Genetic and Evolutionary Algorithms for Time Series Forecasting

Paulo Cortez, Miguel Rocha, and José Neves

Departamento de Informática Universidade do Minho
Braga - PORTUGAL
pcortez@di.uminho.pt, mrocha@di.uminho.pt, jneves@di.uminho.pt

Abstract. Nowadays, the ability to forecast the future, based only on past data, leads to strategic advantages, which may be the key to success in organizations. *Time Series Forecasting* allows the modeling of complex systems as black-boxes, being a focus of attention in several research arenas such as *Operational Research*, *Statistics* or *Computer Science*. On the other hand, *Genetic and Evolutionary Algorithms (GEAs)* are a novel technique increasingly used in *Optimization* and *Machine Learning* tasks. The present work reports on the forecast of several *Time Series*, by *GEA* based approaches, where *Feature Analysis*, based on statistical measures is used for dimensionality reduction. The handicap of the evolutionary approach is compared with conventional forecasting methods, being competitive.

Keywords: Genetic and Evolutionary Algorithms, Time Series Forecasting, Time Series Analysis, ARMA models.

1 Introduction

Time Series Forecasting (TSF), the forecast of a time ordered variable, turns on into a decisive tool in problem solving, since it allows one to model complex systems where the goal is to predict the system's behavior and not how the system works. Indeed, in the last few decades an increasing focus as been put over this field. Contributions from the arenas of *Operational Research*, *Statistics*, and *Computer Science* as lead to solid *TSF* methods (e.g., *Exponential Smoothing* or *Regression*) that replaced the old fashioned ones, which were primarily based on intuition.

An alternative approach for *TSF* arises from the *Artificial Intelligence (AI)* field, where one has observed a trend to look at *Nature* for inspiration, when building problem solving models. In particular, studies on the biological evolution influenced the loom of powerful artifacts, such as *Genetic and Evolutionary Algorithms (GEAs)*, that enriched the potential use of *AI* in a broad set of scientific and engineering problems, such as the ones of *Combinatorial* and *Numerical Optimization* [9].

GEAs are suited for combinatorial optimization problems, where the exhaustion of all possible solutions require enormous computational power, heuristically finding solutions where other methods seem to fail. The use of *GEAs* in *TSF* is expected to increase in importance, motivated by advantages such as explicit model representation and adaptive evolutionary search, which escapes from unsatisfactory local minima.

The present work aims at testing several *TSF* models, inspired on evolutionary strategies, over a broad range of real *TSs*. The paper is organized as follows: firstly, the basic concepts for *TS* analysis, and *GEAs* are defined; then, a description of the different models and experiments is given; finally, the results obtained are presented and compared with other conventional *TSF* methods.

2 Time Series Analysis

A *Time Series (TS)* is a collection of chronologically ordered observations x_t, each one being recorded at a specific time t (period). *TSs* can uprise in a wide set of domains such as *Finance*, *Production* or *Control*, just to name a few. A *TS* model ($\widehat{x}_t$), assumes that past patterns will recur in the near future. The *error* of a forecast is given by the difference between actual values and what was predicted:

$$e_t = x_t - \widehat{x}_t \tag{1}$$

The overall performance of a forecasting model is evaluated by an accuracy measure, namely the *Sum Squared Error (SSE)*, *Root Mean Squared (RMSE)*, and *Normalized Mean Square Error (NMSE)*, which are given in the form:

$$\begin{aligned} SSE &= \sum_{i=1}^{l} e_i^2 \\ RMSE &= \sqrt{\frac{SSE}{l}} \\ NMSE &= \frac{SSE}{\sum_{i=1}^{l} (x_t - \overline{x})^2} \end{aligned} \tag{2}$$

where l denotes the number of forecasts and $\overline{x}$ the mean of the *TS*.

A common statistical instrument for *TS* analysis is the *autocorrelation* coefficient, defined by:

$$r_k = \frac{\sum_{t=1}^{s-k} (x_t - \overline{x})(x_{t+k} - \overline{x})}{\sum_{t=1}^{s} (x_t - \overline{x})} \tag{3}$$

in terms of the k's lag, where s denotes the *TS*' size. Autocorrelations can be useful for decomposition of the *TS* main components (*trend* and *seasonal* effects) (Figure 2).

One quite successful *TSF* method is *Exponential Smoothing (ES)*, which is based on some underlying patterns (e.g., *trend* and *seasonal* ones) that are distinguished from random noise by averaging the historical values. Its popularity is due to advantages such as the simplicity of use, the reduced computational demand and the accuracy of the forecasts, specially with seasonal *TSs*. The general model, also known as *Holt-Winters*, is defined by the basic equations [10]:

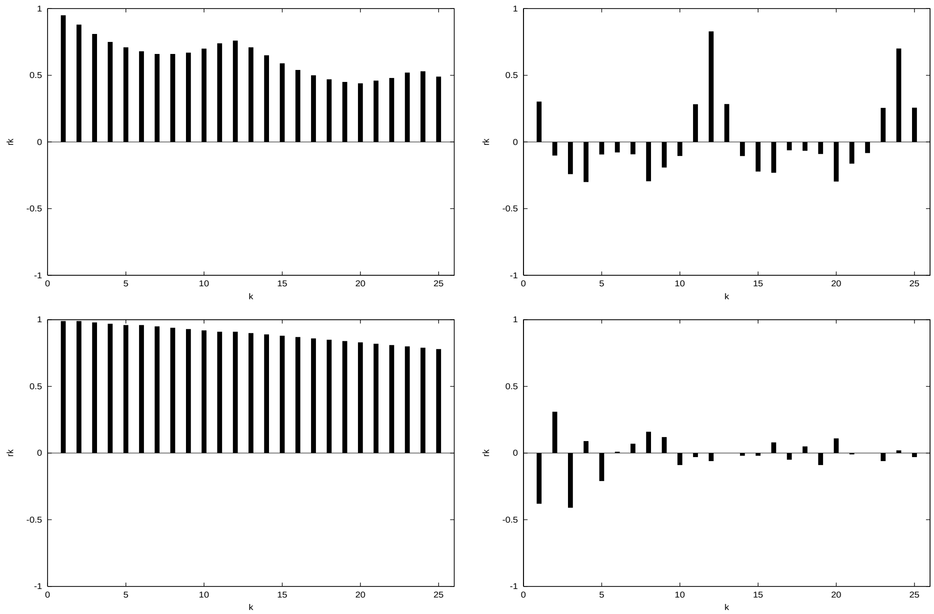

Fig. 1. Autocorrelation coefficients of typical *Seasonal and Trended*, *Seasonal*, *Trended* and *Non-Trended TS*

$$
\begin{array}{l}
F_t = \alpha \frac{x_t}{S_{t-K}} + (1-\alpha)(F_{t-1} + T_{t-1}) \\
T_t = \beta(F_t - F_{t-1}) + (1-\beta)T_{t-1} \\
S_t = \gamma \frac{x_t}{F_t} + (1-\gamma)S_{t-K} \\
\widehat{x}_t = (Fx_{t-1} + T_{t-1}) \times S_{t-K}
\end{array}
\qquad (4)
$$

where F_t, T_t and S_t stand for the smoothing, trend and seasonal estimates, K for the seasonal period, and α, β and γ for the model parameters.

The *AutoRegressive Integrated Moving-Average (ARIMA)* is another important *TSF* methodology, going over model identification, parameter estimation, and model validation [3]. The main advantage of this method relies on the accuracy over a wider domain of *TSs*, despite being more complex, in terms of usability and computational effort, than *ES*. The global model is based on a linear combination of past values (AR components) and errors (MA components) This model can be postulated as an $ARMA(P,Q)$ one, given in the form:

$$
\widehat{x}_t = \mu + \sum_{i=1}^{P} A_i x_{t-i} + \sum_{j=1}^{Q} M_j e_{t-j}
$$

where P and Q denote the AR and MA orders, A_i and M_j the AR and MA coefficients, being μ a constant value. Both the constant and the coefficients of the model are estimated using statistical approaches (e.g., least squares methods). Trended *TSs* require a differencing of the original values and seasonal *TSs* involve a transformation of the model. The methodology also contemplates the possibility of some kind of transformation in the original data (e.g., logarithmic).

Table 1. *Time Series* data

Series	Type	Domain	Description
passengers	*Seasonal*	Tourism	Monthly international airline passengers
paper	*& Trended*	Sales	Monthly sales of French paper
deaths	*Seasonal*	Traffic	Monthly deaths & injuries in UK roads
maxtemp		Meteorology	Maximum temperature in Melbourne
chemical	*Trended*	Chemical	Chemical concentration readings
prices		Economy	Daily *IBM* common stock closing prices
lynx	*Nonlinear*	Ecology	Annual number of lynx
kobe		Geology	Seismograph of the Kobe earthquake

For the experiments presented in this work, a set of eight *TSs* were selected (Table 1), taken from different origins, the majority of which are related with real problems, from different domains, ranging from the financial markets to natural processes [3][10][8] (Figure 2). All *TSs* were classified into four main categories that are expected to encompass all major *TS* types, namely: *Seasonal and Trended*, *Seasonal*, *Trended*, and *Nonlinear*.

3 Genetic and Evolutionary Algorithms

The term *Genetic and Evolutionary Algorithm (GEA)* is used to name a family of computational procedures where a number of potential solutions to a problem makes the way to an evolving population. Each individual codes a solution in a string (*chromosome*) of symbols (*genes*), being assigned a numerical value (*fitness*), that stands for a solution's quality measure. New solutions are created through the application of genetic operators (typically *crossover* or *mutation*). The whole process evolves via a process of stochastic selection biased to favor individuals with higher fitnesses.

The first *GEAs* [6], and most of the ones developed so far, make use of a binary representation; i.e., the solutions to a given problem are coded into a $\{0,1\}$ alphabet. However, some authors have argued that when one is faced with problems where the parameters are given by real values, the best strategy is to represent them directly into the chromosome, thus using a *Real-Valued Representation (RVR)*, which allows the definition of richer genetic operators [11]. In this work, two genetic operators were adopted. Its picture is given below:

Arithmetical Crossover - each gene in the offspring is a linear combination of the values in the ancestors' chromosomes in the same positions [11]. If a_i

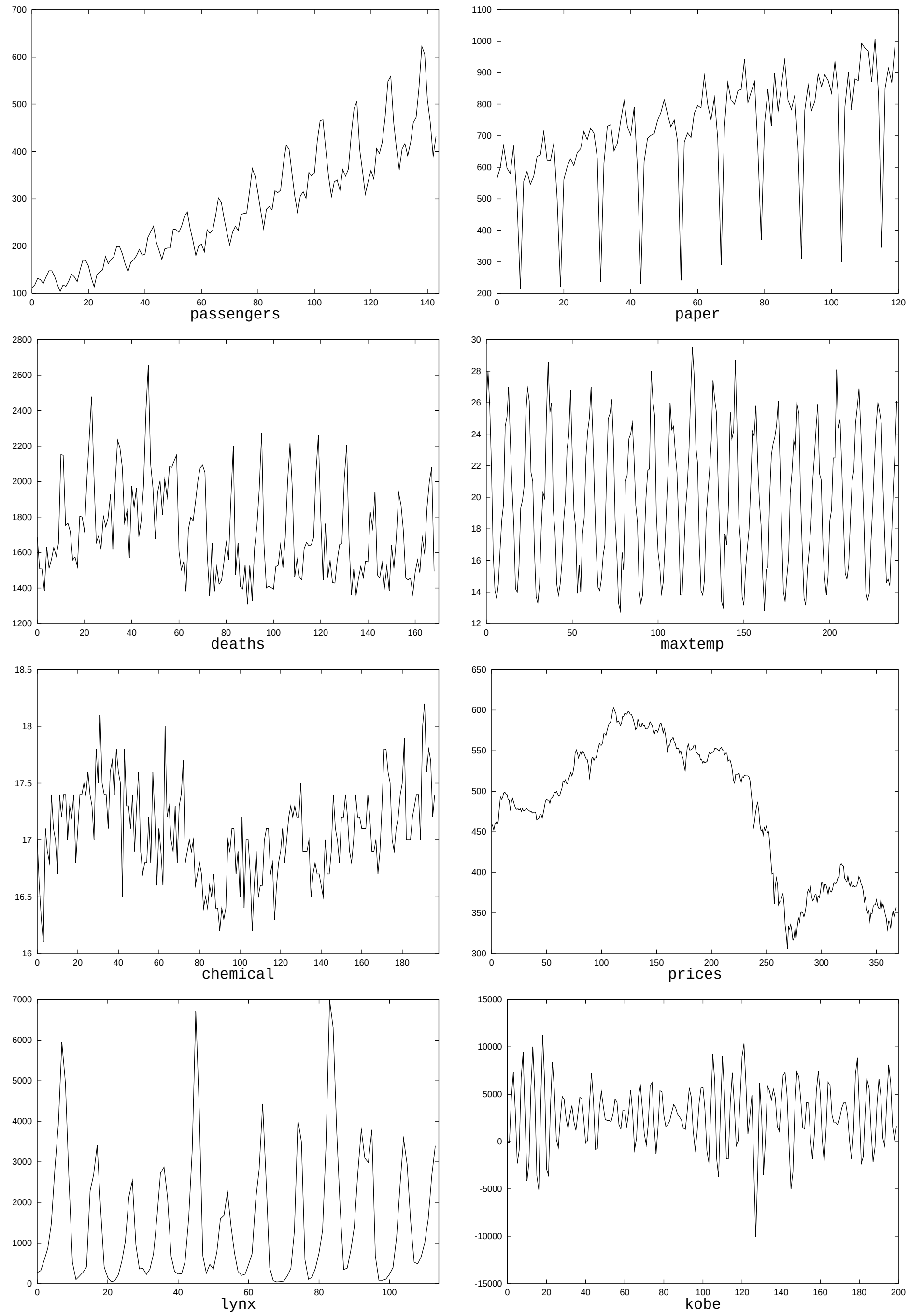

Fig. 2. The series of Table 1

and b_i are the offspring's genes, and z_i and w_i the ancestors' ones, at the position i, then $a_i = \lambda \cdot z_i + (1 - \lambda) \cdot w_i$ and $b_i = \lambda \cdot w_i + (1 - \lambda) \cdot z_i$, where λ is a random number in the range $[0; 1]$.

Gaussian Perturbation - this is a *mutation* operator that adds, to a given gene, a value taken from a Gaussian distribution, with zero mean.

4 Evolutionary Forecasting Models

In spite of its youth, in the *Evolutionary Computation* arena, a stream of new models and techniques for problem solving are coming into life, being particularly useful for numerical or combinatorial optimization processes. It is surprising to realize that the work in applying these techniques to forecasting is so scarce. In fact, although there are some publications in this area, these are not numerous nor noticeable. The existent work focuses mainly in some kind of parameter optimization, under a conventional model such as *Holt-Winters* [1] or *ARIMA* [7][4]. Recent developments such as *Genetic Programming (GP)* [2] and *GEAs* with *RVRs* [11], are expected to improve the performances of these approaches.

In this work, two approaches to forecasting, both based on *GEAs* with *RVRs*, were followed. In the former one, the forecasting model is a linear combination of previous values. Under this scenario, the genes in the chromosome code for the weights by which previous values are multiplied. With the latter, both previous values and errors are taken into account, following a strategy inspired on the *ARMA* models, where the genes code for the coefficients.

Both models make use of a *Sliding Time Window (STW)* that defines the set of time lags used to build a forecast, also defining the number of the model inputs. A *STW* will be denoted by the sequence $STW = < k_1, k_2, ..., k_n >$, for a model with n inputs and k_i time lags. The choice of a given *STW* is crucial to set the performance of a given model. A large sliding window can increase the system entropy, diminishing the learning capacity of the model, while small windows may contain insufficient information. The selection of the relevant time lags can improve forecasting (e.g., *ARIMA* models often use the $1, 12$ and 13 lags for monthly seasonal trended series).

An empirical approach to the problem is to use information based on the *TS* analysis. Four heuristic strategies will be used for *STW* selection, based on the autocorrelation values, and stated as follows:

A - a full *STW* with all time lags from 1 to a given maximum m: $STW = < 1, 2, ..., m >$ (m was set to 13, a value that was considered sufficient to encompass monthly seasonal and trended effects);

B - a *STW* with all lags with autocorrelation values above a given threshold (set to 0.2);

C - a *STW* with the four lags with highest autocorrelations (in the case of the seasonal trended series, these were taken after differencing, since trend effects may prevail over seasonal ones); and

D - the use of decomposable information; i.e.,
  - $STW = < 1, K, K + 1 >$ if the series is seasonal (period K) and trended;

– $STW =< 1, K >$ if the series is seasonal; and
– $STW =< 1 >$ and $STW =< 1, 2 >$ if the series is trended.

The two models considered in this work are given, in terms of a predefined *STW*, by:

G1 - linear combination based *GEA*; i.e.,

$$\widehat{x}_t = g_0 + \sum_{i \in \{1,\ldots,n\}} g_i x_{t-k_i}$$

where g_i stands for the i-th gene of the individuals' chromosome, and n for the *STW* size.

G2 - ARMA based *GEA*; i.e.,

$$\widehat{x}_t = g_0 + \sum_{i \in \{1,\ldots,n\}} (g_i x_{t-k_i} + g_{i+n} e_{t-k_i})$$

A model is said to overfit when it correctly handles the training data but fails to generalize. The usual statistical approach to overfitting is *model selection*, where different candidate models are evaluated according to a generalization estimate. Several complex estimators have been developed (e.g., Bootstrapping), which are computationally burdensome [12]. A reasonable alternative is the use of simple statistics that adds a penalty that is a function of model complexity, such as the *Bayesian Information Criterion (BIC)* [13]:

$$BIC = N \cdot ln(\frac{SSE}{N}) + p \cdot ln(N) \quad (5)$$

where N denotes the number of training examples and p the number of parameters (in this case $p_{G1} = 1 + n$ and $p_{G2} = 1 + 2n$).

5 Experiments and Results

The given models (**G1** and **G2**) were tested on the set of *TSs* from Table 1, using all the sliding window heuristics, when applicable. Thirty independent runs were performed in every case to insure statistical significance, being the results presented in terms of the mean and 95% confidence intervals. The *TSs* are divided into a training set, containing the first 90% values and a test set, with the last 10%. Only the training set is used for model selection and parameter optimization. The test set is used to compare the proposed approach with other methods.

In terms of the *GEA's* setup, the initial populations' genes were randomly assigned values within the range $[-1, 1]$. The population size was set to 100. The fitness of each chromosome was measured by the forecasting error (*RMSE*) over all the training patterns. The selection procedure is done by converting the fitness value into its ranking in the population and then applying a roulette wheel scheme. In each generation, 40% of the individuals are kept from the previous

Table 2. Results of the *GEA's* approach to the **prices** series

Model	**Sliding Window**	**Training RMSE**	**BIC**	**Forecasting RMSE**
	A=B	12.13	1673	10.72±0.69
G1	C	9.18	1443	8.24±0.32
	$D_{1,2}$	8.35	1372	7.49±0.05
	D_1	7.68	**1312**	**7.48**±0.00
	A=B	8.70	1536	8.78±0.33
G2	C	7.73	1357	7.68±0.10
	$D_{1,2}$	**7.63**	1325	7.65±0.02
	D_1	7.68	1318	7.49±0.00

generation, and 60% are generated by the application of the genetic operators described in Section 3. The *crossover* operator is responsible for breeding $\frac{2}{3}$ of the offspring and the *mutation* one is accountable for the remaining ones. Finally, the *GEA* is stopped after 2000 epochs.

For each *TS* both models ($G1$ and $G2$) were applied, considering all possible *STWs*. Therefore, each *TS* has several forecasting candidates. In order to select the best one, the *BIC* criterium was used. As an example, the used methodology will be explained in detail for the **prices** series (Table 2). The results of the last three columns are given in terms of the mean of the thirty runs. The 95% confidence intervals are also shown for the short term forecasting errors [5]. The best training *RMSE* is achieved for the window $< 1, 2 >$ and model *G2*. The *BIC* criterium works better, selecting the model that provides the best forecast. This behavior occurred consistently in all the *TSs*.

Table 3. The selected *GEAs* forecasting models (with lower *BIC*).

Series	**Model**	**Sliding Window**	**Forecasting RMSE**
passengers	G1	D=$< 1, 12, 13 >$	20.9±0.7
paper	G1	D=$< 1, 12, 13 >$	56.3±0.9
deaths	G1	D=$< 1, 12, 13 >$	134±1
maxtemp	G1	C=$< 1, 11, 12, 13 >$	0.915±0.008
chemical	G1	B=$< 1, 2, 3, 7 >$	0.343±0.003
prices	G1	D=$< 1 >$	7.48±0.00
lynx	G2	C=$< 1, 9, 10, 11 >$	262±6
kobe	G2	A=$< 1, 2, ..., 13 >$	524±16

Table 3 shows the best *GEA* models, when adopting the *BIC* criterium for model selection. This criterium, which penalizes complexity, selects the $G1$ models for the linear series and the $G2$ for the nonlinear ones, which is a logical outcome.

A comparison throughout evolutionary and conventional models is given in Table 4. The error values over the test set are given in terms of two measures, namely the *RMSE* and the *NMSE* ones (in brackets). This last measure is included since it makes easier the comparison among the different series and methods considered. The *ES* parameters (α, β and γ) were optimized using a 0.01 grid search for the best *RMSE*, while the *ARIMA* models were achieved using a forecasting package (*FORECAST PRO*).

Table 4. Comparison between different *TSF* approaches

Series	ES	ARIMA	GEA
passengers	**16.7** (0.71%)	17.8 (0.81%)	20.9 (1.12%)
paper	**41.0** (3.1%)	61.0 (6.8%)	56.3 (5.8%)
deaths	145 (43%)	144 (42%)	**134** (37%)
maxtemp	0.917 (4.1%)	1.068 (5.6%)	**0.915** (4.1%)
chemical	0.354 (51%)	0.361 (53%)	**0.343** (48%)
prices	7.50 (0.39%)	7.72 (0.41%)	**7.48** (0.38%)
lynx	876 (57%)	504 (19%)	**262** (5%)
kobe	3199 (105%)	582 (4%)	**524** (3%)

The results of the evolutionary approach are very interesting, with the best forecasting handicaps in 6 of the 8 *TSs*. In terms of the different types of *TSs* considered the proposed method seems to have its weakness in the seasonal and trended series, where the *ES* prevails. In all other kinds of series the results are very good, specially in the non-linear *TSs*.

6 Conclusions

The results of the application of *GEAs* to the *TSF* field are, at least, encouraging. In fact, the methodology proposed presents better results than the traditional methods used in the majority of the *TSs* considered. Furthermore, the *BIC* criterium showed a good performance in model selection, making the approach easier to automate. In fact, the proposed system does not require complicated statistical pre-processing, being easy to use by a beginner in the field.

In the future, it is intended to pursue on the automation of the model selection stage, namely on the process of selecting the best *STW*. An alternative is to enlarge the number of different *STWs* attempted and to find, within this search space, the best alternative. Since this is an optimization task, the use of a *GEA* could be advantageous, thus creating a two-level architecture. Another area of interest may rely in the enrichment of the forecasting models, by considering the integration of nonlinear functions (e.g., logarithmic or trigonometric).

Acknowledgements

The work of Paulo Cortez was supported by the portuguese *Foundation of Science & Technology* through the PRAXIS XXI/BD/13793/97 grant. The work of José Neves was supported by the PRAXIS' project PRAXIS/P/EEI/13096/98.

References

1. Adriana Agapie and Alexandru Agapie. Forecasting the Economic Cycles Based on an Extension of the Holt-Winters Model. A Genetic Algorithms Approach. In *Proceedings of the IEEE Computational Intelligence for Financial Forecasting Engineering*, pages 96–99, 1997.
2. W. Banzhaf, P. Nordin, R. Keller, and F. Francone. *Genetic Programming, An Introduction.* Morgan Kaufmann Publishers, Inc, USA, 1998.
3. G. Box and G. Jenkins. *Time Series Analysis: Forecasting and Control.* Holden Day, San Francisco, USA, 1976.
4. C. Chai, C. Chuek, M. DP, and T. Huat. Time Series Modelling and Forecasting using Genetic Algorithms. In *Proceedings of the First International Conference on Knowledge-Based Intelligent Electronic Systems*, volume 1, pages 260–268, Adelaide, Australia, 1995.
5. A. Flexer. Statistical evaluation of neural networks experiments: Minimum requirements and current practice. In *Proceedings of the 13th European Meeting on Cybernetics and Systems Research*, volume 2, pages 1005–1008, Vienna, Austria, 1996.
6. John Holland. *Adaptation in Natural and Artificial Systems.* PhD thesis, University of Michigan, Ann Arbor, 1975.
7. C. Huang and H. Yang. A Time Series Approach To Short Term Load Forecasting Through Evolutionary Programming Structures. In *Proceedings of the EMPD'95 Internacional Conference*, pages 583–588, 1995.
8. R. Hyndman. *Time Series Data Library.* (http://www-personal.buseco.monash.edu.au/~hyndman/TSDL/), 2000.
9. G. Luger and W. Stubblefield. *Artificial Intelligence, Structures and Strategies for Complex Problem Solving.* Addison Wesley Longman, Inc., USA, 1998.
10. S. Makridakis and S. Wheelwright. *Forecasting Methods for Management.* John Wiley & Sons, New York, fifth edition, 1989.
11. Z. Michalewicz. *Genetic Algorithms + Data Structures = Evolution Programs.* Springer-Verlag, USA, third edition, 1996.
12. W. Sarle. Stopped Training and Other Remedies for Overfitting. In *Proceedings of the 27th Symposium on the Interface of Computer Science and Statistics*, pages 352–360, 1995.
13. G. Schwarz. Estimating the Dimension of a Model. *The Annals of Statistics*, 6:461–4, 1978.