Available online at www.sciencedirect.com
SCIENCE DIRECT®
APPLIED MATHEMATICS AND COMPUTATION
ELSEVIER Applied Mathematics and Computation 164 (2005) 885–912
www.elsevier.com/locate/amc

Model identification of ARIMA family using genetic algorithms

Chorng-Shyong Ong [a], Jih-Jeng Huang [a], Gwo-Hshiung Tzeng [b,c,*]

[a] *Department of Information Management, National Taiwan University, No. 1, Sec. 4, Roosevelt Road, Taipei 106, Taiwan*
[b] *Institute of Management of Technology and Institute of Traffic and Transportation College of Management, National Chiao Tung University, 1001 Ta-Hsueh Road, Hsinchu 300, Taiwan*
[c] *Kai Nan University, No. 1, Kai-Nan Road, Luchu, Taoyuan 338, Taiwan*

Abstract

ARIMA is a popular method to analyze stationary univariate time series data. There are usually three main stages to build an ARIMA model, including model identification, model estimation and model checking, of which model identification is the most crucial stage in building ARIMA models. However there is no method suitable for both ARIMA and SARIMA that can overcome the problem of local optima. In this paper, we provide a genetic algorithms (GA) based model identification to overcome the problem of local optima, which is suitable for any ARIMA model. Three examples of times series data sets are used for testing the effectiveness of GA, together with a real case of DRAM price forecasting to illustrate an application in the semiconductor industry. The results show that the GA-based model identification method can present better solutions, and is suitable for any ARIMA models.

Keywords: ARIMA; Stationary; SARIMA; Genetic algorithms; Model identification

* Corresponding author.
E-mail address: ghtzeng@cc.nctu.edu.tw (G.-H. Tzeng).

0096-3003/$ - see front matter © 2004 Elsevier Inc. All rights reserved.
doi:10.1016/j.amc.2004.06.044

1. Introduction

ARIMA is the method first introduced by Box–Jenkins [1] to analyze stationary univariate time series data, and has since been used in various fields. The generalized form of ARIMA can be described as

$$\phi(B)\Phi(B^s)(1-B)^d(1-B)^D Y_t = \theta(B)\Theta(B^s)Z_t, \tag{1}$$

where B denotes the backward shift operator; d and D denote the non-seasonal and seasonal order of differences taken, respectively; $\phi(B)$, $\theta(B)$, $\Phi(B)$ and $\Theta(B)$ are polynomials in B and B^s of finite order p and q, P and Q, respectively, and usually abbreviated as SARIMA $(p,d,q)(P,D,Q)_s$. When there is no seasonal effect, a SARIMA model reduces to pure ARIMA(p,d,q), and when the time series data set is stationary a pure ARIMA reduces to ARMA(p,q).

The original assumptions and limitations of ARIMA include weak stationarity, equally spaced observation intervals, and a length of about 50–100 observations [1,2]; in addition, it provides better formulation for incremental than for structural change [2]. As we know, there are three main stages in building an ARIMA model: (1) model identification, (2) model estimation and (3) model checking. Although many previous papers have concentrated on model estimation [3–10], model identification is actually the most crucial stage in building ARIMA models [11], because false model identification will cause the wrong stage of model estimation and increase the cost of re-identification. The stages of building an ARIMA model are described in Fig. 1.

The first method uses the sample partial autocorrelation function (PACF) and the sample autocorrelation function (ACF), as proposed by Box and Jenkins [1] to identify the models in AR and MA, respectively. However, when the time series data sets have mixed ARMA effect, the plot cannot provide clear

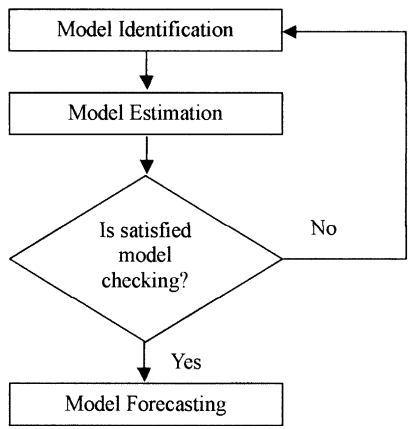

Fig. 1. The stages of building ARIMA models.

lags to identify. In addition, the lags of a mixed ARMA model usually involve subjective judgment, which make the results unstable [11]. Therefore, this paper proposes a method using genetic algorithms (GA) that can effectively find the global optimum solution, are suitable for ARIMA family models, and increase the accuracy of forecasting in business applications.

In order to provide a more objective and consistent method to identify the appropriate order of ARIMA, numerous methods for criterion selection have been proposed [12–21]. Some of these methods, called pattern identification, provide quick and easy methods to pick appropriate lags using a table which is constructed by the integral orders, p and q, of AR and MA, respectively. However, there are some problems such as the lack of pattern identification method for the seasonal ARIMA model and local optimization. These problems may result because the pattern identification cannot be used for seasonal time series models [22], and these methods do not present subset solutions, only searching for local optimum solutions.

The concept of subset regression was described by McClave [23] to propose an algorithm for best subset identification. However that algorithm needs to calculate all possible subsets based on *FPE* criterion, and it may inefficient in large order or multivariate cases. In order to overcome this problem, Krolzig and Hendry [24] proposed the *PcGet* algorithm to test insignificant variables based on the t and F test. Chen and Tsay [25] used *ACE* and *BTUTO* algorithms to identify the best subset regression. Chao and Phillips [26] proposed *PIC* to reduce rank structure and Winker [27] provided a threshold accepting method to select multivariate lag structure automatically. Although many studies have discussed methods to overcome the problem of subset regression, the difference of this paper can be described as follows: First, the studies above generally focused on the ARX or VAR model (also called dynamic regression) rather than on ARIMA model. Second, in this paper, we focus on order selection of the lag rather than variable selection of the lag. Third, we do not know whether these methods can be applied in a seasonal ARIMA model because there are four order parameters (p, q, P, Q) that need to be estimated where ARX or VAR only need two order parameters (VARX(p, s)) to be estimated. In this paper, GA is adopted to provide another method for model selection and is applied in ARIMA family models.

GA was pioneered by John Holland [28] and extended in later works [29–32]. The advantage of GA is its stochastic global search method that mimics "the survival of the fittest" in natural evolution. Although many studies have presented applications of GA for time series [23–35], these applications generally have focused on the problem of parameter estimation. However, there is no doubt that model identification is the most crucial stage in building an ARIMA model, and GA is used for this purpose in this article. The order of ARIMA will be treated as a chromosome, using a genetic operator to select global optimum orders.

In this study, three time series data sets, including ARMA, ARIMA, and SARIMA models, are illustrated to show the effectiveness of GA in the model identification stage. The forecasting of DRAM pricing trends are implemented for business decision making, and the results show that GA is more appropriate than the traditional methods. In additional, the model identification method using GA is suitable for a SARIMA model, whereas the traditional methods are not.

This paper is organized as follows: The statement of the problem caused by traditional model identification methods is described in Section 2. Section 3 describes the procedures of GA used to identify the ARIMA model. Three examples of time series data sets illustrate the effective of GA in Section 4. In Section 5 a real case for forecasting the DRAM pricing trends demonstrates an application in the semiconductor industry. Conclusions are presented in Section 6.

2. Statement of the problem

The first steps in building an ARIMA model are determining the appropriate order for the model identification stage, then estimating the unknown parameters, and checking the residuals from the fitted model. Although many papers [3,6–9,36] concentrate on model estimation, the main problem is assessing the order of the process, rather than estimating the coefficients [37]. The correlogram method, the sample PACF and the sample ACF are used as proposed by Box and Jenkins in appropriate differenced series for identifying the orders p and q of the ARMA (p, q) model. However this is complicated and not easily conducted, particularly for the mixed model, in which neither p nor q vanishes.

Various kinds of information criteria, such as the Akaike Information Criterion (AIC) [12], the corrected Akaike Information Criterion (AICC) [13], the Final Prediction Error criterion (FPE) [14], the Hannan–Quinn Criterion (HQC) [15], and the Schwarz Bayesian Criterion (SBC) [16] have been proposed for model identification to overcome these difficulties. Additionally, in order to effectively and easily identify the order of ARIMA, some pattern identification methods have been proposed, including the R and S array method [17], the *Corner* method [18], the *ESACF* method [19], the *SCAN* method [20], and the *MINIC* method [21].

Although the pattern identification methods seem to provide a better method for determining the appropriate order of ARIMA, there are some problems which need to be considered. First, the pattern identification methods cannot be used for seasonal time series models [22] because the SARIMA needs 4 dimensions. The second problem is that these methods provide only local optimum solutions. The pattern identification methods are used in the following way for determining upper bounds, say $p_{\max}$ and $q_{\max}$, which are set for the

orders of $\phi(B)$ and $\theta(B)$. Then with $\bar{p} = \{0, 1, \ldots, p_{\max}\}$ and $\bar{q} = \{0, 1, \ldots, q_{\max}\}$, a table is formed to select the order that has optimum solution. However the pattern identification methods do not consider the subset solution. The benefits of a global optimum solution are as follows. If the data set fits the model, say AR(5), by the pattern identification method, then the model can be as

$$Z_t = \mu + (1 - \phi_1 B - \phi_2 B^2 - \phi_3 B^3 - \phi_4 B^4 - \phi_5 B^5) Z_t. \tag{2}$$

However if the global optimum solution falls in the subset, said AR((1, 5)), then the equation should be as

$$Z_t = \mu + (1 - \phi_1 B - \phi_5 B^5) Z_t. \tag{3}$$

That is, if we can reduce to three parameters for estimation, then the model will be more easy, robust and accurate. Thus the purpose of this paper is to propose a method that can effectively find the global optimum solution and is suitable for ARIMA family models.

The main problem in finding all the solutions lies in the computational cost and time required. Theoretically, if we want to identify a SARIMA model, the total sample space is $2^{(p_{\max}+q_{\max}+P_{\max}+Q_{\max}+4)}$ which is impractical when a high order model exists. The main advantage of GA is that it simultaneously searches a population of points and effectively finds the approximate optimum solution in complex data set. These powerful characteristics of GA are used in this paper for model identification.

3. Model identification by genetic algorithms

This section first describes the criteria for model identification using the pattern identification method. The characteristics and procedures of GA are presented in next subsection. Then the string representation, the initial population and fitness computation are proposed; and the settings for the genetic operator and the elitist strategy, stopping criterion in this study are stated. The last part of this section presents the method of stationarity test.

3.1. Criterion of model identification

Because the pattern identification methods are quick (compare with an exhaustive search) and easily select (compare with a traditional method such as ACF and PACF) appropriate orders, this concept is used in this paper. The *ESACF* and *SCAN* methods are represented by the symbols "X" and "O" to indicate the inappropriate and appropriate orders, respectively, and the *MINIC* method, which has the property of determination, is more convenient to select the orders. The *MINIC* method can tentatively identify the orders of an ARMA (p, q) process, as proposed in [38–40].

The procedure of the *MINIC* is described as follows. Assume a stationary and invertible time series $\{z_t : 1 \leqslant t \leqslant n\}$ with mean corrected form $\tilde{z}_l = z_l - \mu_z$, with a true autoregressive order of p, and with a true moving-average order of q. Then the *MINIC* method to compute information criteria for various autoregressive and moving average orders and the error series can be approximated by a high-order AR process

$$\hat{\varepsilon}_t = \hat{\phi}_{(p_\varepsilon,q)}(B)\tilde{z}_l \approx \varepsilon_l, \tag{4}$$

where $\hat{\varepsilon}_t$ denotes the error series, $\hat{\phi}_{(p_\varepsilon,q)}$ denotes the coefficient of AR, and the parameter estimates, $\hat{\phi}_{(p_\varepsilon,q)}$ are obtained from the Yule–Walker estimates. The choice of the autoregressive order, p_ε, is determined by the order that minimizes the information criterion, such as *AIC* or *SBC*. Since *SBC* had been proved to be strongly consistent, it determines the true model asymptotically [41], and preferred to *AIC* for comparing different models such as neural network [42]; thus the *SBC* method is adopted in this paper. Once the error series have been estimated for autoregressive test order $m = p_{\min}, \ldots, p_{\max}$ and for moving-average test order $j = p_{\min}, \ldots, p_{\max}$, then the ordinal least square (OLS) method estimates, $\hat{\phi}_{(m,j)}$ and $\hat{\theta}_{(m,j)}$, are computed from the regression model

$$\tilde{z}_l \sum_{i=1}^{m} \phi_i^{(m,j)} \tilde{z}_{l-i} \sum_{k=1}^{j} \theta_k^{(m,j)} \hat{\varepsilon}_{l-k} + \text{error}. \tag{5}$$

From the preceding parameter estimates, the *SBC* is then computed by

$$BIC(m, j) = \ln(\tilde{\sigma}^2_{(m,j)}) + 2(m + j)\ln(n)/n, \tag{6}$$

where

$$\tilde{\sigma}^2_{m,j} = \frac{1}{n} \sum_{l=t_0}^{n} \left(\tilde{z}_l - \sum_{i=1}^{m} \phi_i^{(m,j)} \tilde{z}_{l-i} + \sum_{k=1}^{j} \theta_k^{(m,j)} \hat{\varepsilon}_{l-k} \right)^2, \tag{7}$$

where $t_0 = p_\varepsilon + \max(m,j)$.

The *MINIC* method can tentatively identify the order of a stationary and invertible ARMA process, as described in [43,44]. Through the *MINIC* method can quickly and easily provide a method to identify the order in ARIMA, it is not appropriate for SARIMA, and there is the problem of local optima. In this paper, GA is used to overcome these problems.

3.2. Concepts of the GA approach

GA was pioneered in 1975 by Holland, and its concept is to mimic the natural evolution of a population by allowing solutions to reproduce, creating new solutions, which then compete for survival in the next iteration. The fitness improves over generations and the best solution is finally achieved. The initial

population, $P(0)$, is encoded randomly by strings. In each generation, t, the more fit elements are selected for the mating pool; and then processed by three basic genetic operators, reproduction, crossover, and mutation, to generate new offspring. On the basis of the principle of survival of the fittest, the best chromosome of a candidate solution is obtained. The pseudo code of GA illustrates the procedure of the computation as follows:

```
procedure
GA
begin
   t = 0
   initialize   P(t)
   evaluate   P(t)
   while   not   satisfy   stopping   rule   do
   begin
      t = t + 1
      select   P(t)   from   P(t − 1)
      alter   P(t)
      evaluate   P(t)
   end
end
```

The power of GA lies in its simultaneous searching a population of points in parallel, not a single point. Therefore GA can find the approximate optimum quickly without falling into a local optimum. In addition GA does not have the limitation of differentiability, as do other mathematical techniques. These characteristics of GA are the reasons it is used here for the problem of model identification in ARIMA models.

3.3. Procedures of GA

3.3.1. String representation

In order to represent the order in an ARMA model, there are four parts in each chromosome to represent the order of AR, MA, seasonal AR and seasonal MA. Each chromosome is made up of binary value strings. The ith genotype of each part denotes the status of the ith order entry. For example, if the chromosome is represented by (10011; 00110; 11000; 01110), the model can be SARMA $(p,q)(P,Q)_s$ as SARMA $((1,4,5),(3,4))((1,2),(2,3,4))_s$.

3.3.2. Population initialization

The initial population $P(0)$ is selected at random. Each genotype in the population can be initialized to present the degree of variance from the uniform

distribution. Note that there is no standard to determine the size, $P(0)$, of the initial population. Bhandari et al. [45] showed that as the number of iterations extends to infinity, the elitist model of GA will provide the optimal string for any population size, $P(0)$.

3.3.3. Fitness computation

The purpose of this study is to determine the order in an ARMA model. For this, the most crucial issue is determining the fit index. In this study, we adopt the *SBC* index as the fitness of a chromosome. Note that, although this study uses the *SBC* index to identify the order, other criteria can be used in the same procedures.

3.3.4. Genetic operators

3.3.4.1. Selection. The selection operator selects chromosomes from the mating pool using the "survival of the fittest" concept, as in natural genetic systems. Thus, the best chromosomes receive more copies, while the worst die off. The probability of variable selection is proportional to its fitness value in the population, according to the formula given by

$$P(x_i) = \frac{f(x_i)}{\sum_{j=1}^{N} f(x_j)}, \tag{8}$$

where $f(x_i)$ represents the fitness value of the ith chromosome, and N is the population size.

3.3.4.2. Crossover. The goal of crossover is to exchange information between two parent chromosomes in order to produce two new offspring for the next population. In this study, we use two-point crossover with a crossover probability, P_c. The proceeding in two-point crossover occurs when two parent chromosomes are swapped after two randomly selected points between $[1, N-1]$, creating two children. This instance can be described as follows: If the parent chromosomes are selected by

$$\begin{array}{lccc|cccc|ccc} \alpha_1 = & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ \alpha_2 = & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \end{array}$$

then two children will be produced as

$$\begin{array}{lcccccccccc} \beta_1 = & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ \beta_2 = & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \end{array}$$

3.3.4.3. Mutation. Mutation is a random process where one genotype is replaced by another to generate a new chromosome. Each genotype has the probability of mutation, P_m, changing from 0 to 1 or vice versa.

3.3.5. Elitist strategy and stopping criterion

Elitist strategy. The elitist strategy simply carries the fittest chromosome from the previous generation into the next. The advantage of the elitist strategy lies in insuring selection of the best chromosome and decreasing the time of convergence.

Termination criterion. In GA, two termination criteria are usually used: One is to set up a maximum generation, and the other is used when the chromosome cannot increase the fitness. In this study, we use the first criterion.

3.4. Unit root tests and variance stationarity

Since ARIMA models are only suitable for stationary time series, the data set will be appropriate differentiated when a time series has a unit root. The problems that caused by the unit root in ARIMA and SARIMA models are discussed in [46,47]. In this paper the *ADF* unit root tests [46,47], which are a popular technique in financial engineering fields, are used to test the stationarity and seasonal stationarity in time series data sets.

In addition, one of the assumptions in ARIMA models is weak stationarity, which requires not only mean stationarity, but also variance or homogeneous stationarity as well. The log transformation is a popular method [41] to convert time series that are nonstationary with variance into stationary time series, and this method is adopted in this article. The next section describes the implementation of three time series data sets for testing the effective of GA in the model identification stage.

4. Implementation for testing three examples

This section illustrates the results using GA to identify the order in ARIMA models. There are three time series examples used for testing the effectiveness of the GA-based model identification method. The results using GA are compared with the *SCAN*, the *ESACF*, and the *MINIC* methods.

4.1. Data set

GNP data set. The data set provided in [48] is the US real GNP from the first quarter of 1947 to the first quarter of 1991, a total of 176 observations.

Unemployment data set. The data set used in [48] is composed of seasonally adjusted quarterly US unemployment rates from 1948 to 1993.

Sales data set. This data set is the monthly sales for a souvenir shop, as used in [49].

4.2. The application of model identification by GA

Three time series data sets are implemented in this subsection. All the data sets process the *ADF unit* root test, model identification, and the test for white noise in the GA-based model, and they are compared with the other pattern identification methods.

4.2.1. GNP data set

The results of *ADF* unit root tests are described in Table 1, which shows that the data set has no unit root effect in the GNP data set and it needs no differentiation in the GNP data set. We can process model identification directly in next stage.

The order results of model identification using the *SCAN*, the *ESACF*, and the *MINIC* methods are described in Tables 2–4, respectively. The models are set as AR(1) or MA(2) in the *SCAN* method, and the model is ARMA(1,2) by the *ESACF* method. Based on Table 4, the *MINIC* method fulfills the minimum information criterion in AR(4).

The GA-based model identification is optimum in the fourth generation, and the best model is the subset model fitted as ARMA(1,(2,5)). The results of comparison between the *SCAN*, the *ESACF*, the *MINIC*, and the GA-based model identification methods are illustrated in Table 5. Based on Table 5, although using GA-based model identification is not highest in SBC, the other three criteria show the best results.

In order to determine whether the residuals are satisfied with white noise in the GA-based model, chi-square statistics are used for testing the goodness of fit. Based on Table 6, the residuals are white noise in all lags, indicating that the fitted model is suitable for GNP data set.

Table 1
The *ADF* unit root test in GNP data set

Type	Lags	Rho	*P* value[a]	Tau	*P* value[a]	*F* value	*P* value[a]
Zero mean	1	−42.2457	<0.0001***	−4.57	<0.000l***	–	–
	2	−46.1027	<0.0001***	−4.49	<0.0001***	–	–
	3	−43.2915	<0.0001***	−4.12	<0.0001***	–	–
Single mean	1	−82.0840	0.0013***	−6.28	<0.0001***	19.74	0.001***
	2	−118.7380	0.0001***	−6.49	<0.0001***	21.06	0.001***
	3	−170.7200	0.0001***	−6.35	<0.0001***	20.19	0.001***
Trend	1	−84.6395	0.0005***	−6.39	<0.0001***	20.47	0.001***
	2	−124.2630	0.0001***	−6.61	<0.0001***	21.85	0.001***
	3	−185.6680	0.0001***	−6.50	<0.0001***	21.19	0.001***

[a] The significant level is 0.05 and **denotes $p < 0.05$, ***denotes $p < 0.01$.

Table 2
The *SCAN* method in GNP data set

Lags	MA 0	MA 1	MA 2	MA 3	MA 4	MA 5
AR 0	X	X	O	O	O	O
AR 1	O	X	O	O	O	O
AR 2	O	O	O	O	O	O
AR 3	O	O	O	O	O	O
AR 4	O	O	O	O	O	O
AR 5	O	O	O	O	O	O

The significant level is 0.05 and X denotes $p < 0.05$, O denotes $p > 0.05$.

Table 3
The *ESACF* method in GNP data set

Lags	MA 0	MA 1	MA 2	MA 3	MA 4	MA 5
AR 0	X	X	O	O	O	O
AR 1	X	X	O	O	O	O
AR 2	X	X	O	O	O	O
AR 3	X	O	X	O	O	O
AR 4	X	X	X	O	O	O
AR 5	X	X	O	O	O	O

The significant level is 0.05 and X denotes $p < 0.05$, O denotes $p > 0.05$.

Table 4
The *MINIC* method in GNP data set

Lags	MA 0	MA 1	MA 2	MA 3	MA 4	MA 5
AR 0	−9.10158	−9.16985	−9.24115	−9.23599	−9.29382	−9.28837
AR 1	−9.23622	−9.22119	−9.22290	−9.21252	−9.28045	−9.26637
AR 2	−9.22864	−9.19936	−9.21010	−9.20027	−9.25453	−9.23701
AR 3	−9.23789	−9.21620	−9.19958	−9.17130	−9.22616	−9.22490
AR 4	**−9.30267**	−9.27538	−9.25587	−9.22723	−9.20040	−9.19739
AR 5	−9.28241	−9.25371	−9.23989	−9.21777	−9.20183	−9.17247

Table 5
The comparison of each method in GNP data set

Criterion	SCAN		ESACF	MINIC	GA-based
	$p = 1$; $q = 0$	$p = 0$; $q = 2$	$p = 1$; $q = 2$	$p = 4$; $q = 0$	$p = 1$; $q = (2,5)$
AIC	−1121.10	−1124.48	−1124.04	−1123.62	−1125.67
SBC	−1114.75	−1114.97	−1111.35	−1107.77	−1112.99
SSE	0.017249	0.016729	0.0165823	0.016433	0.016429
Variance	0.000099	0.000097	0.0000960	0.000096	0000096
Standard error	0.009957	0.009834	0.0098190	0.009803	0.009773

Table 6
Autocorrelation checking of residuals in GNP data set

Lags	Chi-square	DF	*P* value[a]
6	1.55	3	0.6706
12	7.70	9	0.5647
18	11.50	15	0.7165
24	13.25	21	0.8996
30	19.28	27	0.8596

[a] The significant level is 0.05 and **denotes $p < 0.05$, ***denotes $p < 0.01$.

4.2.2. Unemployment data set

The *ADF* unit root tests show that the unemployment data set has the unit root effect (Table 7) and appropriate differentiation is needed. In Table 8, after

Table 7
The *ADF* unit root tests in unemployment data set

Type	Lags	Rho	*P* value[a]	Tau	*P* value[a]	*F* value	*P* value[a]
Zero mean	1	−0.3943	0.5924	−0.31	0.5731	–	–
	2	−0.2115	0.6338	−0.20	0.6124	–	–
	3	−0.0871	0.6621	−0.10	0.6497	–	–
Single mean	1	−29.8348	0.0013***	−3.89	0.0027***	7.61	0.001***
	2	−22.3179	0.0054***	−3.24	0.0196**	5.31	0.0294**
	3	−18.2629	0.0155**	−2.89	0.049**	4.24	0.0735
Trend	1	−46.2274	0.0005***	−4.76	0.0008***	11.34	0.001***
	2	−37.2884	0.001***	−4.03	0.0094***	8.14	0.0069***
	3	−31.8174	0.004***	−3.59	0.0341**	6.44	0.0472**

[a] The significant level is 0.05 and **denotes $p < 0.05$, ***denotes $p < 0.01$.

Table 8
The *ADF* unit root test after first-order differencing

Type	Lags	Rho	*P* value[a]	Tau	*P* value[a]	*F* value	*P* value[a]
Zero mean	1	−97.2297	<0.0001***	−6.91	<0.0001***	–	–
	2	−144.6490	0.0001***	−7.03	<0.0001***	–	–
	3	−361.0200	0.0001***	−7.70	<0.0001***	–	–
Single mean	1	−97.4482	0.0013***	−6.90	<0.0001***	23.80	0.001***
	2	−145.1890	0.0001***	−7.02	<0.0001***	24.63	0.001***
	3	−362.4800	0.0001***	−7.68	<0.0001***	29.50	0.001***
Trend	1	−97.5151	0.0005***	−6.88	<0.000l***	23.66	0.001***
	2	−145.5030	0.0001***	−7.00	<0.000l***	24.50	0.001***
	3	−360.2250	0.0001***	−7.64	<0.000l***	29.32	0.001***

[a] The significant level is 0.05 and **denotes $p < 0.05$, ***denotes $p < 0.01$.

first-order differentiation the *ADF* unit root tests show that the unemployment data set is stationary, so we can process model identification next.

The results of the *SCAN*, the *ESACF*, and the *MINIC* methods are described in Tables 9–11, and the model settings are as ARIMA(2,1,1), ARIMA(0,1,2), and ARIMA(1,1,4), respectively.

The GA is optimum occurs in the fourth generation and the fitted model is ARIMA ((1,4,5),1,(4)). The results of model-selection criterion is compared with the other methods in Table 12, indicating that all the criteria show the GA-based model has highest value.

Table 9
The *SCAN* method in unemployment data set

Lags	MA 0	MA 1	MA 2	MA 3	MA 4	MA 5
AR 0	X	X	O	X	X	O
AR 1	X	X	X	X	X	O
AR 2	O	O	O	O	O	O
AR 3	X	O	O	O	O	O
AR 4	O	O	O	O	O	O
AR 5	O	O	O	O	O	O

The significant level is 0.05 and X denotes $p < 0.05$, O denotes $p > 0.05$.

Table 10
The *ESACF* method in unemployment data set

Lags	MA 0	MA 1	MA 2	MA 3	MA 4	MA 5
AR 0	X	X	O	X	X	O
AR 1	X	X	O	X	X	O
AR 2	X	O	X	O	X	O
AR 3	X	X	X	O	O	O
AR 4	O	X	X	O	O	O
AR 5	X	O	O	X	O	O

The significant level is 0.05 and X denotes $p < 0.05$, O denotes $p > 0.05$.

Table 11
The MINIC method in unemployment data set

Lags	MA 0	MA 1	MA 2	MA 3	MA 4	MA 5
AR 0	−5.22816	−5.46309	−5.58178	−5.58326	−5.61499	−5.6502
AR 1	−5.64028	−5.6187	−5.62651	−5.59700	**−5.68619**	−5.6796
AR 2	−5.68406	−5.66269	−5.63609	−5.62604	−5.66124	−5.65128
AR 3	−5.67048	−5.64235	−5.61342	−5.61167	−5.63186	−5.62658
AR 4	−5.68196	−5.65349	−5.62722	−5.59774	−5.60249	−5.59983
AR 5	−5.65547	−5.62574	−5.59831	−5.56877	−5.63260	−5.61592

Table 12
The comparison of each method in unemployment data set

Criterion	SCAN $p = 2; q = 0$	ESACF $p = 0; q = 2$	MINIC $p = 1; q = 4$	GA-based $p = (1,4,5); q = (4)$
AIC	−481.0600	−463.641	−481.68	−493.173
SBC	−468.4470	−454.181	−462.76	−477.406
SSE	0.649852	0.670807	0.583782	0.552612
Variance	0.003548	0.003946	0.003496	0.003289
Standard error	0.059564	0.062817	0.059124	0.057353

Table 13
Autocorrelation checking of residuals in unemployment data set

Lags	DF	Chi-square	P value[a]
6	2	3.650	0.1610
12	8	8.090	0.4243
18	14	20.280	0.1215
24	20	22.950	0.2912
30	26	30.020	0.2667

[a] The significant level is 0.05 and **denotes $p < 0.05$, ***denotes $p < 0.01$.

The results of the white noise test are described in Table 13. Based on Table 13 it can be seen that there is no autocorrelation between residuals in all lags, indicating that the GA-based model is suitable for forecasting the unemployment data set.

4.2.3. Sales data set

The *ADF* unit root tests show that the sales data set has a unit root in Table 14. The ADF unit root tests results of first-order differentiation are described in Table 15, indicating that there is seasonal effect in the 12th lag and the sales data set follows a SARIMA model. The sales data set process 12th order differentiation and model identification is next.

In Box–Jenkin's seasonal ARIMA model, there is no pattern identification method that can conduct a seasonal ARIMA model which needs 4-dimensions to be presented. Here, we use the correlogram method (ACF and PACF graphs) to judge the suitable ARIMA model. After the 1st and 12th differences, the graph of ACF and PACF is plotted as Fig. 2.

Based on the ACF and the PACF, it is reasonable to set the model in ARIMA$(2,1,0)(0,1,1)_{12}$. On the other hand, the GA-based model identification isoptimum in the 7th generation and the model is set as SARIMA$(0,1,1)(1,1,(2,3))_{12}$. After computing the fitness criteria, the comparison of the correlogram method and GA-based model are described as follow.

Table 14
The *ADF* unit root test in sales data set

Type	Lags	Rho	*P* value[a]	Tau	*P* value[a]	*F* value	*P* value[a]
Zero mean	1	0.3106	0.7549	0.69	0.8619	–	–
	2	0.3251	0.7586	1.15	0.9346	–	–
	3	0.3049	0.7534	1.20	0.9399	–	–
Single mean	1	−18.7486	0.0114**	−2.78	0.0654	4.29	0.0747
	2	−7.6605	0.2232	−1.65	0.4518	2.18	0.5214
	3	−5.9663	0.3384	−1.33	0.6102	1.73	1.6351
Trend	1	−85.9587	0.0003***	−6.10	<0.0001***	18.65	0.001***
	2	−55.8660	0.0002***	−4.17	0.0076***	8.73	0.001***
	3	−83.9668	0.0002***	−4.02	0.0116**	8.19	0.0111**

[a] The significant level is 0.05 and **denotes $p < 0.05$, ***denotes $p < 0.01$.

Table 15
ADF unit test after first-order differencing

Type	Lags	Rho	*P* value[a]	Tau	*P* value[a]	*F* value	*P* value[a]
Zero mean	1	−268.8630	0.0001***	−11.23	<0.0001***	–	–
	2	−489.1180	0.0001***	−7.46	<0.0001***	–	–
	12	**−11.6032**	**0.0154****	**−1.91**	**0.0536**	–	–
Single mean	1	−277.0160	0.0001***	−11.34	<0.0001***	64.28	0.001***
	2	−600.9190	0.0001***	−7.60	<0.0001***	28.87	0.001***
	12	**45.3336**	**0.9999**	**−2.96**	**0.0435*****	**4.47**	**0.0641**
Trend	1	−276.6280	0.0001***	−11.27	<0.0001***	63.55	0.001***
	2	−593.8650	0.0001***	−7.55	<0.0001***	28.62	0.001***
	12	**43.2993**	**0.9999**	**−3.02**	**0.1335**	**4.91**	**0.2088**

[a] The significant level is 0.05, and **denotes $p < 0.05$, ***denotes $p < 0.01$.

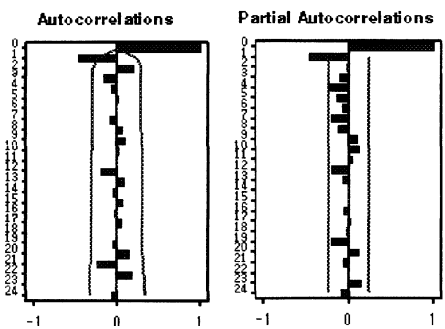

Fig. 2. The graph of ACF and PACF in sales data set.

Table 16
The comparison of the correlogram and GA-based method

Criterion	The correlogram method $p = 2$, $P = 0$; $q = 0$, $Q = 1$	GA-based model identification $p = 0$, $P = 1$; $q = 1$, $Q = (2,3)$
AIC	−24.409300	−29.054600
SBC	−15.358600	−17.741200
SSE	2.633600	2.398247
Variance	0.039307	0.036337
Standard error	0.198259	0.190623

Table 17
Autocorrelation checking of residuals in sales data set

Lags	DF	Chi-square	P value[a]
6	2	0.260	0.8787
12	8	8.820	0.3576
18	14	9.250	0.8146
24	20	15.570	0.7426

[a] The significant level is 0.05 and **denotes $p < 0.05$, ***denotes $p < 0.01$.

Based on Table 16, the GA-based model identification can obtain better results than the correlogram method. Additionally, the test for white noise is described in Table 17, which shows that there is no autocorrelation between residuals in all lags, and the GA-based model identification is suitable for the sales data set.

In this section, GA is used to identify the model by three time series. In the first-three data sets, the results show that the GA-based model identification has the highest value for model-selection criteria. In additional, the GA-based model identification is suitable for SARIMA models, for which the *SCAN*, the *ESACF*, and the *MINIC* methods are not suitable. The diagnoses of white noise are all satisfied in three data sets, indicating the excellent goodness of fit using the GA-based model identification method. Next, we apply this method to real-life problems for forecasting DRAM pricing trends.

5. Implementation: a real case for forecasting DRAM pricing trends

Dynamic Random Access Memory (DRAM) is a volatile memory that uses capacitors to hold electrical charges or store information, forming the simplest working memory cell. Recently, DRAMs have been widely used in computing applications, communication systems, graphics peripherals and electronic devices. The price of DRAM is critical in the final product cost and related inventory planning.

The data set is the 256 Mb double data rate (DDR) spot price collected from 2001/10/01 to 2003/03/12, a total of 276 records. DDR is one kind of DRAMs, which transfers data on both the rising and falling edge of the clock so that DDR is twice as fast as synchronous dynamic random access memory (SDRAM), and has now become the mainstream product in the DRAM market.

5.1. Problem descriptive

The pricing trend of DRAM is crucial in the semiconductor industry. In order to decrease the risks for DRAM firms in planning various kinds of inventory or building plants, the correct forecasting of DRAM price is critical. Next, the GA-based ARIMA is used for predicting the DDR spot price, and the implementation is described as follows.

5.2. DRAM spot price forecasting

The pricing trend of 256 Mb DDR, as shown in Fig. 3, is fluctuating and dynamic. Next, the *ADF* unit root test is used for testing the stationarity of DDR data set to determine the appropriate difference.

Based on Table 18, the *ADF* test shows that the DDR data set has the unit root effect, and appropriate differentiation is needed. In Table 19, after first-order differentiation, the *ADF* test shows that the DDR data set is stationary, so model identification can be processed in the next stage.

The final optimum model is ARIMA$(1,1,(7))$ by using GA; and the comparison of the *SCAN*, the *ESACF*, and the *MINIC* methods are described as Table 20. The GA-based ARIMA have the best fitness in all criteria, indicating the advantage over traditional model identification methods.

The pricing trend of DDR plot is described in Fig. 4. The real spot and forecasting price is very close in Fig. 3, indicating that the GA-based model identification method picked the appropriate order.

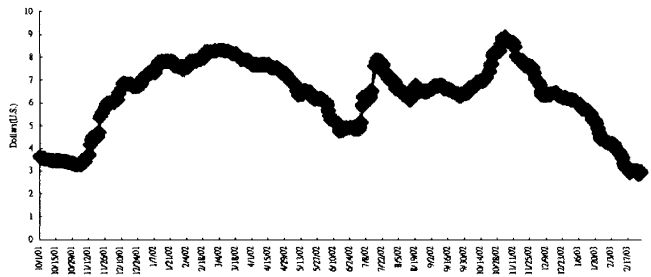

Fig. 3. The pricing trend of 256 Mb DDR.

Table 18
The *ADF* unit root test in DDR data set

Type	Lags	Rho	*P* value[a]	Tau	*P* value[a]	*F* value	*P* value[a]
Zero mean	1	−0.2531	0.6250	−0.36	0.5547	–	–
	2	−0.2576	0.6240	−0.36	0.5539	–	–
	3	−0.2774	0.6195	−0.37	0.5489	–	–
Single mean	1	−4.1519	0.5211	−1.42	0.5735	1.01	0.8132
	2	−4.3375	0.5010	−1.45	0.5599	1.05	0.8032
	3	−4.7895	0.4542	−1.51	0.5264	1.15	0.7782
Trend	1	−4.4415	0.8582	−1.59	0.7960	3.60	0.4554
	2	−4.5737	0.8493	−1.62	0.7845	3.63	0.4478
	3	−4.9320	0.8243	−1.68	0.7604	3.57	0.4613

[a] The significant level is 0.05 and **denotes $p < 0.05$, ***denotes $p < 0.01$.

Table 19
The *ADF* unit root test after first differencing

Type	Lags	Rho	*P* value[a]	Tau	*P* value[a]	*F* value	*P* value[a]
Zero mean	1	−167.265	0.0001***	−9.12	<0.0001***	–	–
	2	−152.133	0.0001***	−7.91	<0.0001***	–	–
	3	−125.890	0.0001***	−6.80	<0.0001***	–	–
Single mean	1	−167.266	0.0001***	−9.11	<0.0001***	41.49	0.001***
	2	−152.136	0.0001***	−7.90	<0.0001***	31.17	0.001***
	3	−125.886	0.0001***	−6.79	<0.0001***	23.09	0.001***
Trend	1	−178.618	0.0001***	−9.40	<0.0001***	44.21	0.001***
	2	−167.642	0.0001***	−8.20	<0.0001***	33.61	0.001**
	3	−143.097	0.0001***	−7.10	<0.0001***	25.20	0.001***

[a] The significant level is 0.05 and **denotes $p < 0.05$, ***denotes $p < 0.01$.

Table 20
The comparison of each criterion in DDR data set

Criterion	SCAN $p = 1; q = 0$	ESACF $p = 1; q = 2$	MINIC $p = 1; q = 0$	GA-based $p = 1; q = (7)$
AIC	−481.060	−463.641	−481.68	−493.173
SBC	−468.447	−454.181	−462.76	−477.406
SSE	4.344314	4.297699	4.344314	4.171514
Variance	0.003548	0.003946	0.003496	0.003289
Standard error	0.059564	0.062817	0.059124	0.057353

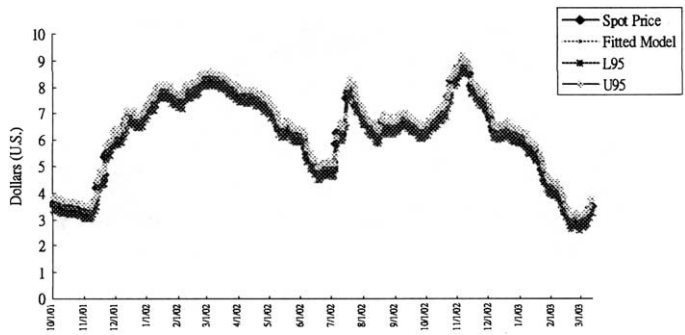

Fig. 4. The fitted model of spot price for 256 Mb DDR.

Table 21
Autocorrelation checking of residuals in DDR data set

Lags	DF	Chi-square	*P* value[a]
6	2.31	4	0.6797
12	5.10	10	0.8844
18	8.18	16	0.9432
24	15.50	22	0.8397
30	22.37	28	0.7637
36	31.17	34	0.6069
42	31.75	40	0.8210
48	32.98	46	0.9252

[a] The significant level is 0.05 and **denotes $p < 0.05$, ***denotes $p < 0.01$.

The test of white noise is described in Table 21. The results show there is no autocorrelation between residuals in all lags, so GA-based model identification is suitable for the DDR data set.

5.3. Results and discussion

DRAM has played a significant role in the development of the semiconductor industry. Because the DRAM industry is highly capital-intensive and the price of DRAM is volatile, in order to decrease the risk to DRAM firms, the correct forecasting of DRAM price is critical. Although the price of DRAM is of crucial importance to the electronics industry, it shows unexpected and high fluctuations. This uncertainty of DRAM prices makes it difficult for producers to make decisions such as the timing for establishing a DRAM plant, inventory policies, etc.

In this study, GA is used for model identification and it is compared with traditional pattern identification methods. The results of three examples and this real case show that the GA-based model identification method can provide

Table 22
The comparison of model identification methods

Model identification	Correlogram identification	Information criterion	Pattern identification	GA-based identification
Method	ACF and PACF	AIC, AICC, FPE, HQC, and SBC	SCAN, ESACF and MIMIC	Recursive model identification by GA
Criterion	Judgement by experience	Minimum information criterion	Statistic testing or minimum information criterion	Minimum information criterion
Disadvantage	Subjective and local optimum	Do not provide any method to identify the order	Local optimum	Computer efficient and cost
Advantage	Easily and quickly	Global optimum	Easily and quickly	Global optimum automatically

more accurate criteria, such as *AIC*, *BIC* and *SSE*; and provide more robust criteria, such as variance and standard error estimation. In addition, this case study also shows that all the estimate parameters are significant and the significant lags can actually be found by the GA-based method in the model estimate stage. This characteristic can help us to easily find the correct lags in a complex problem. The checking results of the model show that the error is random and no information is lost in building the ARIMA model. All the evidence shows that the GA-based model identification method can provide more accurate forecasting results than traditional pattern identification methods in ARIMA type models.

The traditional methods used to identify the order include the correlogram method, the information criterion, and the pattern identification method. However, none of these can provide a sound and objective method to determine the appropriate order in all ARIMA models. In this study, the GA-based model identification method is proposed to overcome the problem of local optima, and is suitable for all ARIMA-type models. The comparison of the traditional and the GA-based model identification methods is described in Table 22.

6. Conclusions

ARIMA is one of the most popular techniques for forecasting the trend of a time series data set. There are three main stages in building an ARIMA model, of which model identification is the most crucial stage. In order to identify the appropriate order, the correlogram, the information criterion and other pat-

tern identification method were previously developed. However they each have some shortcomings, including the problem of local optimum and together there is no method suitable for either ARIMA or SARIMA models both.

In this paper, we present a data-driven method using GA-based model identification in three example data sets, including ARMA, ARIMA and SARIMA models. A real case of DDR data set is used for forecasting pricing trends. All the results which are compared with the traditional pattern identification methods show that the GA-based model identification method provides a more correct model and more accurate results. In addition, GA-based model identification is more flexible then the traditional pattern identification methods in SARIMA models.

Table A.1
Repeat GA five times in GNP data set

Iterative	Model	AIC	SBC	SSE	Variance	Standard error
1	$p = 1; q = (2,5)$	−1125.67	−1112.99	0.016429	0.000096	0.009773
2	$p = (4); q = 2$	−1123.40	−1110.72	0.016642	0.000097	0.009837
3	$p = 0; q = 2$	−1124.48	−1114.97	0.016729	0.000097	0.009834
4	$p = 1; q = (2)$	−1125.88	**−1116.37**	0.016729	0.000096	0.009795
5	$p = 0; q = 2$	−1124.48	−1114.97	0.016729	0.000097	0.009834

Table A.2
Repeat GA five times in unemployment data set

Iterative	Model	AIC	SBC	SSE	Variance	Standard error
1	$p = (1,4,5); q = (4)$	−493.173	−477.406	0.552612	0.003289	0.057353
2	$p = (1,3,4,5); q = (3,4)$	−496.061	−473.988	0.531042	0.003199	0.05656
3	$p = (1,4,5); q = (3,4)$	−497.74	**−478.821**	0.5320264	0.003186	0.056443
4	$p = (1,4,5); q = (1,4)$	−495.312	−476.392	0.5395478	0.003231	0.05684
5	$p = (1,4,5); q = (3,4)$	−497.74	**−478.821**	0.5320264	0.003186	0.056443

Table A.3
Repeat GA five times in sales data set

Iterative	Model	AIC	SBC	SSE	Variance	Standard error
1	$p = 0, P = 1; q = 1, Q = (2,3)$	−29.0546	−17.7412	2.398247	0.036337	0.190623
2	$p = 1, P = 0; q = 0, Q = (1,3,4,5)$	−26.6147	−19.8266	2.625946	0.038617	0.196512
3	$p = 0, P = (4); q = 1, Q = 0$	−27.9293	−21.1412	2.577772	0.037908	0.194701
4	$p = 0, P = 0; q = (1,3,5), Q = 0$	−29.6915	−20.6408	2.4447345	0.036489	0.19102
5	$p = 0, P = 0; q = 3, Q = 0$	−30.9537	**−21.9030**	2.4016580	0.035846	0.189329

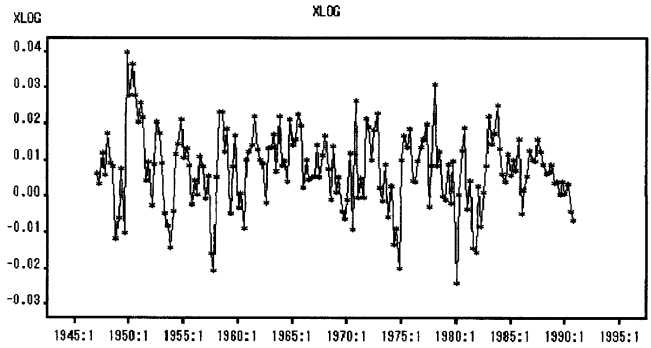

Fig. A.1. The time series graph of the GNP dataset.

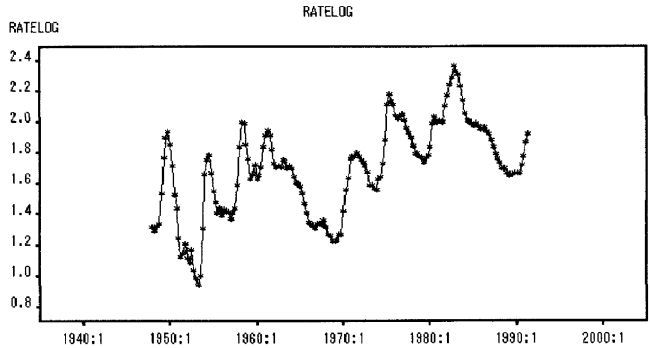

Fig. A.2. The time series graph of the unemployment dataset.

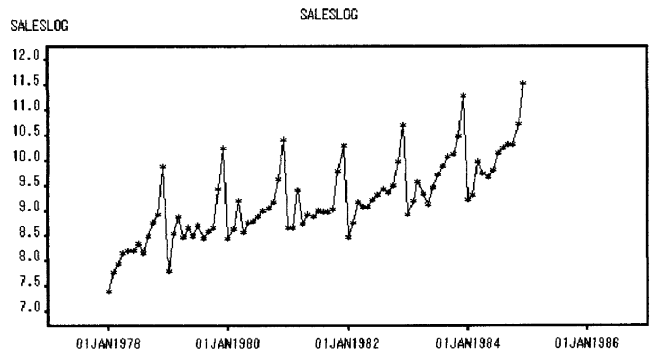

Fig. A.3. The time series graph of the sales dataset.

Table B.1
The 60 records in model 1

Date	Z_t	Date	Z_t	Date	Z_t	Date	Z_t
February 93	0.811594	May 94	1.477868	August 95	1.266903	November 96	1.790872
March 93	1.030023	June 94	1.27366	September 95	−0.92982	December 96	1.215285
April 93	1.286882	July 94	1.13069	October 95	0.793999	January 97	1.996357
May 93	−1.03849	August 94	0.432763	November 95	0.97822	February 97	2.11393
June 93	1.426607	September 94	0.076861	December 95	1.716972	March 97	2.96773
July 93	1.912128	October 94	0.495236	January 96	−1.63829	April 97	0.915401
August 93	0.161922	November 94	−1.20429	February 96	0.593432	May 97	1.227891
September 93	0.587937	December 94	−0.09032	March 96	1.218859	June 97	1.046889
October 93	−0.02159	January 95	−0.58174	April 96	0.181988	July 97	0.393808
November 93	0.551486	February 95	−1.09846	May 96	0.306072	August 97	1.275309
December 93	1.711534	March 95	−1.78524	June 96	2.461848	September 97	2.069215
January 94	1.821313	April 95	0.670302	July 96	−0.26715	October 97	1.755848
February 94	1.878967	May 95	−0.59777	August 96	1.059022	November 97	0.764436
March 94	2.625465	June 95	−0.14252	September 96	0.709361	December 97	3.680563
April 94	1.487225	July 95	−0.25121	October 96	0.066163	January 98	1.326553

Table B.2
The 60 records in model 2

Date	Z_t	Date	Z_t	Date	Z_t	Date	Z_t
February 93	1.545383	May 94	−0.65247	August 95	1.946484	November 96	0.744231
March 93	1.229991	June 94	−0.92949	September 95	−1.18926	December 96	0.566487
April 93	1.800354	July 94	−1.44876	October 95	1.467471	January 97	0.829815
May 93	−1.83627	August 94	−1.75261	November 95	0.875147	February 97	1.488865
June 93	0.667611	September 94	−1.28734	December 95	1.006989	March 97	1.536786
July 93	0.782234	October 94	−0.73312	January 96	−2.1772	April 97	−1.29377
August 93	−1.28207	November 94	−2.19464	February 96	0.947175	May 97	−0.5614
September 93	0.665923	December 94	−0.28297	March 96	−0.06152	June 97	−1.46442
October 93	−0.28474	January 95	−0.46296	April 96	−1.35372	July 97	−2.34622
November 93	−1.21194	February 95	−0.8699	May 96	0.330395	August 97	−0.59308
December 93	0.81587	March 95	−0.79588	June 96	2.976061	September 97	0.999679
January 94	1.278936	April 95	2.057517	July 96	−1.82776	October 97	0.430324
February 94	1.370175	May 95	−0.12609	August 96	0.525771	November 97	0.016339
March 94	2.142929	June 95	1.265879	September 96	0.152435	December 97	2.836601
April 94	−0.28252	July 95	1.435106	October 96	−1.58916	January 98	−1.03762

Table B.3
The 60 records in model 3

Date	Z_t	Date	Z_t	Date	Z_t	Date	Z_t
February 93	−0.31411	May 94	−0.04562	August 95	1.816242	November 96	1.941682
March 93	2.403397	June 94	−1.2235	September 95	0.212156	December 96	−1.02271
April 93	0.944848	July 94	−0.84921	October 95	−0.1503	January 97	1.706559
May 93	−1.93778	August 94	−0.95784	November 95	1.1896	February 97	1.016562
June 93	1.496522	September 94	−1.87542	December 95	1.160518	March 97	0.958278
July 93	0.685673	October 94	−0.07479	January 96	−1.81756	April 97	0.003486
August 93	−1.07005	November 94	−1.8203	February 96	−0.78164	May 97	−1.59705
September 93	0.134207	December 94	−0.35589	March 96	1.895245	June 97	−0.46377
October 93	−1.5533	January 95	−0.3521	April 96	−0.78021	July 97	−1.44992
November 93	1.335305	February 95	−0.96591	May 96	−0.24719	Aug 97	−0.17826
December 93	1.031615	March 95	−0.59516	June 96	0.960484	September 97	−0.231973
January 94	−0.33564	April 95	1.143206	July 96	0.264783	October 97	0.852753
February 94	1.631217	May 95	0.956505	August 96	0.488508	November 97	−0.28826
March 94	1.387184	June 95	0.319077	September 96	−0.80925	December 97	2.31946
April 94	0.238523	July 95	0.427438	October 96	−0.91081	January 98	0.047635

Although GA needs more computer time in the model identification stage, this can easily be overcome with current technology and accurate forecasting results are the only purpose. In addition, incorrect model identification will result in incorrect model estimation and increase the cost of model re-identification.

Appendix A

We repeated GA procedures five times to assess the quality of the GA in all example data sets and the results are shown in Tables A.1–A.3. Based on the results, the *SBC* which is fitness index in GA can get better results than the traditional methods which are described in content and all example data sets also pass residual test. These results also indicate the GA is a good technique in model selection in ARIMA family models.

The graph of GNP dataset is plotted in Fig. A.1.

The graph of unemployment dataset is plotted in Fig. A.2.

The graph of sales dataset is plotted in Fig. A.3.

Appendix B

The 60 records derived by each model are obtained as Tables B.1–B.3.

References

[1] G.E.P. Box, G.M. Jenkins, Time Series Analysis: Forecasting and Control, Holden-Day, San Francisco, 1976.
[2] R. McCleary, R. Hay, D. McDowell, Applied Time Series Analysis for the Social Sciences, Sage, Los Angeles, 1980.
[3] C.F. Ansley, An algorithm for the exact likelihood of a mixed autoregressive moving average process, Biometrika 66 (1) (1979) 59–65.
[4] C.F. Ansley, Finite sample properties of estimators for autoregressive moving average models, Journal of Econometrics 13 (1) (1980) 159–185.
[5] M. Morf, G.S. Sidhu, T. Kailath, Some new algorithms for recursive estimation on constant linear discrete time systems, IEEE Transactions on Automatic Control 19 (4) (1974) 315–323.
[6] J.G. Pearlman, An algorithm for the exact likelihood of a high-order autoregressive-moving average process, Biometrika 67 (1) (1980) 232–233.
[7] S.C. Hillmer, G.C. Tiao, Likelihood function of stationary multiple autoregressive-moving average models, Journal of the American Statistical Association 74 (3) (1979) 652–660.
[8] G.M. Ljung, G.E.P. Box, The likelihood function of stationary autoregressive-moving average models, Biometrika 66 (2) (1979) 265–270.
[9] P. Newbold, The exact likelihood function for a mixed autoregressive-moving average process, Biometrika 61 (3) (1974) 423–426.
[10] R. Biondini, Y. Lin, Estimating the Hurst parameter in fractional ARIMA (p, d, q), Mathematics and Computers in Simulation 48 (4) (1999) 407–416.

[11] C. Chatfield, Time-Series Forecasting, Chapman & Hall/CRC, Florida, 2001.
[12] H. Akaike, A new look at the statistical model identification, IEEE Transactions on Automatic Control 19 (6) (1974) 716–723.
[13] H. Akaike, Time Series Analysis and Forecasting. The Box–Jenkins Approach, Butterworths, London, 1978.
[14] H. Akaike, Fitting autoregressive models for prediction, Annals of the Institute of Statistical Mathematics 21 (2) (1969) 243–247.
[15] E.J. Hannan, B.G. Quinn, The determination of the order of an autoregression, Journal of the Royal Statistical Society B 41 (2) (1979) 190–195.
[16] G. Schwarz, Estimating the dimension of a model, Annals of Statistics 6 (2) (1978) 461–464.
[17] W.A. Woodward, H.L. Gray, On the relationship between the *S* array and the Box–Jenkins method of ARMA model identification, Journal of the American Statistical Association 76 (3) (1981) 579–587.
[18] J.M. Beguin, C. Gourieroux, A. Monfort, Identification of a mixed autoregressive-moving average process: the Corner method, in: O.D. Anderson (Ed.), Time Series, North-Holland, Amsterdam, 1980.
[19] R.S. Tasy, G.C. Tiao, Consistent estimates of autoregressive parameters and extended sample autocorrelation function for stationary and nonstationary ARMA model, Journal of the American Statistical Association 79 (1) (1984) 84–96.
[20] R.S. Tasy, G.C. Tiao, Use of canonical analysis in time series model identification, Biometrika 72 (2) (1985) 299–315.
[21] E.J. Hannan, J. Rissanen, Recursive estimation of mixed autogressive-moving average order, Biometrika 69 (1) (1982) 81–94.
[22] W.S. Chan, A comparison of some of pattern identification methods for order determination of mixed ARMA models, Statistics & Probability Letters 42 (1) (1999) 69–79.
[23] J. McClave, Subset autoregression, Technometrics 17 (3) (1975) 213–220.
[24] H.M. Krolzig, D.R. Hendry, Computer automation of general-to-specific model selection procedures, Journal of Economic Dynamics & Control 25 (6/7) (2001) 831–866.
[25] R. Chen, R.S. Tsay, Nonlinear additive ARX models, Journal of the American Statistical Association 88 (3) (1993) 955–967.
[26] J.C. Chao, P.C.B. Phillips, Model selection in partially nonstationary vector autoregressive processes with reduced rank structure, Journal of Econometrics 91 (2) (1999) 227–271.
[27] P. Winker, Optimized multivariate lag structure selection-using the global optimization heuristic threshold accepting, Computational Economics 16 (l/2) (2000) 87–103.
[28] J.M. Holland, Adaptation in Natural and Artificial Systems, The University of Michigan Press, Ann Arbor, 1975.
[29] D.E. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley, Reading, MA, 1989.
[30] L. Davis, Handbook of Genetic Algorithms, Van Nostrand Reinhold, 1991.
[31] J.R. Koza, Genetic Programming, The MIT Press, Cambridge, MA, 1992.
[32] Z. Michalewicz, Genetic Algorithms + Data Structures = Evolution Programs, Springer-Verlag, 1992.
[33] B.S. Chen, B.K. Lee, S.C. Peng, Maximum likelihood parameter estimation of F-ARIMA processes using the genetic algorithm in frequency domain, IEEE Transaction on Signal Processing 50 (9) (2002) 2208–2220.
[34] B. Wu, C.L. Chang, Using genetic algorithms to parameters (d,r) estimation for threshold autoregressive models, Computational Statistics & Data Analysis 38 (3) (2002) 315–330.
[35] M. Beenstock, G. Szpiro, Specification search in nonlinear time-series models using the genetic algorithm, Journal of Economic Dynamics & Control 26 (5) (2002) 811–835.
[36] C.R. Morf, G.S. Sidhu, T. Kailath, Some new algorithms for recursive estimation on constant linear discrete time systems, IEEE Transactions on Automatic Control 19 (4) (1974) 315–323.

[37] C. Chatfield, Time-Series Forecasting, Chapman & Hall/CRC, London/Boca Raton, 2001.
[38] S. Doreisha, T. Pukkila, Fast linear estimation methods for vector autoregressive moving average models, Journal of Time Series Analysis 10 (4) (1989) 325–339.
[39] B.G. Quinn, Order determination for a multivariate autoregression, Journal of the Royal Statistical Society Series B 42 (2) (1980) 182–185.
[40] H. Spliid, A fast estimation for the vector autoregressive moving average models with exogenous variables, Journal of the American Statistical Association 78 (4) (1983) 843–849.
[41] T.C. Mills, The Econometric Modeling of Financial Time Series, second ed., Cambridge University Press, Cambridge, 1999.
[42] J. Faraway, C. Chatfield, Time series forecasting with neural networks: a comparative study using the airline data, Applied Statistics 47 (2) (1998) 231–250.
[43] E.J. Hannan, J. Rissanen, Recursive estimation of mixed autogressive-moving average order, Biometrika 69 (1) (1982) 81–94.
[44] G.E.P. Box, G.M. Jenkins, G.C. Reinsel, Time Series Analysis, Forecasting and Control, third ed., Prentice-Hall, New Jersey, 1994.
[45] D. Bhandari, C.A. Murthy, S.K. Pal, Genetic algorithm with elitist model and its convergence, International Journal of Pattern Recognition Artificial Intelligence 10 (6) (1996) 731–747.
[46] D.A. Dickey, D.P. Hasza, W.A. Fuller, Testing for unit roots in seasonal time series, Journal of the American Statistical Association 79 (2) (1984) 355–367.
[47] D.A. Dickey, W.A. Fuller, Distribution of the estimators for autoregressive time series with a unit root, Journal of the American Statistical Association 74 (2) (1979) 427–431.
[48] D. Pena, G.C. Tiao, R.S. Tsay, A Course in Time Series Analysis, Wiley, New York, 2001.
[49] S. Makridakis, S.C. Wheelwright, R.J. Hyndman, Forecasting: Methods and Applications, third ed., Wiley, New York, 1998.