetic programming. Two fitness measures are used to generate the rules in this study, gross profitability and the ratio of profitability to maximum intermediate loss. These two criteria will be explained in the next section.

In order to prevent over-fitting, the rules are generated using two sets of futures price data, as in AK. Rules are evaluated for selection and operation based upon their fitness in 'training' data, which are prices for two years for a given commodity. After each generation is evaluated using the training data, the fittest rule is applied to the 'selection' data, which is also two years of price data. If this rule is fitter than the previous rules evaluated with selection data, it is retained.

An initial generation of 20,000 random rules are created. Each successive generation consists of the fittest rule from the previous generation, 1,999 randomly-chosen rules are inserted unaltered (reproduction), and the remaining 18,000 are the product of recombination of randomly-chosen pairs. Analogously to the evolutionary process, rules are not truly randomly chosen. Instead, the probability that a rule will be chosen for insertion or recombination is a function of its fitness. Specifically, the probability is a function of a rule's rank within the population,

$$p_i = \frac{r_i}{\sum_j r_j^3} \tag{1}$$

where p_i is the probability that the i^{th} rule will be chosen, and i is the ordinal rank of the rule, with $i = N$ the most fit, and $i = 1$ the least fit. Successive generations are created until the 'best' rule (when applied to the selection data) doesn't change for five generations, to a maximum of 200 generations.

Because GP cannot guarantee convergence, either locally or globally, the quality of a solution is a monotonic function of its computational cost; as larger populations of larger rules are allowed to evolve longer, the probability of convergence increases. Balancing this need is the time required for estimation. The population size is 20000 rules, each of which is constrained to 50 nodes. In the initial rule generation, the rules are constrained to be no more than 10 levels deep, but in recombination, the rules can grow to be 16 levels deep. To further improve the results, 20 optimizations are performed over each set of training/selection data, differing only in the seed value to the random number generator, and the best rule of the twenty is used in the out-of-sample testing. The out-of-sample evaluation uses one year of prices of the same maturity month as the testing and selection data, but from the following contract year.

2 Trading Strategy Evaluation

Net profits are the simplest and most common measure of the usefulness of a trading strategy. The leveraged nature of futures contracts makes the use of simple return-based measures of performance more difficult, as it is unclear what denominator should be used in computing the return. One could assume that no leverage is possible, although this seems a very strong assumption, especially as leverage is frequently cited as an advantage of futures markets. Alternatively, one could use the margin requirement as the denominator. This is also problematic, as US Treasury Bills can be pledged as collateral, meanwhile still accruing interest for the futures-holder, which reduces the forgone interest of holding futures to zero.

For these reasons, simple profitability is used to measure fitness instead of returns. The fitness measure used is

$$\pi = \sum_{t}^{T} \left(p_{t+1} - p_t\right) I_t - \phi \operatorname{abs}\left(I_t - I_{t-1}\right) \tag{2}$$

where $I_t \in [-1, 0, 1]$ is a trinary signal variable that indicates the trading position at time t and ϕ is the transactions cost. As suggested by Neely, Weller and Dittmar, higher transactions costs discourage rules which over-trade, which may be a symptom of over-fitting. They recommend using a transaction cost that is higher than otherwise may be realistic for training and selection, and a more realistic rate for out-of-sample testing. Lukac, Brorsen and Irwin suggest commissions of $100 per round-turn, accounting for both commissions and liquidity costs, although they suggest that this number is likely too high. Therefore, transactions costs of $200 per round-trip are used for training and selection, and $50/round-trip are used in out-of-sample testing.

3 Simulation Results

To test the power of the genetic programming algorithm at various sample sizes, a simulation study of the method was performed. The technical trading rule in figure 1 can be seen as a description of a threshold auto-regressive model with three regimes,

$$
\begin{aligned}
y_t &= y_{t-1} + \mu_t + \epsilon_t \\
\mu_t &= \begin{cases} a & : \; y_{t-1} < (1-k)\frac{1}{n}\sum_{i=1}^{n} y_{t-i-1} \\ b & : \; y_{t-1} > (1+k)\frac{1}{n}\sum_{i=1}^{n} y_{t-i-1} \\ 0 & : \; \text{otherwise} \end{cases} \\
\epsilon_t &\sim iid(0, \sigma^2)
\end{aligned} \tag{3}
$$

For this simulation study, $a = -0.5$, $b = 0.5$, $n = 10$, $k = 0.02$ and $\sigma^2 = 1$. Using the model in equation 3, 100 data series are generated using lengths 750, 1500, 2250, and 3000 observations, approximately corresponding to three, six, nine, and 12 years of futures price data. Each simulated data set is subdivided into three equally-sized subsamples, called the training, selection, and testing data. Following the procedure of AK and NWD, each generation of rules is evolved and evaluated using the training data. The best rule of that generation is then applied to the selection data, and retained. Then, the next generation of rules is evolved and evaluated using the training data; the best rule is applied to the selection data, if it is fitter than the best rule from the prior generation, the new rule is retained, otherwise the rule of the prior generation is retained. For each simulation, the process terminates when 25 generations fail to improve the retained rule, or a maximum of 200 generations. The resulting 'best' rule is then evaluated using the testing data to measure its out-of-sample fit.

Table 3: Results of Estimating Models with Simulated Price Data

	Simulation Length			
	N=250	N=500	N=750	N=1000
Daily Returns Using the Rule in Figure 1				
Mean	0.3237	0.3242	0.3136	0.3061
St. Dev.	0.1520	0.1436	0.1369	0.1342
Median	0.3215	0.2943	0.3108	0.3180
Daily Returns Using *Ex Ante* Optimal Trading Rules				
Mean	0.2741	0.3031	0.2869	0.2822
St. Dev.	0.1846	0.1533	0.1406	0.1340
Median	0.2763	0.2930	0.2901	0.2888
Proportion of Coincident Daily Position Identification				
‘True’ Rule	**Position of Estimated Rule**			
Long	0.9356	0.9500	0.9505	0.9420
Neutral	0.1537	0.1003	0.0857	0.1059
Short	0.9360	0.9646	0.9682	0.9722
Correlation between Average Daily Returns of ‘True’ and Estimated Rules				
Pearson	0.7981	0.9523	0.9475	0.9620
Test of Equality of Means of ‘True’ and Estimated Rules				
Test Statistic	1.5522	0.7189	0.9747	0.8976
p-value	0.1206	0.4722	0.3297	0.3694

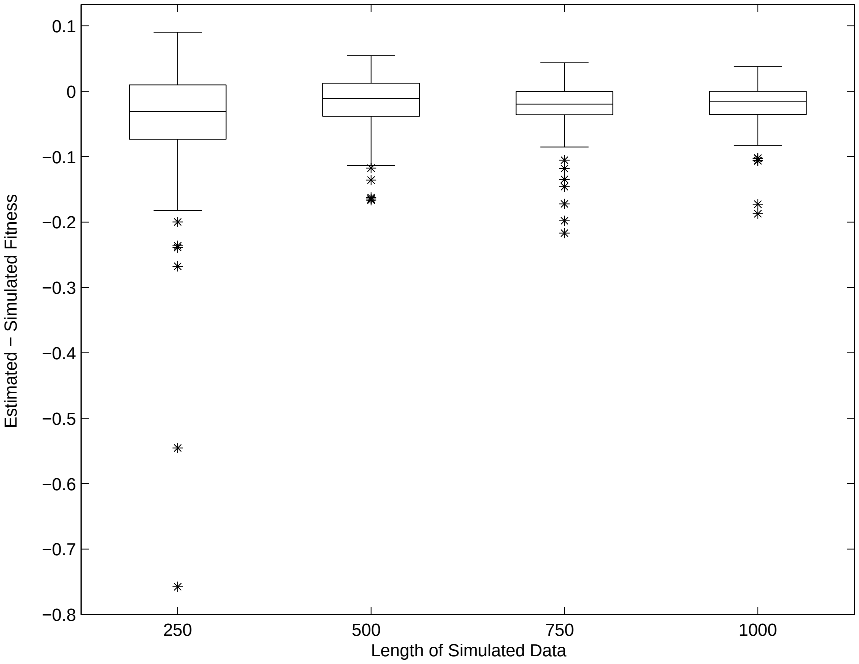

Figure 3: Comparison of Trading Rule Performance by Length of Simulated Data

4 Results

Optimal trading rules are estimated for the 24 commodities listed in table 4. Front-month futures prices are used to create a rolling price series. Observations that occur during the delivery month are excluded. The futures prices used are drawn from the CRB futures price database. Prices from January 2, 1980 through December 29, 2000 were used. Subtracting the four years allocated to training and selection data, 204 months of data were available for out-of-sample testing. Crude oil, unleaded gasoline heating oil began trading during 1983, so that data is only used from January 6, 1984, resulting in 156 observations. S&P 500 futures began trading during 1982, therefore only prices since January 3, 1983 were used, resulting in 168 monthly out-of-sample observations.

The first four columns of table 4 report the mean monthly return, monthly return standard error, the Student's t-statistic of the one-tailed test of $r \leq 0$, and the number of months with positive returns. Returns are reported as average gross monthly profit minus

Table 4: Out of Sample Returns to GP Technical Trading Rules

Commodity	r (%)	σ_r	t	$n > 0$	Trades/ Month	% Long Correct	% Short Correct
Chicago Board of Trade							
Treasury Bonds	10.58	9.28	1.14	84	0.52	50.71	49.38
Wheat	-12.44	11.36	-1.10	79	0.25	51.48	52.37
Corn	2.07	14.31	0.14	90	0.50	50.40	53.19
Soybeans	5.46	14.37	0.38	94	0.74	52.15	52.55
Soybean Meal	-6.97	13.08	-0.53	91	0.24	50.21	51.74
Soybean Oil	-12.96	15.03	-0.86	80	0.15	50.23	52.97
Chicago Mercantile Exchange							
S&P 500	8.05	4.17	1.93^{**}	91	1.16	46.51	46.83
Japanese Yen	-2.22	1.39	-1.60	85	0.19	51.03	53.25
Canadian Dollar	-0.66	0.77	-0.86	64	0.08	50.96	49.84
British Pound	0.67	3.77	0.18	87	0.32	52.54	51.07
Lean Hogs	29.23	12.23	2.39^{***}	104	0.35	48.22	48.90
Pork Bellies	6.66	7.86	0.85	100	0.63	49.51	48.96
Live Cattle	5.83	10.71	0.54	100	0.20	52.18	49.36
Feeder Cattle	-1.92	10.80	-0.18	96	0.19	51.28	51.05
New York Mercantile Exchange							
Crude Oil	-3.59	5.74	-0.63	65	0.51	51.15	50.65
Heating Oil	2.71	5.05	0.54	74	0.30	51.47	49.66
Unleaded	6.28	6.69	0.94	68	0.25	53.55	49.62
Gold	3.68	5.87	0.63	102	0.41	49.15	54.68
Silver	12.45	11.98	1.04	97	0.53	50.43	51.77
Copper	-11.81	11.16	-1.06	84	0.35	52.80	51.11
New York Board of Trade							
Coffee	-62.73	31.22	-2.01	83	1.20	51.78	52.27
World Sugar	14.12	13.19	1.07	101	0.18	50.36	51.63
Cocoa	1.11	6.30	0.18	92	0.48	48.80	50.90
Cotton	11.53	13.73	0.84	86	0.56	49.31	50.97
Futures Portfolio	0.52	0.89	0.59	105			
Equity Index Returns							
Russell 2000	0.42	0.39	1.08	121			
S&P 500	0.93	0.32	2.92^{***}	132			
Dow Jones	1.02	0.33	3.13^{***}	133			

Notes: r is the average monthly return, σ_r is the standard error of r, t is Student's t-statistic, and $n > 0$ is the number of months in which $r > 0$. Crude oil, unleaded, and heating oil have 156 monthly observations, all other commodities have 204 observations. ** and *** denote rejection of the null hypothesis $r \leq 0$ at the 5% and 1% levels.

transactions costs divided by the exchange minimum initial margin.

In contrast to much of the previous research in commodity markets, these rules find little evidence of profitability for technical trading strategies. Of the 24 commodities studied only two, CME Lean Hogs and CME S&P 500 futures, produced rules for which the trading returns were statistically positive at the 5% level, which is only marginally above the 1.2 that would be expected with 24 tests at the 5% level. Examining the remainder of the commodities reveals no particular patterns in success or failure. Of the financial futures, three of the five are positive; only one is significant. Of the agricultural and foodstuffs futures, eight of 13 have positive returns, with one significantly positive. Of the remaining six metals and energy futures, four are positive, but none are significant.

The fifth column reports the average number of trades per month for the rules generated. Cocoa and the S&P 500 produce rules with the highest frequency of trades, but are still average just over one trade per month. Many other commodities' rules trade only every third month, on average. This indicates that the $200 transactions cost used in rule training and selection was too successful in discouraging the evolution of over-fitted and oft-transacting rules.

The remaining two columns report the proportion of days on which the trading rules were correct, conditioned on holding long or short positions.[4] If the prices and positions were generated randomly, these statistics would be expected to be near 50%, which almost all of them are. The two commodities with statistically positive returns, lean hogs and the S&P 500, both have the *lowest* correct rates.

The results of the individual rules indicate no success in generating consistent profits from technical analysis in out-of-sample testing. While rules are generated that return statistically significant profits for two of the commodities, given that 24 commodities were

---

[4] These are actually the percentage of days that are 'not incorrect'; i.e. a day in which prices did not change is not considered a failure for either long or short positions. Since futures prices change in discrete increments, there is a positive probability of a zero return.

Table 5: Correlation of Futures Portfolio and Equity Index Returns

	Futures	Russell 2000	S&P 500	DJIA
Futures	1.0000	0.0166	0.0565	0.0284
Russell 2000	0.0166	1.0000	0.8179	0.7654
S&P 500	0.0565	0.8179	1.0000	0.9543
DJIA	0.0284	0.7654	0.9543	1.0000

tested, such results are not surprising.

To compare the results of a fund of technically-traded futures, a portfolio of futures is created in which 30% of the assets are devoted to initial margin, as in Lukac, Brorsen, and Irwin, and all of the assets are held in US Treasury Bills. The return for such a portfolio is equal to the T-Bill rate plus 30% of the average of the individual commodity returns.

The returns to such a portfolio are also reported on Table 4 along with the returns of three popular equity indices over a similar time period. The returns of the futures portfolio are higher than the returns on the Russell 2000 index, but are lower than the S&P500 and the Dow Jones Industrials. Further, the standard deviation of the futures portfolio is much higher than any of the equity indices.

Table 5 reports the correlations among the futures portfolio and the equity indices. As expected, the futures portfolio does exhibit a low correlation with equity returns, between 1.6% and 5.6%.

These results offer little support for the use of technical analysis in commodity markets. While previous research in commodity markets has indicated the ability to generate profits in excess of transactions costs, when *ex ante* optimal trading rules are chosen and used, they are not capable of consistently generating profits in excess of transactions costs.

5 Summary and Conclusion

Technical analysis has a long history in commodity markets, and remains very popular in spite of a lack of theoretical foundation. Because of this, a rich literature exists on whether technical analysis is actually profitable. Most studies have failed to find profitability for technical trading strategies.

As pointed out by NWD and White, previous studies suffer from data-snooping biases introduced when historically popular trading rules are applied to historical, though 'out-of-sample' data. One remedy for data-snooping biases is to produce *ex ante* optimal technical trading rules from primitive operators, as in NWD and AK. This study uses genetic programming, an evolutionary method for algorithmic design, to evolve technical trading rules for 24 futures markets. When evaluated using data not available to the optimization process, only two of the 24 markets could be traded at a statistically significant level of profit.

A portfolio of these commodity returns was constructed that produced returns that were in the range of broad equity index returns over the time period studied. While the correlation of the portfolio returns to the equity index returns is low, the futures portfolio has a standard deviation two to three times higher than the equity indices.

The results of this study are primarily limited by the constraints of genetic programming itself. Evolutionary methods such as genetic programming are not guaranteed to find global, or even local, optima. Further, as pointed out by Cooper and Gulen, data-snooping biases can be induced through the process used to choose the data lengths, the operations available, or almost any other parameter of the optimization process. The use of $200 as a transaction cost for model fitting demonstrates the dangers of the arbitrary selection of model parameters.

While the results of this study do not preclude the existence of profitable technical trading strategies, these results do undermine the notion that technical analysis is clearly profitable in commodity trading.

One obvious avenue for the extension of this work is to explicitly incorporate various trading rule families (such as momentum, or RSI, or others listed in table 2) into the set of function nodes. This would allow a more direct examination of the trading rules currently in practice with the broader set of rules available to the genetic programming method.

References

Alexander, S. S., "Price Movements in Speculative Markets: Trends or Random Walks, No. 2," in Paul Cootner, ed., *The Random Character of Stock Market Changes*, MIT Press, 1964, pp. 338–372.

Allen, Franklin and Risto Karjalainen, "Using genetic algorithms to find technical trading rules," *Journal of Financial Economics*, 1999, *51*, 245–271.

Brorsen, B. Wade and Scott H. Irwin, "Futures Funds and Price Volatility," *Review of Futures Markets*, 1987, *7*, 118–135.

Campbell, John Y., Andrew W. Lo, and A. Craig MacKinlay, *The Econometrics of Financial Markets*, Princeton, NJ: Princeton University Press, 1997.

Cooper, Michael and Huseyin Gulen, "Is Time-Series Based Predictability Evident in Real-time?," November 2001. Working Paper, Purdue University, Krannert School of Management.

Fama, Eugene and Marshall Blume, "Filter Rules and Stock Market Trading," *Journal of Business*, 1966, *3*, 226–241.

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

Lui, Yu Hon and David Mole, "The Use of Fundamental and Technical Analyses by foreign exchange dealers: Hong Kong evidence," *Journal of International Money and Finance*, 1998, *17*, 535–545.

Lukac, Louis P. and B. Wade Brorsen, "A Comprehensive Test of Futures Market Disequilibrium," *The Financial Review*, 1990, *25* (4), 593–622.

___ , ___ , and Scott H. Irwin, "Similarity of Computer Guided Technical Trading Systems," *Journal of Futures Markets*, 1988, *8*, 1–13.

___ , ___ , and ___ , "A Test of Futures Market Disequilibrium using Twelve Different Technical Trading Systems," *Applied Economics*, 1988, *20*, 623–639.

Malkiel, Burton G., *A Random Walk Down Wall Street*, 4th ed., New York: Norton, 1985.

___ , "Efficient Markets Hypothesis," in P. Newman, M. Milgate, and J. Eatwell, eds., *New Palgrave Dictionary of Money and Finance*, London: Macmillan, 1992.

Menkhoff, L., "Examining the use of technical currency analysis," *International Journal of Finance and Economics*, 1997, *2*, 307–318.

Murphy, John J., *Technical Analysis of the Financial Markets*, revised ed., New York: New York Institute of Finance, 1999.

Neely, Christopher J., Paul Weller, and Rob Dittmar, "Is technical analysis in the foreign exchange market profitable? A genetic programming approach," *Journal of Financial and Quantitative Analysis*, 1997, *32* (4), 405–426.

Oberlechner, Thomas, "Importance of Technical and Fundamental Analysis in the European Foreign Exchange Market," *International Journal of Finance and Economics*, 2001, *6*, 81–93.

Osler, Carol, "Identifying Noise Traders: The Head-and-Shoulders Pattern in U.S. Equities," 1998. Staff Report No. 42, Federal Reserve Bank of New York.

Sweeney, Richard J., "Some New Filter Rule Tests: Methods and Results," *Journal of Financial and Quantitative Analysis*, 1988, *23* (3), 285–300.

Taylor, Mark P. and Helen Allen, "The Use of Technical Analysis in the Foreign Exchange Market," *Journal of International Money and Finance*, 1992, *11* (3), 304–314.

White, Halbert, "A Reality Check for Data-Snooping," *Econometrica*, 2000, *68* (5), 1097–1126.