Horizontal Generalization Properties of Fuzzy Rule-Based Trading Models

Célia da Costa Pereira and Andrea G.B. Tettamanzi

Università degli Studi di Milano
Dipartimento di Tecnologie dell'Informazione
via Bramante 65, I-26013 Crema, Italy
pereira@dti.unimi.it, andrea.tettamanzi@unimi.it

Abstract. We investigate the generalization properties of a data-mining approach to single-position day trading which uses an evolutionary algorithm to construct fuzzy predictive models of financial instruments. The models, expressed as fuzzy rule bases, take a number of popular technical indicators on day t as inputs and produce a trading signal for day $t + 1$ based on a dataset of past observations of which actions would have been most profitable.

The approach has been applied to trading several financial instruments (large-cap stocks and indices), in order to study the *horizontal*, i.e., cross-market, generalization capabilities of the models.

Keywords: Data Mining, Modeling, Trading, Evolutionary Algorithms.

1 Introduction

Single-position automated day-trading problems (ADTPs) involve finding an automated trading rule for opening and closing a single position within a trading day. They are a neglected subclass of the more general automated intraday trading problems, which involve finding profitable automated technical trading rules that open and close positions within a trading day.

An important distinction that may be drawn is the one between static and dynamic trading problems. A *static* problem is when the entry and exit strategies are decided before or on market open and do not change thereafter. A *dynamic* problem allows making entry and exit decisions as market action unfolds.

Dynamic problems have been the object of much research, and different flavors of evolutionary algorithms (EAs) have been applied to the discovery and/or the optimization of dynamic trading rules (cf., e.g., [3].

Static problems are technically easier to approach, as the only information that has to be taken into account is information available before market open. This does not mean, however, that they are easier *to solve* than their dynamic counterparts.

This paper focuses on the generalization properties of the solutions to a class of static single-position automated day-trading problems found by means of a data-mining approach which uses an EA to construct a fuzzy predictive model of a

M. Giacobini et al. (Eds.): EvoWorkshops 2008, LNCS 4974, pp. 93–102, 2008.

financial instrument. The model takes the values of a number of popular technical indicators computed on day t as inputs to produce a *go short*, *do nothing*, *go long* trading signal for day $t+1$ based on a dataset of past observation of which actions would have been most profitable.

2 Evaluating Trading Rules

Informally, we may think of a trading rule R as some sort of decision rule which, given a time series $X = \{x_t\}_{t=1,2,\ldots,N}$ of prices of a given financial instrument, for each time period t returns some sort of trading signal or order.

Following the financial literature on investment evaluation [2], the criteria for evaluating the performance of trading rules, no matter for what type of trading problem, should be measures of risk-adjusted investment returns. The reason these are good metrics is that, in addition to the profits, consistency is rewarded, while volatile patterns are not.

While the Sharpe ratio [7] is probably the most popular measure of risk-adjusted returns for mutual funds and other types of investments, it has been criticized for treating positive excess returns, i.e., windfall profits, the same way as it treats negative returns; however, traders, just like investors, do not regard windfall profits as something to avoid as unexpected losses. A variation of the Sharpe ratio which acknowledges this fact is Sortino ratio [8], which may be defined as

$$\mathrm{SR}_d(R;X) = \frac{r(R;X) - r_f}{\mathrm{DSR}_{r_f}(R;X)}, \tag{1}$$

where $r(R;X)$ is the annualized average log-return of rule R applied to time series X, r_f is the risk-free rate r_f, assumed to be constant during the timespan covered by X, and $\mathrm{DSR}_\theta(R;X) = \sqrt{\frac{Y}{N}\sum_{t=1}^{N}\min\{0, \theta - r(R;X,t)\}^2}$ is called the downside risk [4,8] of rule R on X. Unlike the Sharpe ratio, the Sortino ratio adjusts the expected return for the risk of falling short of the risk-free return; positive deviations from the least acceptable return θ are not taken into account to calculate risk.

3 The Trading Problem

We focus on a particular class of static ADTP, whereby the trading strategy allows taking both long and short positions at market during the opening auction, a position is closed as soon as a pre-defined profit r_{TP} has been reached, or otherwise at market during the closing auction as a means of preventing losses beyond the daily volatility of an instrument.

Such problems make up the simplest class of problems when it comes to rule evaluation: all is required is open, high, low, and close quotes for each day, since a position is opened at market open, if the rule so commands, and closed either with a fixed profit or at market close.

A trading rule for this static problem has just to provide a ternary decision: *go short*, *do nothing*, or *go long*.

Given time series $X = \{x_t^O, x_t^H, x_t^L, x_t^C\}_{t=1,\ldots,N}$, of daily open, high, low, and close quotes, the log-return generated by rule R in the t^{th} day of time series X is

$$r(R; X, t) = \begin{cases} r_{TP} & \text{if signal is } \textit{go long} \text{ or } \textit{short} \text{ and } \bar{r} > r_{TP}, \\ s \ln \frac{x_t^C}{x_t^O} & \text{if signal is } \textit{go long} \text{ or } \textit{short} \text{ and } \bar{r} \leq r_{TP}, \\ 0 & \text{otherwise,} \end{cases} \tag{2}$$

where $\bar{r} = \ln \frac{x_t^H}{x_t^O}$ for a long position, and $\bar{r} = \ln \frac{x_t^O}{x_t^L}$ for a short one.

This problem is therefore among the most complex single-position day-trading problems whose solutions one can evaluate when disposing only of open, high, low, and close quotes for each day. The reason we chose to focus on such problem is indeed that while such kind of quotes are freely available on the Internet for a wide variety of securities and indices, more detailed data can in general only be obtained for a fee.

We approach this problem by evolving trading rules that incorporate fuzzy logic. The adoption of fuzzy logic is useful in two respects: first of all, by recognizing that concept definitions may not always be crisp, it allows the rules to have what is called an *interpolative behavior*, i.e., gradual transitions between decisions and their conditions; secondly, fuzzy logic provides for linguistic variables and values, which make rules more natural to understand for an expert.

4 The Approach

Data mining is a process aimed at discovering meaningful correlations, patterns, and trends between large amounts of data collected in a dataset. A model is determined by observing past behavior of a financial instrument and extracting the relevant variables and correlations between the data and the dependent variable. We describe below a data-mining approach based on the use of EAs, which recognize patterns within a dataset, by learning models represented by sets of fuzzy rules.

4.1 Fuzzy Models

A model is described through a set of fuzzy rules, made by one or more antecedent clauses (“IF ...”)and a consequent clause (“THEN ...”). Clauses are represented by a pair of indices referring respectively to a variable and to one of its fuzzy sub-domains, i.e., a membership function.

Using fuzzy rules makes it possible to get homogenous predictions for different clusters without imposing a traditional partition based on crisp thresholds, that often do not fit the data, particularly in financial applications. Fuzzy decision rules are useful in approximating non-linear functions because they have a good interpolative power and are intuitive and easily intelligible at the same time.

Their characteristics allow the model to give an effective representation of the reality and simultaneously avoid the “black-box” effect of, e.g., neural networks.

The intelligibility of the model is useful for a trader, because understanding the rules helps the user to judge if a model can be trusted.

4.2 The Evolutionary Algorithm

The described approach incorporates an EA for the design and optimization of fuzzy rule-based systems originally developed to learn fuzzy controllers [9,6], then adapted for data mining, [1].

A model is a rule base, whose rules comprise up to four antecedent and one consequent clause each. Input and output variables are partitioned into up to 16 distinct linguistic values each, described by as many membership functions. Membership functions for input variables are trapezoidal, while membership functions for the output variable are triangular.

Models are encoded in three main blocks:

1. a set of trapezoidal membership functions for each input variable; a trapezoid is represented by four fixed-point numbers, each fitting into a byte;
2. a set of symmetric triangular membership functions, represented as an area-center pair, for the output variable;
3. a set of rules, where a rule is represented as a list of up to four antecedent clauses (the IF part) and one consequent clause (the THEN part); a clause is represented by a pair of indices, referring respectively to a variable and to one of its membership functions.

An island-based distributed EA is used to evolve models. The sequential algorithm executed on every island is a standard generational replacement, elitist EA. Crossover and mutation are never applied to the best individual in the population.

The recombination operator is designed to preserve the syntactic legality of models. A new model is obtained by combining the pieces of two parent models. Each rule of the offspring model can be inherited from one of the parent models with probability 1/2. When inherited, a rule takes with it to the offspring model all the referred domains with their membership functions. Other domains can be inherited from the parents, even if they are not used in the rule set of the child model, to increase the size of the offspring so that their size is roughly the average of its parents’ sizes.

Like recombination, mutation produces only legal models, by applying small changes to the various syntactic parts of a fuzzy rulebase.

Migration is responsible for the diffusion of genetic material between populations residing on different islands. At each generation, with a small probability (the migration rate), a copy of the best individual of an island is sent to all connected islands and as many of the worst individuals as the number of connected islands are replaced with an equal number of immigrants.

A detailed description of the algorithm and of its genetic operators can be found in [6].

4.3 The Data

In principle, the modeling problem we want to solve requires finding a function which, for a given day t, takes the past history of time series X up to t and produces a trading signal *go short*, *do nothing*, or *go long*, for the next day.

Instead of considering all the available past data, we try to take advantage of *technical analysis*, an impressive body of expertise used everyday by practitioners in the financial markets, which is about summarizing important information of the past history of a financial time series into few relevant statistics. The idea is then to reduce the dimensionality of the search space by limiting the inputs of the models we look for to a collection of the most popular and time-honored technical analysis statistics and indicators.

For lack of space, we cannot give here mathematical definitions for the indicators used, and we refer the interested reader to specialized publications [5].

After a careful scrutiny of the most popular technical indicators, we concluded that more data were needed if we wanted an EA to discover meaningful models expressed in the form of fuzzy IF-THEN rules. Combinations of statistics and technical indicators are required that mimic the reasonings analysts and traders carry out when they are looking at a technical chart, comparing indicators with current price, checking for crossings of different graphs, and so on.

Combinations may take the form of differences between indicators that are pure numbers or that have a fixed range, or of ratios of indicators such as prices and moving average, that are expressed in the unit of measure of a currency. Following the use of economists, we consider the natural logarithm of such ratios, and we define the following notation: given two prices x and y, we define

$$x : y \equiv \ln \frac{x}{y}. \tag{3}$$

Eventually, we came up with the following combinations:

- all possible combinations of the Open (O), High (H), Low (L), Close (C), and previous-day Close (P) prices: $O : P$, $H : P$, $L : P$, $C : P$, $H : O$, $C : O$, $O : L$, $H : L$, $H : C$, $C : L$;
- close price compared to simple and exponential moving averages, $C : \mathrm{SMA}_n$, $C : \mathrm{EMA}_n$, $n \in \{5, 10, 20, 50, 100, 200\}$;
- the daily changes of the close price compared to simple and exponential moving averages, $\Delta(C : \mathrm{SMA}_n)$, $\Delta(C : \mathrm{EMA}_n)$, where $\Delta(x) \equiv x(t) - x(t-1)$;
- the MACD histogram, i.e., MACD − signal, and the daily change thereof, $\Delta(\text{Histogram})$;
- Fast stochastic oscillator minus slow stochastic oscillator, $\%K - \%D$, and the daily change thereof, $\Delta(\%K - \%D)$.

The full list of the statistics, technical indicators, and their combinations used as model inputs is given in Table 1.

4.4 Fitness

Modeling can be thought of as an optimization problem, where we wish to find the model M^* which maximizes some criterion which measures its accuracy in

Table 1. The independent variables of the dataset

Name	Formula	Explanation
Open	x_t^O	the opening price on day t
High	x_t^H	the highest price on day t
Low	x_t^L	the lowest price on day t
Close	x_t^C	the closing price on day t
Volume	x_t^V	the volume traded on day t
O:P	$x_t^O : x_{t-1}^C$	opening price on day t vs. previous-day closing price
H:P	$x_t^H : x_{t-1}^C$	high on day t vs. previous-day closing price
L:P	$x_t^L : x_{t-1}^C$	low on day t vs. previous-day closing price
C:P	$x_t^C : x_{t-1}^C$	close on day t vs. previous-day closing price
H:O	$x_t^H : x_t^O$	high on day t vs. same-day opening price
C:O	$x_t^C : x_t^O$	closing on day t vs. same-day opening price
O:L	$x_t^O : x_t^L$	opening price on day t vs. same-day lowest price
H:L	$x_t^H : x_t^L$	high on day t vs. same-day low
H:C	$x_t^H : x_t^C$	high on day t vs. same-day closing price
C:L	$x_t^C : x_t^L$	closing price on day t vs. same-day low
dVolume	$x_t^V : x_{t-1}^V$	change in volume traded on day t
C:MAn	$x_t^C : \mathrm{SMA}_n(t)$	n-day simple moving averages, for $n \in \{5, 10, 20, 50, 100, 200\}$.
dC:MAn	$\Delta(x_t^C : \mathrm{SMA}_n(t))$	daily change of the above
C:EMAn	$x_t^C : \mathrm{EMA}_n(t)$	n-day exponential moving averages, for $n \in \{5, 10, 20, 50, 100, 200\}$.
dC:EMAn	$\Delta(x_t^C : \mathrm{EMA}_n(t))$	daily change of the above
MACD	$\mathrm{MACD}(t)$	Moving average convergence/divergence on day t
Signal	$\mathrm{signal}(t)$	MACD signal line on day t
Histogram	$\mathrm{MACD}(t) - \mathrm{signal}(t)$	MACD histogram on day t
dHistogram	$\Delta(\mathrm{MACD}(t) - \mathrm{signal}(t))$	daily change of the above
ROC	$\mathrm{ROC}_{12}(t)$	rate of change on day t
K	$\%K_{14}(t)$	fast stochastic oscillator on day t
D	$\%D_{14}(t)$	slow stochastic oscillator on day t
K:D	$\%K_{14}(t) - \%D_{14}(t)$	fast vs. slow stochastic oscillator
dK:D	$\Delta(\%K_{14}(t) - \%D_{14}(t))$	daily change of the above
RSI	$\mathrm{RSI}_{14}(t)$	relative strength index on day t
MFI	$\mathrm{MFI}_{14}(t)$	money-flow index on day t
AccDist	$\Delta(\mathrm{AccDist}(t))$	The change of the accumulation/distribution index on day t
OBV	$\Delta(\mathrm{OBV}(t))$	The change of on-balance volume on day t
PrevClose	x_{t-1}^C	closing price on day $t-1$

predicting $y_i = x_{im}$ for all records $i = 1, \ldots, N$ in the training dataset. The most natural criteria for measuring model accuracy are the mean absolute error and the mean square error.

One big problem with using such criteria is that the dataset must be *balanced*, i.e., an equal number of representative for each possible value of the predictive attribute y_i must be present, otherwise the underrepresented classes will end up being modeled with lesser accuracy. In other words, the optimal model would be very good at predicting representatives of highly represented classes, and quite poor at predicting individuals from other classes.

To solve this problem, we divide the range $[y_{\min}, y_{\max}]$ of the predictive variable into 256 bins. The b^{th} bin, X_b, contains all the indices i such that

$$1 + \lfloor 255 \frac{y_i - y_{\min}}{y_{\max} - y_{\min}} \rfloor = b. \tag{4}$$

For each bin $b = 1, \dots, 256$, it computes the mean absolute error for that bin

$$\mathrm{err}_b(M) = \frac{1}{\|X_b\|} \sum_{i \in X_b} |y_i - M(x_{i1}, \dots, x_{i,m-1})|, \tag{5}$$

then the total absolute error (TAE) as an integral of the histogram of the absolute errors for all the bins, $\mathrm{tae}(M) = \sum_{b:\|X_b\| \neq 0} \mathrm{err}_b(M)$. Now, the mean absolute error for every bin in the above summation counts just the same no matter how many records in the dataset belong to that bin. In other words, the level of representation of each bin (which, roughly speaking, corresponds to a class) has been factored out by the calculation of $\mathrm{err}_b(M)$. What we want from a model is that it is accurate in predicting all classes, independently of their cardinality.

The fitness used by the EA is given by $f(M) = \frac{1}{\mathrm{tae}(M)+1}$, in such a way that a greater fitness corresponds to a more accurate model.

5 Experiments

A desirable property for models is their capability of generalizing, i.e., correctly predicting other data than those used to discover them. There are two dimensions of generalization that might be of interest here:

1. a *vertical* dimension, which has to do with being able to correctly model the behavior of the financial instrument used for learning for a timespan into the future;
2. a *horizontal* dimension, which has to do with being able to correctly model the behavior of other financial instruments than the one used for learning: here we might be interested in applying the model to similar instruments (i.e., same sector, same market, same asset class) or to instruments taxonomically further away.

We have tested our approach with the specific aim of assessing its *horizontal* generalization properties. The reason why this type of generalization is desirable is that it would allow the user to trade "young" financial instruments, for which too few data are available, by using models trained on similar, but "older" financial instruments.

5.1 Experimental Protocol

The following financial instruments have been used for the experiments:

- the Dow Jones Industrial Average index (DJI);
- the Nikkei 225 index (N225);
- the common stock of Italian oil company ENI, listed since June 18, 2001 on the Milan stock exchange;
- the common stock of world's leading logistics group Deutsche Post World Net (DPW), listed since November 20, 2000 on the XETRA stock exchange;
- the common stock of Intel Co. (INTC), listed on the NASDAQ.

For all the instruments considered, three datasets of different length have been generated, in an attempt to gain some clues on how much historical data is needed to obtain a reliable model:

- a "long-term" dataset, generated from the historical series of prices since January 1, 2002 till December 31, 2006, consisting of 1,064 records, of which 958 are used for training and the most recent 106 are used for testing;
- a "medium-term" dataset, generated from the historical series of prices since January 1, 2004 till December 31, 2006, consisting of 561 records, of which 505 are used for training and the most recent 56 are used for testing;
- a "short-term" dataset, generated from the historical series of prices since January 1, 2005 till December 31, 2006, consisting of 304 records, of which 274 are used for training and the most recent 30 are used for testing;

Table 2. Summary of experimental results. Minimum, average, and maximum values are over the best models produced in ten independent runs of the island-based EA, when applied to the corresponding validation set.

Performance Measure	**Dataset**								
	Long-Term			Medium-Term			Short-Term		
	min	avg	max	min	avg	max	min	avg	max
Dow Jones Industrial Average									
Fitness	0.3297	0.3394	0.3484	0.3188	0.3327	0.3457	0.3183	0.3398	0.3671
Return*	0.1618	0.2303	0.4017	0.1062	0.2280	0.5503	0.0996	0.3225	0.5416
Sortino Ratio	1.5380	2.5572	4.7616	0.7642	2.7949	6.5557	0.7215	4.0799	6.9968
Nikkei 225									
Fitness	0.3211	0.3414	0.3651	0.3241	0.3418	0.3575	0.3205	0.3351	0.3529
Return*	−0.1467	−0.0006	0.2119	−0.1118	0.0006	0.1436	−0.1063	−0.0161	0.1040
Sortino Ratio	−1.9181	0.0782	4.1485	−1.5070	0.0253	2.1311	−1.9135	−0.1033	3.2197
ENI Stock									
Fitness	0.2459	0.3268	0.3500	0.2475	0.2907	0.3425	0.2402	0.2949	0.3277
Return*	−0.1389	0.0122	0.2120	−0.0856	0.0248	0.1547	−0.1936	−0.0372	0.2643
Sortino Ratio	−2.3274	−0.2751	3.0867	−2.4578	−0.1799	2.4460	−2.8959	−0.9655	3.2188
Deutsche Post World Net Stock									
Fitness	0.3182	0.3306	0.3451	0.3200	0.3342	0.3506	0.3118	0.3299	0.3403
Return*	−0.0607	0.0476	0.2646	−0.0246	0.0547	0.2480	0.0117	0.1169	0.2820
Sortino Ratio	−15.8114	−2.3809	10.5500	−15.8114	−0.1780	12.7425	−10.2067	0.0920	4.6700
Intel Co. Stock									
Fitness	0.2490	0.3050	0.3443	0.2433	0.2838	0.3658	0.2394	0.2665	0.3333
Return*	0.0247	0.1015	0.1669	0.0131	0.2254	0.4292	−0.0244	0.1252	0.3632
Sortino Ratio	−0.2467	0.8624	3.2520	−0.4569	2.9042	6.1129	−15.8114	−0.7107	3.4903

*) Annualized logarithmic return.

The validation dataset, in all cases, consists of records corresponding to the first half of 2007, which require a historical series starting from March 17, 2006 (200 market days before January 2, 2007) to be generated, due to the 200-day moving averages and their changes that need to be computed.

5.2 Results

For each combination of instrument and dataset, ten runs of the EA with four islands of size 100 connected according to a ring topology and with a standard parameter setting have been performed. Each run lasted as many generations as required to reach convergence, defined as no improvement for 100 consecutive generations. The results are summarized in Table 2.

A superior performance of models evolved against the short-term dataset can be noticed. That is an indication that market conditions change over time and a profitable trading model one year ago may not be profitable today. Furthermore, while very profitable models for the DJIA are found on average, performance is much less consistent on the other four instruments, probably due to the specific volatility patterns of the instruments considered.

6 Horizontal Generalization

In order to study *horizontal* generalization, the best performing models (in terms of their Sortino ratio) for each instrument have been applied to the other four. The results of this experiment are reported in Table 3.

From those results, we can draw the following conclusions: it appears that the DJI model has interesting generalization capabilities and performs well on N225, DPW, and ENI, but, surprisingly enough, fails on one of its components, namely

Table 3. Results of applying to an instrument models trained on another instrument. The instruments used for training the models (with an indication of the relevant trainin set) are in the rows; the instruments to which the models have been applied are in the columns.

Performance Measure	**Validation Instrument**				
	DJI	N225	DPW	ENI	INTC
Dow Jones Industrial Average, Short-Term, Run #1					
Annualized Log-Return	0.5416	0.1195	0.2768	0.0630	−0.0428
Sortino Ratio	6.9968	2.5756	4.4929	0.4528	−1.2073
Nikkei 225, Long-Term, Run #6					
Annualized Log-Return	0.2620	0.2119	0.1183	0.2653	0.0627
Sortino Ratio	3.6495	4.1485	0.6594	3.8662	0.1410
Deutsche Post World Net Stock, Medium-Term, Run #2					
Annualized Log-Return	0	0.0150	0.0688	0.1034	0.0011
Sortino Ratio	−15.8114	20.9006	12.7425	1.9863	−15.5462
ENI Stock, Short-Term, Run #3					
Annualized Log-Return	0.0135	0.1815	0.3626	0.2643	0.1601
Sortino Ratio	−11.6329	3.7711	4.3744	3.2188	1.1145
Intel Co. Stock, Medium-Term, Run #10					
Annualized Log-Return	0	−0.0207	0.2492	0.1177	0.1968
Sortino Ratio	−15.8114	−1.4857	2.4810	0.9493	6.1129

INTC; the other model trained on an index, N225, extends satisfactorily to all other instruments; models trained on stocks, instead, show poor generalization capabilities when it comes to modeling the two indices, but, with one exception, extend quite well to the other instruments of the same type.

7 Conclusions

An experimental test of the generalization capabilities of a fuzzy-evolutionary data-mining approach to static ADTPs has been performed. The results demonstrate that the idea of using high-performance models discovered for a financial instrument for trading other somehow related financial instruments is feasible, although with a grain of salt. As a matter of fact, evidence has been gathered that models trained on indices tend to perform well, with rare exceptions, on other indices and stocks, whereas models trained on individual stocks tend to perform well on other stocks, but poorly on indices.

Future work will involve, besides examining a larger data set with more runs, evolving the models on a heterogenous set of securities to boost the robustness and generalization capabilities of the evolved rules.

References

1. Beretta, M., Tettamanzi, A.: Learning fuzzy classifiers with evolutionary algorithms. In: Pasi, G., Bonarini, A., Masulli, F. (eds.) Soft Computing Applications, pp. 1–10. Physica Verlag, Heidelberg (2003)
2. Bodie, Z., Kane, A., Marcus, A.J.: Investments, 7th edn. McGraw-Hill, New York (2006)
3. Brabazon, A., O'Neill, M.: Biologically Inspired Algorithms for Financial Modelling. Springer, Berlin (2006)
4. Harlow, H.V.: Asset allocation in a downside-risk framework. Financial Analysts Journal, 30–40 (September/October 1991)
5. Kirkpatrick, C.D., Dahlquist, J.R.: Technical Analysis: The Complete Resource for Financial Market Technicians. FT Press, Upper Saddle River (2006)
6. Tettamanzi, A., Poluzzi, R., Rizzotto, G.G.: An evolutionary algorithm for fuzzy controller synthesis and optimization based on SGS-Thomson's W.A.R.P. fuzzy processor. In: Zadeh, L.A., Sanchez, E., Shibata, T. (eds.) Genetic algorithms and fuzzy logic systems: Soft computing perspectives, World Scientific, Singapore (1996)
7. Sharpe, W.F.: The Sharpe ratio. Journal of Portfolio Management 21(1), 49–58 (1994)
8. Sortino, F.A., van der Meer, R.: Downside risk — capturing what's at stake in investment situations. Journal of Portfolio Management 17, 27–31 (1991)
9. Tettamanzi, A.: An evolutionary algorithm for fuzzy controller synthesis and optimization. In: IEEE International Conference on Systems, Man and Cybernetics. IEEE Systems, Man, and Cybernetics Society, vol. 5/5, pp. 4021–4026 (1995)