2010 International Conference on Business and Economics Research
vol.1 (2011) © (2011) IACSIT Press, Kuala Lumpur, Malaysia

Genetic Algorithm for Trading Signal Generation

Solution to trader's dilemma: Is it right time to trade?

Mehul N Vora
Tata Consultancy Services (TCS) Limited
Mumbai, India
e-Mail: mehul.vora@tcs.com

***Abstract*— Technical indicators are widely used by traders and analysts in stock and commodity markets to predict market movements and to identify trading opportunity thereby enhancing trading profitability. There are more than 100 indicators used in practice today to understand the market behavior. Identification of the right combination of indicators for optimal portfolio performance has always been a challenging problem. In this paper, we propose a supervised learning approach to generate trading signals (buy / sell) using a combination of technical indicators. Proposed system uses a genetic algorithm along with principle component analysis to identify a subset of technical indicators which detect rise and fall of the market with greater accuracy and generates realistic trading signals. The performance of this new algorithm was tested using trading data obtained from National Stock Exchange (NSE), India. Simulation shows the enhanced profitability for the proposed trading strategy.**

Keywords - Technical Analysis; Indicator Selection; Supervised Learning; Genetic Algorithm

I. INTRODUCTION

It has always been challenging tasks for a trader – *what to trade* and *when to act*. This is primarily because of the uncertainties involved in the movement of the market. Movement in the market is caused by several reasons including fundamentals, financial, economical, political as well as human parameters. Therefore identification of the right stock and determination of the right time to trade become increasingly difficult. One popular approach, to answer the trader's problem of *what* and *when*, is Technical Analysis. Number-driven technical analysis is a statistical study of past trading history (price movements, trading volumes, volatility etc.) for the purpose of forecasting probable future market trends [1]. Philosophy of technical analysis is based on the assumption that stock prices move in trends and history tends to repeat itself. Prices evolve dynamically in a highly nonlinear fashion over time. Some market movements in price and volume are significant as they contribute to the formation of a specific pattern whereas others are merely random fluctuations. The purpose of technical analysis is to identify those nonlinear patterns in the noisy time series of price data.

Number-driven technical analysts extensively use technical indicators, which are typically mathematical transformations of price or volume or both to quantify the market trend. There are hundreds of indicators in use today, with new indicators being created every week. The diverse choice of technical indicators available to a trader possesses a problem in a new dimension -- selection of appropriate indicator(s) for the analysis/trading. While some indicators are used for generation of trading signals (buy/sell) where as others are used to confirm the signal with the trend of price movements in the market.

Technical indicator analysis has attracted many researchers from different fields. For example, [2] have investigated the relationship between momentum indicators and kinetic energy theory. A number of machine learning techniques have also been used over the past decade to forecast the market movements. Artificial Neural Networks (ANN) and Support Vector Machines (SVM) are by far the most widely used techniques ([3], [4], [5]). Other methods that have been used include Bayesian Belief Networks [6], evolutionary algorithms like Genetic Algorithms (GA) [7] and Genetic Programming (GP) [8], and fuzzy logic [9]. [10] and [11] suggest using multiple machine learning techniques to improve the prediction power. One school of thought tends to integrate the rule-based technique with supervise learner like ANN or GP to predict the direction of stock and derivative markets ([7], [12]).

In this paper, we present a new algorithm to identify the appropriate trading horizon (buy / sell signals) using a combination of technical indicators identified by a genetic algorithm. The rest of this paper is organized as follows: Section II describes our approach to technical trading as technical indicator selection problem. The simulation results and the performance of the proposed system are described in Section III. In the end, we conclude in Section IV. List of indicators used in this study are summarized in the Appendix following the references.

II. TECHINCAL TRADING: APPROACH

Thorough analysis of a secondary market to identify trading opportunity using multiple technical indicators can produce some reasonable buy / sell suggestions. But how do we determine a proper subset of technical indicators and how good is this combination? To answer this question empirically and systematically, we must first develop a method for automating the identification of a combination of technical indicators which can truly determine the trading pattern from the price-volume data; that is, we require an indicator selection algorithm. Once such an algorithm is developed, it can be applied to a large number of stocks over different time horizon to determine the efficacy of various technical indicators. In our approach we treat technical trading as two step process: (A) Indicator Selection and (B) Generation of Trading Signal (buy / sell)

A. *Indicator Selection*

We have modeled the indicator selection problem as a classification problem i.e. separating buy position (just beginning or upcoming upward movement in the market) and sell position (just beginning or upcoming downward movement in the market) from the random fluctuations in the market. To quantify the market trend and to identify trading patterns in the market movement, we use following indicators ([1], [19], [20]): moving averages (simple and exponential), average directional index (ADX), moving average convergence and divergence (MACD), relative strength index (RSI), stochastic oscillators, StochRSI, Bollinger bands (%b and bandwidth), money flow index (MFI), accumulation / distribution index (ADI), Chaikin oscillator as described in the appendix. These indicators are computed by using stock price and volume overtime.

The problem of identifying a subset of technical indicators for optimal portfolio performance can be viewed as an optimization problem. Genetic Algorithm (GA), formally introduced by John Holland in the late 60s, is an adaptive heuristic search algorithm premised on the evolutionary ideas of natural selection and genetics (Darwin's theory of evolution) ([13], [14]). Ever since, it has been widely studied, experimented and applied in diverse fields including finance as an optimization tool. The fundamental idea behind the evolutionary model is that the genetic pool of a given population potentially contains the solution, or a better solution, to a given adaptive problem. The genetic combination leading to this better solution is split between several individuals. During reproduction (crossover and mutation), new genetic combinations occur and, finally, an individual can inherit a better gene from both parents leading to an optimal solution.

This paper presents a new genetic algorithm specifically designed for technical trading using selected indicators. The procedure constructs a population of chromosomes, each of which represents a potential candidate solution. Fit solutions are retained and manipulated by mathematical operators that model biological processes. The outline for this new GA developed for trend detection and technical trading is shown in Figure 1. The important components of the GA are described below.

1. Population Encoding: Generate a random population of n chromosomes. Each chromosome represents a candidate solution.
2. Fitness Evaluation: Evaluate the fitness function *f(x)* of each chromosome *x* in the population.
3. End Criterion: If the end condition is satisfied then stop and return the best solution in the current population else continue.
4. Boosting: Focus on data points that are difficult to classify by recalibrating their weights.
5. Reproduction: Apply reproduction and exchange genetic information as per the following steps.
   a. Selection: Select two parent chromosomes according to their fitness - the better fitness, the more chance to be selected
   b. Crossover: With a crossover probability, cross over the parents to form new offspring
   c. Mutation: With a mutation probability, mutate new offspring at random locus / position in the chromosome.
6. New population: Use newly generated population for the next generation (iteration).
7. Loop: Go to step 2

Figure 1. Outline for Genetic Algorithm for Technical Trading

1) *Population*

Mimicking nature, each indicator considered to be a gene. A chromosome is a single large macro molecule of DNA, which contains many genes. A population consists of a large number of randomly generated chromosomes. Each chromosome of the population is a unique subset and represents a potential solution. For example, the following represents a population of four individuals: {[1, 3], [4], [2, 3], [1, 2, 4]} where numbers 1 through 4 represents different technical indicators.

2) *Fitness Function*

Uniqueness of the proposed genetic algorithm for technical trading is its fitness function. We have developed a new approach to calculate fitness by the way of Principal Component Analysis (PCA). PCA is widely used technique for dimensionality reduction of collinear data. The technical indicators tend to be highly collinear because all of them are derived from the price and volume data.

The first principal component (PC) is formed by determining the direction of the largest variation in the original measurement space, and modeling the first PC with a line fitted by linear least-squares that passes through the center of the data. The second largest PC lies in the direction of the next largest variation, passes through the center of the data, and is orthogonal to the first PC. The third largest PC lies in the direction of the next largest variation and passes through the center of the data; it is orthogonal to the first two PCs, and so forth. The key idea of PCA is not only finding the important relationships among the variables, but preserving those relationships in a lower dimensionality space that is easier to work with ([15], [16]).

Each PC describes a different source of information because each defines a different direction of variance in the data. The orthogonality constraint ensures that there will be no correlation between PCs. A measure of the amount of information conveyed by each PC is expressed in terms of the eigenvalues of the covariance matrix of the data, since the eigenvectors of the covariance matrix are the axes of maximum variance. For this reason, PCs are arranged in order of decreasing eigenvalues. Experience shows that only the first few PCs would convey information about the signal [17].

The fitness function of the GA for technical trading emulates human pattern recognition through machine learning by assessing the information content of a set of indicators by characterizing the amount of class separation on a PC plot built from those indicators. To facilitate characterizing the amount of class separation in M-dimension PC space, the centers of two clusters (buy / sell positions) are calculated as follows:

Let n_1 = buy position data point; n_2 = sell position data point; and PC_{k_i}= k^{th} principle component score for i^{th} data point.

$$C_1 = \left(\frac{1}{n_1} \sum_{i=1}^{n_1} PC_{1_i} \ , \ .. \ , \ \frac{1}{n_1} \sum_{i=1}^{n_1} PC_{M_i} \right)$$

$$C_2 = \left(\frac{1}{n_2} \sum_{j=1}^{n_2} PC_{1_j} \ , \ .. \ , \ \frac{1}{n_2} \sum_{j=1}^{n_2} PC_{M_j} \right)$$

Then the fitness of an individual is calculated as the Euclidian distance between the two centers. Greater the distance between two centers, better the fitness of an individual.

3) Boosting

We have used Adaboost algorithm, the most widely used algorithm for boosting, that classify its outputs applying a simple learning algorithm (weak learner) to several iterations of the training set where the misclassified observations receive more weight [18]. A weak learning mechanism computes the average distance of a data point from its respective cluster center. A data point with greater distance from the cluster center will weigh more than the data point closer to the respective center. The fitness function is able to focus on data points that are difficult to classify by adjusting their weights over successive generations thereby helping to steer the genetic algorithm in the search for an optimal solution.

B. Generation of Trading Signal

To generate trading signal, we have used the loading matrix generated while calculating principle component scores using the subset of technical indicators identified by the genetic algorithm. The loading matrix conveys the information about calculating PCs from the original technical indicators.

Let D_1 = Euclidian distance of the prediction set data point from C_1; D_2 = Euclidian distance of the prediction set data point from C_2. Then one approach in generating trading signal is as follows:

- Buy Signal if $D_1 < D_2$
- Sell Signal if $D_2 < D_1$

The down-point of this approach is that it does not isolate the true trading signals from random fluctuation of the market. The data point will always be classified as either Buy signal or Sell signal. Second approach is to calculate the average distance for each cluster using training set data points and generate the trading signal as follows. Let n_1 = buy position data point in the training set; n_2 = sell position data point in the training set; d_i = Euclidian distance of i^{th} data point of training set from the respective cluster center; and K_1 and K_2 are constants. Then generate signal as follows:

- Buy Signal if $D_1 < D_2$ and $D_1 \leq K_1 * \frac{1}{n_1} \sum_{i=1}^{n_1} d_i$
- Sell Signal if $D_2 < D_1$ and $D_2 \leq K_2 * \frac{1}{n_2} \sum_{j=1}^{n_2} d_j$
- Ignore otherwise

III. Technical Trading Performance

To demonstrate the efficiency and the efficacy of the proposed technical trading, we have chosen the trade data from NSE (National Stock Exchange), India. In the past, most of the work in this area has focused on the American and European markets; there exists very little published work on the Indian Stock Exchanges. We have used the trading data from 1st January, 2006 to 30th June, 2010. The data were collected for the following companies: Cipla, Dabur, GAIL, Hero-Honda, ITC, ONGC, Suzlon Energy Ltd., Tata Tea, Tata Consultancy Services (TCS) Ltd., and UniTech. The data used for this study were obtained from the NSE website in the form of '*bhavcopy*'. Bhavcopy provides the stock-wise data for the opening, highest, lowest and closing values of the stock prices as well as total trading value and volumes for the trading day. Based on this trade data, the data for typical price and weighted-close price were derived as follows:

$$Typical\ Price = \frac{High + Low + Close}{3}$$

$$Weighted\text{-}Close\ Price = \frac{High + Low + Close + Close}{4}$$

The training-set for the proposed genetic algorithm consist of the following technical indicators calculated from the trade data (bhavcopy) as well as derived data (typical and weighted-close price) from January 1st, 2006 to December 31st, 2009 as described in the appendix: SMA, EMA, ADX, MACD, RSI, Stochastic %K, Stochastic %D, StochRSI, Bollinger bands (%b and bandwidth), MFI, CLV, ADI and Chaikin oscillator. Each indicator was calculated multiple times using different look-back (historical) period.

Using the subset of indicators identified by the GA, PCA was performed and a loading matrix was calculated for each stock under study. This loading matrix was used to calculate the principle component scores for the data-points in the prediction set. The prediction-set data comprised of the data from January 1st, 2010 to June 30th, 2010. A trading simulation for this period was carried out for each stock using the second approach described above for signal generation but restricting short selling and limiting portfolio. Short selling restriction implies that necessary a sell trade would be executed only after a successful buy trade execution. Limiting portfolio restriction implies that a buy trade would be followed by a sell trade only i.e. two buy (or sell) trades would not be executed in succession.

The performance of technical trading was evaluated in terms of strategy realization as defined below. This realization was also compared with no-strategy realizations i.e. buy on the first day of trading period and sell in the end without applying any strategy for trading. Table I represents the performance of our approach to technical trading.

$$Strategy\ Realization = 100 * \left(\sum_{Trades} \frac{Sell\ Price - Buy\ Price}{Buy\ Price} \right)$$

TABLE I. PERFORMANCE OF TECHNICAL TRADING

Stock	% No-Strategy Realization	% Strategy Realization	% Strategy Advantage	% Winning Trades	Max. Draw-downs
Cipla	0.28	15.26	5321.99	85.71	(3.07)

Dabur	30.48	24.39	(20.00)	100.00	0.00
GAIL	14.31	16.93	18.30	85.71	(0.64)
Hero Honda	19.59	24.82	26.67	84.62	(6.57)
ITC	20.42	26.74	30.95	100.00	0.00
ONGC	11.23	27.19	142.11	83.33	(3.57)
Suzlon	(35.56)	4.31	112.11	66.67	(2.52)
Tata Tea	26.91	36.10	34.18	70.00	(0.15)
TCS	(4.05)	10.73	365.06	69.23	(3.49)
UniTech	(10.21)	8.28	181.14	100.00	0.00

IV. CONCLUSION

In this paper, we have proposed one machine learning technique using technical analysis indicators for stock-trading. Technical analysis may well be an effective means for extracting useful information from market movements because of the fact that the joint distribution of prices and volume contains important information about the market sentiments. Genetic algorithm with a novel fitness function using principle component analysis can identify the true discriminatory indicators from the highly collinear data.

The above work is just a sample of what can be produced using such a system. Further adaptation techniques could be attempted; for example various fundamental parameters including political and economic factors which affect the stock market can also be taken into consideration other than just technical indicators as input variables.

REFERENCES

[1] M. J. Pring, Technical Analysis Explained: The Successful Investor's Guide to Spotting Investment Trends and Turning Points, 4th ed, New York:McGraw Hill, 2002.

[2] M. Ausloos, K. Ivanova, "Mechanistic approach to generalized technical analysis of share prices and stock market indices", The European Physical Journal B, Vol. 27, No. 2, 2002, pp. 177-187.

[3] E. Saad, D. Prokhorov, D. Wunsch, "Advanced neural-network training methods for low false alarm stock trend prediction", Proceedings of IEEE International Conference on Neural Networks, Vol. 4, June 1996, pp. 2021-2026.

[4] F. Tay, L. Cao, "Improved financial time series forecasting by combining support vector machines with self-organizing feature map', Intelligent Data Analysis Vol. 5, No. 4, September 2001, pp. 339–354.

[5] K. Kim, "Financial time series forecasting using Support Vector Machines", Neurocomputing, Vol. 55, 2003, pp. 307–319.

[6] K. Wolfe, "Turning point identification and Bayesian forecasting of a volatile time series", Computers and Industrial Engineering, Vol. 15, No. 1, December 1988, pp. 378–386.

[7] M. Dempster, C. Jones, "A real-time adaptive trading system using genetic programming", Quantitative Finance, Vol. 1, 2001, pp. 397–413.

[8] U. Markowska-Kaczmar, H. Kwasnicka, M. Szczepkowski, "Genetic Algorithm as a Tool for Stock Market Modelling", Artificial Intelligence and Soft Computing – ICAISC 2008, Volume 5097/2008, 2008, pp. 450-459

[9] O. Castillo, P. Melin, "Simulation and forecasting complex financial time series using neural networks and fuzzy logic", Proceedings of IEEE International Conference on Systems, Man, and Cybernetics, Vol. 4, 2001, pp. 2664–2669.

[10] H. Kim, K. Shin, "A hybrid approach based on neural networks and genetic algorithms for detecting temporal patterns in stock markets", Applied Soft Computing, Vol. 7, No. 2, March 2007, pp. 569-576.

[11] H. Ince, T. Trafalis, "Kernel principal component analysis and support vector machines for stock price prediction", Proceedings of IEEE International Joint Conference on Neural Networks, Vol. 3, 2004, 2053-2058.

[12] R. Tsaih, Y. Hsu,C. Lai, "Forecasting S&P 500 stock index futures with a hybrid AI system", Decision Support Systems, Vol. 23, No. 2, June 1998, pp. 161–174.

[13] D. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley Publishing Company, Reading:MA, 1989.

[14] M. Mitchell, An Introduction to Genetic Algorithms, MIT Press, Cambridge:MA, 1998.

[15] J. Jackson, A user's guide to principal components, John Wiley and Sons, New York, 1991.

[16] A. Cichocki, W. Kasprzak, W. Skarbek, "Adaptive Learning Algorithm for Principal Component Analysis with Partial Data", Cybernetics and Systems, Vol. 2, 1996, pp.1014-1019.

[17] E. Oja, J. Karhunen, "On stochastic approximation of the eigenvectors and eigenvalues of the expectation of a random matrix", Journal of Mathematical Analysis and Applications, Vol. 106, 1985, pp. 69-84.

[18] Y. Freund, R. Schapire, "A Decision-Theoretic Generalization of on-Line Learning and an Application to Boosting", Journal of Computer and System Sciences, Vol. 55, No. 1, 1997, pp. 119-139.

[19] http://en.wikipedia.org/wiki/Technical_analysis

[20] http://stockcharts.com/school/doku.php?id=chart_school:technical_indicators

APPENDIX A: TECHNICAL INDICATORS

Simple Moving Average (SMA): It is the simple mean of the previous *n* price data points of a security.

$$SMA(M,n) = \frac{1}{n}\sum_{i=0}^{n-1} P_{M-i} = SMA(M-1,n) + \left(\frac{P_M - P_{M-n}}{n}\right)$$

where P_i is i^{th} data point and M is the current data point.

Exponential Moving Average (EMA): It is a weighted mean of the previous *n* data points where the weighting for each older data point decreases exponentially, never reaching zero.

$$EMA(M,n) = \lambda \sum_{i=0}^{n-1} (1-\lambda)^i P_{M-i}$$
$$= \lambda P_M + (1-\lambda) EMA(M-1,n)$$

where $\lambda = 2/(n+1)$

Average Directional index (ADX): This indicator measures the strength of the upward or downward movement by comparing the current price with the previous price range and displays the result as a positive (upward) movement line (pDI), a negative (downward) movement line (nDI), and a trend strength line (ADX) between 0 and 100. The ADX is calculated in terms of upward (U) and downward (D) price movements, and the true range (TR). Average True Range (ATR) measures the volatility.

$$ADX = 100 \times EMA_n\left(\left|\frac{pDI - nDI}{pDI + nDI}\right|\right)$$

$$pDI = 100 \times \frac{EMA_n(pDM)}{ATR}$$

$$nDI = 100 \times \frac{EMA_n(nDM)}{ATR}$$

$$U = High_{n+1} - High_n$$

$$D = Low_n - Low_{n+1}$$

$$pDM = \begin{cases} U & U > D \,\&\, U > 0 \\ 0 & Otherwise \end{cases}$$

$$nDM = \begin{cases} D & D > U \,\&\, D > 0 \\ 0 & Otherwise \end{cases}$$

$$ATR = EMA_n(TR)$$

$$TR = MAX(High_{n+1}, Close_n) - MIN(Low_{n+1}, Close_n)$$

Moving Average Divergence Convergence (MACD): It turns two trend-following indicators, moving averages, into a momentum oscillator by subtracting the longer moving average from the shorter moving average. As a result, MACD offers the best of both worlds: trend following and momentum.

$$MACD(f,s) = EMA_f(Close) - EMA_s(Close)$$

$$Signal(S) = EMA_S(MACD)$$

$$Hist = MACD(f,s) - Signal(S)$$

Relative Strength Index (RSI): RSI is a momentum oscillator that measures the speed and change of price movements. It is a trading indicator which intended to indicate the current and historical strength or weakness of a market based on the closing prices of completed trading periods. It assumes that prices close higher in strong market periods, and lower in weaker periods and computes this as a ratio of the number of incrementally higher closes to the incrementally lower closes.

$$RSI = 100 - 100 \times \left(\frac{1}{1 + RS}\right)$$

$$RS = \frac{EMA_n U}{EMA_n D}$$

$$U = \begin{cases} Close_{n+1} - Close_n & Close_{n+1} > Close_n \\ 0 & Otherwise \end{cases}$$

$$D = \begin{cases} Close_n - Close_{n+1} & Close_{n+1} < Close_n \\ 0 & Otherwise \end{cases}$$

Stochastic Oscillator: Stochastic Oscillator (%K) is a momentum indicator that shows the location of the close relative to the high-low range over a set number of periods. Does not follow the price but the momentum of the price. As a rule, the momentum changes direction before price.

$$\%K = 100 \; * \; \left(\frac{C_M - L_{min}}{H_{max} - L_{min}}\right)$$

$$\%D = 3 - day \;\; SMA \; of \; \%K$$

Where C_M represents current closing price, L_{min} and H_{max} represents lowest and highest trading price over the horizon of *n* trading period.

StochRSI: StochRSI is an oscillator that measures the level of RSI relative to its high-low range over a set time period. StochRSI applies the stochastic formula to RSI values, instead of price values.

$$StochRSI = \frac{RSI_M - RSI_{min}}{RSI_{max} - RSI_{min}}$$

where RSI_M represents RSI value calculated using trading price over the horizon of *n* trading period.

Bollinger Bands: These are volatility bands placed above and below a moving average. Volatility is based on the standard deviation, which changes a volatility increase and decreases. An indicator derived from Bollinger Bands called %b tells us where we are within the bands. Bandwidth, another indicator derived from Bollinger Bands, may also interest traders. It is the width of the bands expressed as a percent of the moving average.

$$Middle\ Band = n\text{-}period\ SMA$$

$$Upper\ Band = middle\ ban + \ n\text{-}period\ std\ deviation\ of\ price * m$$

$$Lower\ Band = middle\ band - \ n\text{-}period\ std\ deviation\ of\ price * m$$

$$\%\, b = \frac{Close - Lower\ Band}{Upper\ Band - Lower\ Band}$$

$$Bandwidth = \frac{Upper\ Band - Lower\ Band}{Middle\ Band}$$

where typical values for *m* is 2 and *n* is 20

Money Flow Index (MFI): MFI is an oscillator calculated over *n* period, ranging from 0 to 100, showing money flow on up days as a percentage of the total of up and down days. MFI is a momentum indicator that is similar to the RSI in both interpretation and calculation. However, MFI is a more rigid indicator in that it is volume-weighted, and is therefore a good measure of the strength of money flowing in and out of a security.

$$MFI = 100 - 100 \times \left(\frac{1}{1 + MR}\right)$$

$$MR = \frac{positive\ money\ flow}{negative\ money\ flow}$$

$$positive\ money\ flow = \sum_{i=M-n+1}^{M} \begin{cases} volume * TP_i & TP_i > TP_{i-1} \\ 0 & Otherwise \end{cases}$$

$$negative\ money\ flow = \sum_{i=M-n+1}^{M} \begin{cases} volume * TP_i & TP_i < TP_{i-1} \\ 0 & Otherwise \end{cases}$$

where TP_i represents typical price for the i^{th} trading period.

Accumulation/Distribution Index (ADI): ADI assess the cumulative flow of money into and out of a security. The degree of buying or selling pressure can be determined by the location of the Close, relative to the High and Low for the corresponding period. There is buying pressure when a stock closes in the upper half of a period's range and there is selling pressure when a stock closes in the lower half of the period's trading range.

$$ADI_M = ADI_{M-1} + volume * \left(\frac{(C - L) - (H - C)}{H - L}\right)$$

where H, L and C represent the highest, the lowest and the closing price of current trading period respectively.

Chaikin Oscillator: The Chaikin Oscillator is simply the MACD indicator applied to the ADI indicator.