Research Article

A Genetic Algorithm Approach for Prediction of Linear Dynamical Systems

Za'er Abo-Hammour,¹ Othman Alsmadi,² Shaher Momani,^{3,4} and Omar Abu Arqub⁵

¹ Department of Mechatronics Engineering, Faculty of Engineering, The University of Jordan, Amman 11942, Jordan

² Department of Electrical Engineering, Faculty of Engineering, The University of Jordan, Amman 11942, Jordan

³ Department of Mathematics, Faculty of Science, The University of Jordan, Amman 11942, Jordan

⁴ Department of Mathematics, Faculty of Science, King AbdulAziz University, Jeddah 21589, Saudi Arabia

⁵ Department of Mathematics, Faculty of Science, Al Balqa Applied University, Salt 19117, Jordan

Correspondence should be addressed to Shaher Momani; s.momani@ju.edu.jo

Received 19 June 2013; Accepted 9 October 2013

Academic Editor: Ben T. Nohara

Copyright © 2013 Za'er Abo-Hammour et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Modelling of linear dynamical systems is very important issue in science and engineering. The modelling process might be achieved by either the application of the governing laws describing the process or by using the input-output data sequence of the process. Most of the modelling algorithms reported in the literature focus on either determining the order or estimating the model parameters. In this paper, the authors present a new method for modelling. Given the input-output data sequence of the model in the absence of any information about the order, the correct order of the model as well as the correct parameters is determined simultaneously using genetic algorithm. The algorithm used in this paper has several advantages; first, it does not use complex mathematical procedures in detecting the order and the parameters; second, it can be used for low as well as high order systems; third, it can be applied to any linear dynamical system including the autoregressive, moving-average, and autoregressive moving-average models; fourth, it determines the order and the parameters in a simultaneous manner with a very high accuracy. Results presented in this paper show the potentiality, the generality, and the superiority of our method as compared with other well-known methods.

1. Introduction

Modelling of linear dynamical systems is encountered in many fields including financial markets, environmental sciences, control engineering, and many other fields (Zivot and Wang [1], Landau and Zito [2]). The primary goal of modelling of physical and real-life problems is for the purpose either controlling the process or performing future forecasts. The modelling process can be achieved using one of two main scenarios: the first scenario is based on the complete understanding of the physical process that leads to the derivation of the governing differential equations describing the process. In this case, the model might be fully known in terms of the order and the parameters or might be partially known where some or all of the parameters are unknown. In the second scenario, the process is considered as a black box where some information might be available about the order

of the process and some of the parameters, or it might be fully unknown in terms of the order and the parameters of the process (Landau and Zito [2]).

Modelling of linear dynamical systems leads to either a transfer function or a state space representation (Landau and Zito [2], Garcia-Hiernaux et al. [3]). In case of transfer functions, autoregressive (AR), moving-average (MA), or autoregressive moving-average (ARMA) processes are the results of the modelling process of the physical and real-life problems (Wahlberg et al. [4], AbdulRahim et al. [5], and Dahlen [6]). The previous models play an important role in system identification and have a wide range of applications such as communication, signal processing, control systems, biomedical engineering, image processing and compression, prediction of spectrum estimation, and controlling of dynamic systems (Fenga and Politis [7], Broersen and de-Waele [8]).

It is well known, in general, that the spectral density of a stationary stochastic process may have high peaks as well as deep valleys which are referred to as zeros and poles. The spectral densities of AR models typically have high peaks while, on the other hand, the MA models are characterised by having deep valleys. As a result, it is difficult for AR models to reproduce deep valleys and difficult for MA models to reproduce high peaks when each of these model classes is used to approximate some processes as stated by Dahlen [6].

In AR and MA models, the objective of the modelling process is to determine the order of the AR model given as (p) and the order of the MA model given as (q) , where p refers to the number of the model poles and q refers to the number of the model zeros. In addition to that, one main objective of modelling is to determine the coefficients of the considered model. However, the model coefficients cannot be found without knowing the order of such model. In most cases, the user assumes the order and then the parameters are estimated according to this assumption. However, in practice, the order is unknown, so it has to be properly found in order to estimate the correct parameters of the model.

The ARMA (p, q) model can generally be represented by a transfer function with p presenting the number of poles and q representing the number of zeros. Many researchers have concentrated on the issue of calculating the model order of the ARMA processes. Most recently, Abo-Hammour et al. [9] introduced a new approach for ARMA modelling. Al-Smadi and Al-Zaben [10] and Broersen [11] have also used some techniques for modelling of ARMA processes. The most well-known methods for ARMA modelling, however, include the final prediction error introduced by Akaike [12], Akaike information criterion (AIC) presented by Akaike [13], the minimum description length (MDL) introduced by Rissanen [14] and Hannan [15], and the minimum eigenvalue criterion (MEV) introduced by Liang et al. [16]. In some of these methods, the order is computed from the prediction error variance. As investigated by Abo-Hammour et al. [9], it was concluded that these approaches can be computationally expensive. On the other hand, the MEV is derived from the MDL criterion and it is based on the minimum eigenvalue of a covariance matrix computed from the observed data. In MEV, neither the parameters nor the prediction error variance needs to be estimated, and it has reduced computation time compared to other MDL-based model order determination schemes. This approach also provides more accurate estimates of the true model orders than the previous methods.

Genetic algorithm (GA), which will be used in this paper, has been previously applied for the purpose of autoregressive moving-average modelling. However, it was used for either parameter estimation or order estimation of a given model. Rolf et al. [17] calculated the order of the model using the empirical autocorrelation function and the empirical partial autocorrelation function. The parameters were estimated using GA but with complex calculations in fitness function. Nevertheless, this method was only useful for low model orders and did not provide a reliable tool for model identification since only 20% of the models have been identified correctly as they have stated. Recently, Ursu and Turkman [18] introduced a GA approach for periodic

AR models identification. They introduced an automatic procedure using the Bayesian information to identify the order of the AR model. Baragona et al. [19] introduced a GA to estimate the parameters of a self-exciting threshold subset of autoregressive moving-average models. The threshold model is composed of several linear AR and MA models where each one of these models is applied according to a switch mechanism depending on the delayed observation and some threshold values. Chang [20] introduced a new modification in the GA operation to estimate the parameters of autoregressive moving-average models. The results were accurate but the process order was assumed before the estimation of the parameters.

The previous paragraphs show that most of the available methods, including GA, for the order estimation are either computational intensive or work for low order models. The motivation of the work presented in this paper is embedded in the needs for a robust, general purpose algorithm that is capable of simultaneous estimation of the correct model order and parameters. Hence, in this paper, we present an extension of the previously proposed ARMA model algorithm introduced by Abo-Hammour et al. [9] to deal with AR and MA models and thus to provide a general umbrella for the simultaneous estimation of the order and parameters of any linear dynamical system. As a result, it should estimate the model (order and parameters) for low as well as high order models. Our approach depends on GA and simple calculations of the fitness function to estimate the model order and parameters. In the fitness function, the error is calculated using the least square error. The model order is detected without any prior assumptions. Because of these simple calculations, high order models can be solved using this methodology, where the parameters of the model are directly obtained according to the detected order.

The organization of this paper is as follows. In the next section, we formulate the problem as an optimization problem. Section 3 covers the description of GA. Discussion and illustrations are given in Section 4 in order to verify the mathematical simulation of the proposed algorithm. Finally, conclusions are presented in Section 5.

2. Problem Formulation

Consider a discrete time process of a univariate linear time-invariant system having an input data sequence $x(n)$ and an output data sequence $y(n)$. The process is assumed to be of stationary and ergodic type. The general autoregressive moving-average model is described in the form of a p th order difference equation as given by

$$y(n) = - \sum_{i=1}^p a_i y(n-i) + \sum_{j=0}^q b_j x(n-j). \quad (1)$$

The AR model can be represented in the form of a p th order difference equation as follows:

$$y(n) = - \sum_{i=1}^p a_i y(n-i) + x(n), \quad (2)$$

FIGURE 1: GA model estimation architecture.

where p is the number of the AR model poles and the coefficients a_i are the AR parameters, as presented by Al-Smadi [21]. On the other hand, the MA model can be represented in the form of a q th order difference equation as follows:

$$y(n) = \sum_{j=0}^q b_j x(n-j), \quad (3)$$

where q is the number of the MA zeros and the coefficients b_j are the MA parameters.

It is important to mention that the input $x(n)$ is a zero-mean, stationary, and independent identically distributed data sequence (Liang et al. [16]), while the output data sequence is assumed to be stationary and ergodic where this can be achieved by satisfying the following conditions (Zivot and Wang [1]).

- (1) An AR process is stationary and ergodic provided that the roots of the characteristic equation lie inside the complex unit circle (having modulus less than one, or the process is stable).
- (2) The MA model is stationary and ergodic provided that its coefficients are finite, and the MA model is invertible if all of the roots of the MA characteristic polynomial lie inside the complex unit circle.
- (3) An ARMA model is stationary and ergodic if the roots of the characteristic equation lie inside the complex unit circle (stable process corresponding to (1)), and it is invertible if the roots of the MA characteristic polynomial lie inside the unit circle. The ARMA model is also assumed to have no pole-zero cancellation.

The models can be written in Z-domain as transfer functions to produce the following AR and MA models, respectively:

$$H(z) = \begin{cases} \frac{1}{A(z)} = \frac{1}{\sum_{i=0}^p a_i z^{-i}} & \text{with } a_0 = 1 \text{ for AR,} \\ B(z) = \sum_{j=0}^q b_j z^{-j} & \text{for MA.} \end{cases} \quad (4)$$

For the AR and MA models in (4) and the ARMA model given in (1), the signal observed with an additive white Gaussian noise $v(n)$ is given by

$$y_t(n) = y(n) + v(n). \quad (5)$$

3. Genetic Algorithm

GA performs its tasks based on the Darwinian concepts of natural selection and evolution. GA is well applicable in many scientific fields and proven to provide very powerful optimization algorithms (Goldberg [22]).

The GA modelling approach for the order and parameter estimation is performed as illustrated in Figure 1 and can be described in the following recursive sequence (Abo-Hammour [23, 24], Abo-Hammour et al. [25–29], Abu Arqub [30], and Abu Arqub et al. [31, 32]).

(1) *Initialization*. An initial population is randomly generated with a specific population size. The population is performed from a set of individuals (chromosomes). Each chromosome contains a group of genes where each gene represents a variable.

(2) *Evaluation*. In a GA, a fitness function is used to evaluate the degree of goodness of a chromosome. The fitness function must be devised for each problem to be solved. It is the part that characterizes the GA application for any specific problem type. The fitness function used in this paper is given as (Nissinen et al. [33])

$$\text{Fitness} = \frac{1}{1 + \text{MSE}}, \quad (6)$$

where MSE is mean square error obtained based on the true system output and the estimated GA model output. The MSE is given by

$$\text{MSE} = \frac{1}{N} \sum_{n=0}^N (y_t(n) - \hat{y}(n))^2, \quad (7)$$

where N is the number of samples and $\hat{y}(n)$ is the GA estimated output.

(3) *Selection*. In the selection process, the population fitness and associated chromosomes are ranked from the highest fitness to the lowest fitness. Only the best are selected to continue, while the rest are left. The selection rate is the fraction of population that survives for the next step of mating. Deciding how many chromosomes to keep is somewhat arbitrary; however, we often keep 50% in the natural selection process.

(4) *Crossover*. Each pair of chromosomes is taken and cut at some randomly chosen point to perform two segments

TABLE 1: Selected values GA estimation approach.

Parameter name	Parameter value
Mutation rate	0.02
Crossover rate	0.75
Replacement factor	0.5
Selection rate	0.5
Population size	1000

in each chromosome. The segments are then swapped over to produce two new full length chromosomes. Crossover is not usually applied to all pairs of chromosomes selected for mating. A random crossover choice is made and is typically between 0.6 and 1.0. Different methodologies exist for crossover operation; in this paper, the multipoint crossover with five distinct points will be used.

(5) *Mutations*. This operation is applied to each child individually. It randomly flips any individual element with a small mutation rate (between 0.1 and 0.001) to guard against premature convergence.

(6) *Replacement*. After evaluating the new children population using the fitness function and ranking the chromosomes in a descending order, replacement operation takes place. The parent population is totally or partially replaced by the children population. In the algorithm, the rank-based overlapping replacement scheme is used. In this scheme, the next population is produced by selecting the best, highest fitness, N_p members of both populations, and the replacement factor.

(7) *Termination*. GA is terminated when some convergence criterion is met. The most common GA convergence criteria are the maximum number of generations and fitness value.

For the GA modelling procedure, the algorithm starts with suggesting an order r for p (in the case of AR model) in (4) and an order r for q (in the case of MA model) in (4) (Broersen [11]). This means that there are an r number of parameters per individual as seen in Figures 2(a) and 2(b) for the AR and MA models, respectively. This step is a soft constraint that can be modified (increase or decrease) when necessary if the application requires that. In every generation, the algorithm adaptively determines the order of the model from the best of generation individuals according to the drop in the parameters values. For a detailed illustration, the following may be stated.

For the AR model, the order p is determined if there is a parameter (a_l) in (4) such that its value is greater than those of all of the remaining coefficients (a_k) by at least a factor of r . This means that there exists a value of l such that $a_l/a_k \geq r$ for all k/l . All of the parameters of the AR model, from a_{l+2} to a_r , are set to zero in all individuals in the population. The element a_{l+1} is kept nonzero to give indications if an increase in the order is required according to the strategy applied in step 2 throughout the evolution process.

For the MA model, the order q is determined if there is a parameter (b_j) in (4) such that its value is greater than those of all of the remaining coefficients (b_k) by at least a factor of r . Again, this indicates that there exists a value of j such that $b_j/b_k \geq r$ for all $k > j$. All of the parameters of the MA model, from b_{j+2} to b_r , are set to zero in all individuals in the population. The element b_{j+1} is kept nonzero to give indications if an increase in the order is required according to the strategy applied in the evaluation process.

As a result of the above illustrative points, the estimated order of the AR model is ($p = l$) and the order of the MA model is ($q = j$). In addition to that, as the order is determined, all of the model parameters are determined as well.

4. Discussion and Illustration

In this section, we will consider different examples to illustrate the presented work. The GA used in this paper is used to find the order and parameters of AR, MA, and ARMA model. Thus, the applicability and efficiency of the algorithm will be demonstrated to cover all types of linear dynamical systems.

The input data to the algorithm is divided into two parts; the dynamical model related parameters and the GA related parameters. The dynamic model data are as follows: the input-output data sequence of the process, the number r representing the highest possible order for the model, and based on that number (a_i) for the AR model and (b_j) for the MA model that will be generated by the algorithm. It is to be noted that r is selected to have a value of 10 in all the simulation results that appear in this paper. However, this is not a hard constraint for the algorithm since the value of r can be increased or reduced according to the maximum possible order that we expect from the real model.

The GA related parameters, on the other hand, include the population size, N_p , the crossover probability, P_c , the mutation probability, P_m , and the replacement factor. The settings of these parameters for all examples covered in this paper are given in Table 1. The GA modelling analysis is performed based on the average fitness, average MSE, and the average number of generations required for convergence. Due to the stochastic nature of GA, different runs are made with arbitrary number using different random generation seed. The average of these runs will be calculated for the result obtained from the model parameters. Also, the average number of generations, the average total fitness, and the average MSE are also obtained.

Problem related analysis is divided into two simultaneous parts: order detection analysis and parameters optimization analysis. Parameters optimization analysis consists of convergence analysis for the parameters of the dynamical model. The algorithm detects the true order and the true parameters in a simultaneous recursive manner. Different runs will be made for a given dynamical model to detect the true order before operating parameter estimation.

Based on the working principle of the algorithm shown in Figure 2 for the AR and MA cases, it is to be noted that the basic cycle of the GA is of 100 generations. During this

FIGURE 2: GA model identification and parameter estimation for (a) AR model and (b) MA model.

cycle, the algorithm will start from scratch and estimate the parameters of the model based on the mean square error for the assumed model order that will be r for the initial runs. When the generation number reaches 100, then the algorithm will take the best 5 individuals and check for reduction in p for the case of AR, or a reduction in q for the case of MA. After the reduction process, the algorithm will start again from scratch and estimate the parameters of the model based on the new reduced model order. This means that after every reduction process, there will be a random initialization of the values of the parameters, which in return leads to a sharp impulse in the fitness value at this stage and a sharp impulse in the parameter values. However, the impulse jump in the fitness value will be generally in the downward direction which means a reduction in the fitness value at this point, while the sharp impulse in the parameter values will be in the upward and downward directions due to the oscillation in the parameter values at the first few generations of every 100 generation cycle.

4.1. AR Model. In this subsection, we consider an AR model of a 5th order. The model is given by

$$\begin{aligned} & \left(1 + 0.34z^{-1} + 0.543z^{-2} + 0.92z^{-3} + 1.3z^{-4} + 0.98z^{-5}\right) \\ & \times y(z) = x(z), \end{aligned} \quad (8)$$

with five poles.

Performing order and parameter estimation with suggested $r = 10$, the AR-GA approach was able to successfully predict the order of the model and to estimate its parameters, which were in fact identical to the true model. The model has converged to its final values with a fitness value of 100%. Figure 3 shows the convergence of the model coefficients including the model order. As seen in this figure, the coefficients a_6 up to a_{10} , which are not really part of the true model, have converged to the zero value very quickly, while on the other hand, the other coefficients took longer time to converge to their final values.

FIGURE 3: Convergence of AR model parameters, a_1 to a_{10} .

FIGURE 4: Convergence of MA model parameters, b_1 to b_{10} .

FIGURE 5: Fitness of the estimated ARMA model for all generation history.

The AR-GA approach was also tested for the order and the parameter estimation with different signal to noise ratio additive white Gaussian noise. The AR-GA was simulated for 20 runs and the correct run results are presented as shown in Table 2. As observed, the results show an accurate performance of the method for the AR model order estimation with all recorded runs giving an exact estimation of the order. Table 2 also shows the comparisons of the AR-GA method with the other commonly used methods for estimating AR models' order, and as seen, the superiority of the new approach is clearly obvious.

4.2. MA Model. In the following illustration, we consider a 4th order MA model given by

$$y(z) = (1 + 0.25081z^{-1} + 0.10424z^{-2} + 0.53387z^{-3} + 0.18593z^{-4})x(z). \quad (9)$$

Using the MA-GA approach to estimate this model with suggested $r = 10$, the MA-GA was able to predict the order of the model as well as estimating parameters with a 100% fitness value. Performance of the estimation progress is shown in Figure 4 with result observations as previously analyzed and concluded.

As previously done, the MA-GA was tested for order and parameter estimation with different signal to noise ratio additive white Gaussian noise. The MA-GA was simulated for 20 runs with results presented as shown in Table 3. The results are obtained with similar observations as in the previous examples; superiority of the new approach is clearly obvious.

4.3. ARMA Model. The ARMA model is given by the following equation:

$$\begin{aligned} & (1 + 0.3452z^{-1} + 0.53z^{-2} + 0.3985z^{-3} + 0.8138z^{-4})y(z) \\ &= (1 - 0.3055z^{-1} + 0.16328z^{-2})x(z), \end{aligned} \quad (10)$$

with four poles and two zeros.

TABLE 2: Comparison between different methods and SNRs for AR model order estimation using 20 runs.

Method used	SNR = 50	SNR = 30	SNR = 20	SNR = 10
MDL	0	0	0	0
AIC	0	0	0	0
MEV	20	20	11	0
GA	20	20	20	20

TABLE 3: Comparison between different methods and SNRs for MA model order estimation using 20 runs.

Method used	SNR = 50	SNR = 30	SNR = 20	SNR = 10
MDL	20	18	18	17
AIC	13	10	12	13
MEV	20	20	19	17
GA	20	20	20	20

Performing order and parameter estimation with suggested $r = 10$, the ARMA-GA approach was able to successfully predict the order of the model and to estimate the model parameters. In fact, the estimated model was obtained with a fitness value of 99.999999999%. Figure 5 shows the fitness convergence throughout the history of generations. Notice that, at later stages, it converges more quickly to almost 100 and that is because the order is getting closer to that of the true model. Figure 6 shows the convergence of the numerator coefficients toward the true model parameters versus the generation number. Notice that the higher order coefficients (i.e., b_3, b_4 and similarly b_5 to b_{10}) converge faster to zero, while on the other hand, the parameters that truly make the model (i.e., b_1 and b_2) converge with larger number of generations. This is because the ARMA-GA method has to deal with the unnecessary parameters first. It is also to be noted that the GA performs the reduction of the parameters number each 100 generations. This fact leads to the generation of different spikes, as seen in Figure 6, since all of the non-vanishing parameters will be randomly generated after the reduction operation. This process will be repeated until the first occurrence where the true orders of the numerator and denominator are found. Then, two more additional runs are performed using the GA-ARMA algorithm to insure that there is no more order reduction and that we have reached the exact true order and parameters of the given system. Figure 7 shows the convergence of the denominator coefficients with similar observations as in the numerator coefficient convergence.

Next, we consider the same model with different noise ratios as seen in Table 4. The introduced approach is compared with the most commonly used methods of ARMA modelling to provide an overall approach validation. The given true model is tested with an additive white Gaussian noise of varying degrees. Each method was run for a total of 20 trials and the number of correct order estimations was returned as shown in Table 4. Please notice that the MDL and AIC failed to estimate the correct order in all trials and for all values of SNRs, although they can provide some better

FIGURE 6: Convergence of ARMA numerator parameters, b_1 to b_{10} .

FIGURE 7: Convergence of ARMA denominator parameters, a_1 to a_{10} .

TABLE 4: Comparison between different methods and SNRs for ARMA model order estimation using 20 runs.

Method used	SNR = 50	SNR = 30	SNR = 20	SNR = 10
MDL	0	0	0	0
AIC	0	0	0	0
MEV	20	20	3	1
GA	20	20	20	20

results for different models (MA models). This fact has been observed by Liang et al. [16] and Al-Smadi and Wilks [34] as the MDL and AIC are not reliable for the ARMA or AR modelling. For the MEV criterion, we were able to estimate the order in all of the runs for the 50 and 30 SNRs, but failed for lower SNRs. However, using the ARMA-GA approach, we succeeded in estimating the order for all the runs and for all the ranges of SNRs.

5. Conclusions

AR-GA and MA-GA modelling approach is introduced in this paper as novel system identification and modelling methodology of linear dynamical systems. The introduced algorithm extends the ARMA-GA approach previously proposed by the authors to the AR and MA models which results in a successful modelling of all types of linear dynamical systems. The modelling process depends mainly on the fitness function which uses only the MSE obtained based on the actual and estimated system outputs. The algorithm performs the parameter estimation with a relatively very small error as it first provides the correct model order. The GA modelling approach was applied to AR, MA, and ARMA models with different SNRs of 50, 30, and 10. The results obtained based on the proposed approach are superior to the reported methods in terms of accuracy, repeatability, and robustness. The new technique of dynamical system modelling (order and parameter estimation) is considered as an efficient search algorithm with much of potential for different types of system modelling.

References

- [1] E. Zivot and J. Wang, *Modelling Financial Time Series with S-PLUS*, Springer, New York, NY, USA, 2003.
- [2] L. D. Landau and G. Zito, *Digital Control Systems-Design, Identification, Implementation*, Springer, London, UK, 2006.
- [3] A. García-Hiernaux, J. Casals, and M. Jerez, "Estimating the system order by subspace methods," *Computational Statistics*, vol. 27, no. 3, pp. 411–425, 2012.
- [4] L. Wahlberg, P. Lindskog, and B. Wahlberg, "Applications of Kautz models in system identification," in *Proceedings of the 12th IFAC World Congress*, Sydney, Australia, 1993.
- [5] H. AbdulRahim, F. Ibrahim, and M. N. Taib, "System identification of nonlinear autoregressive models in monitoring dengue infection," *International Journal on Smart Sensing and Intelligent Systems*, vol. 3, no. 4, pp. 783–806, 2010.
- [6] A. Dahlen, *Identification of stochastic systems: subspace methods and covariance extension [Ph.D. thesis]*, Royal Institute of Technology, Stockholm, Sweden, 2001.
- [7] L. Fenga and D. N. Politis, "Bootstrap-based ARMA order selection," *Journal of Statistical Computation and Simulation*, vol. 81, no. 7, pp. 799–814, 2011.
- [8] P. M. Broersen and S. de-Waele, "Finite sample properties of ARMA order selection," *IEEE Transactions on Instrumentation and Measurement*, vol. 53, no. 3, pp. 645–651, 2004.
- [9] Z. S. Abo-Hammour, O. M. K. Alsmadi, A. M. Al-Smadi, M. I. Zaqout, and M. S. Sarairoh, "ARMA model order and parameter estimation using genetic algorithms," *Mathematical and Computer Modelling of Dynamical Systems*, vol. 18, no. 2, pp. 201–221, 2012.
- [10] A. Al-Smadi and A. Al-Zaben, "ARMA model order determination using edge detection: a new perspective," *Circuits, Systems, and Signal Processing*, vol. 24, no. 6, pp. 723–732, 2005.
- [11] P. M. Broersen, *Automatic Autocorrelation and Spectral Analysis*, Springer, London, UK, 2006.
- [12] H. Akaike, "Fitting autoregressive models for prediction," *Annals of the Institute of Statistical Mathematics*, vol. 21, no. 1, pp. 243–247, 1969.
- [13] H. Akaike, "Statistical predictor identification," *Annals of the Institute of Statistical Mathematics*, vol. 22, no. 1, pp. 203–217, 1970.
- [14] J. Rissanen, "Modeling by shortest data description," *Automatica*, vol. 14, no. 5, pp. 465–471, 1978.
- [15] E. J. Hannan, "The estimation of the order of an ARMA process," *Annals of Statistics*, vol. 8, no. 5, pp. 1071–1081, 1980.
- [16] G. Liang, D. M. Wilkes, and J. A. Cadzow, "ARMA model order estimation based on the eigenvalues of the covariance matrix," *IEEE Transactions on Signal Processing*, vol. 41, no. 10, pp. 3003–3009, 1993.
- [17] S. Rolf, J. Sprave, and W. Urfer, "Model identification and parameter estimation of ARMA models by means of evolutionary algorithms," in *Proceedings of the 1997 IEEE/IAFE Conference on Computational Intelligence for Financial Engineering*, pp. 237–243, March 1997.
- [18] E. Ursu and K. F. Turkman, "Periodic autoregressive model identification using genetic algorithms," *Journal of Time Series Analysis*, vol. 33, no. 3, pp. 398–405, 2012.
- [19] R. Baragona, F. Battaglia, and D. Cucina, "Estimating threshold subset autoregressive moving-average models by genetic algorithms," *International Journal of Statistics*, vol. 62, no. 1, pp. 39–61, 2004.
- [20] W. Chang, "Coefficient estimation of IIR filter by a multiple crossover genetic algorithm," *Computers and Mathematics with Applications*, vol. 51, no. 9–10, pp. 1437–1444, 2006.
- [21] A. Al-Smadi, "Automatic identification of ARMA systems," *International Journal of General Systems*, vol. 38, no. 1, pp. 29–41, 2009.
- [22] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, New York, NY, USA, 1989.
- [23] Z. S. Abo-Hammour, *Advanced continuous genetic algorithms and their applications in the motion planning of robotic manipulators and the numerical solution of boundary value problems [Ph.D. thesis]*, Quaid-Azam University, Islamabad, Pakistan, 2002.
- [24] Z. S. Abo-Hammour, "A novel continuous genetic algorithms for the solution of the cartesian path generation problem of robot manipulators," in *Robot Manipulators: New Research*, J. X. Lui, Ed., pp. 133–190, Nova Science, New York, NY, USA, 2005.

- [25] Z. S. Abo-Hammour, A. G. Asasfeh, A. M. Al-Smadi, and O. M. K. Alsmadi, "A novel continuous genetic algorithm for the solution of optimal control problems," *Optimal Control Applications and Methods*, vol. 32, no. 4, pp. 414–432, 2011.
- [26] Z. S. Abo-Hammour, O. M. Alsmadi, S. I. Bataineh, M. A. Al-Omari, and N. Affach, "Continuous genetic algorithms for collision-free Cartesian path planning of robot manipulators regular paper," *International Journal of Advanced Robotic Systems*, vol. 8, no. 6, pp. 14–36, 2011.
- [27] Z. S. Abo-Hammour, O. M. K. Alsmadi, and A. M. Al-Smadi, "Multi-time-scale systems model order reduction via genetic algorithms with eigenvalue preservation," *Journal of Circuits, Systems and Computers*, vol. 20, no. 7, pp. 1403–1418, 2011.
- [28] Z. S. Abo-Hammour, N. M. Mirza, S. M. Mirza, and M. Arif, "Cartesian path generation of robot manipulators using continuous genetic algorithms," *Robotics and Autonomous Systems*, vol. 41, no. 4, pp. 179–223, 2002.
- [29] Z. S. Abo-Hammour, M. Yusuf, N. M. Mirza, S. M. Mirza, M. Arif, and J. Khurshid, "Numerical solution of second-order, two-point boundary value problems using continuous genetic algorithms," *International Journal for Numerical Methods in Engineering*, vol. 61, no. 8, pp. 1219–1242, 2004.
- [30] O. Abu Arqub, *Numerical solution of fuzzy differential equation using continuous genetic algorithms [Ph.D. thesis]*, University of Jordan, Amman, Jordan, 2008.
- [31] O. Abu Arqub, Z. S. Abo-Hammour, and S. Momani, "Application of continuous genetic algorithm for nonlinear system of second-order boundary value problems," *Applied Mathematics and Information Sciences*, vol. 8, no. 1, pp. 1–14, 2014.
- [32] O. Abu Arqub, Z. S. Abo-Hammour, S. Momani, and N. Shawagfeh, "Solving singular two-point boundary value problems using continuous genetic algorithm," *Abstract and Applied Analysis*, vol. 2012, Article ID 205391, 25 pages, 2012.
- [33] A. S. Nissinen, H. N. Koivo, and H. Koivisto, "Optimization of neural network topologies using genetic algorithm," *Intelligent Automation and Soft Computing*, vol. 5, no. 3, pp. 211–224, 1999.
- [34] A. Al-Smadi and D. M. Wilkes, "Robust and accurate ARX and ARMA model order estimation of non-Gaussian processes," *IEEE Transactions on Signal Processing*, vol. 50, no. 3, pp. 759–763, 2002.