Coevolutionary Algorithm for Multivariable Discrete Linear Time-variant System Identification

Alexander E. Robles and Mateus Giesbrecht
School of Electrical and Computer Engineering, University of Campinas, Av. Albert Einstein 400, Campinas - SP, Brazil

Keywords: Evolutionary Algorithms, Coevolutionary Algorithms, System Identification, Time-variant System Identification.

Abstract: A significant part of the works in system identification is focused on time-invariant dynamic systems. However, most of systems in the real applications have nonlinear and time variant behavior. In this paper, we present a multivariable time-variant identification method based on a paradigm in the field of evolutionary algorithms: The coevolutionary algorithm. This new method focuses on the relationship between the fitness of an individual in relation to the fitness of the other individuals (or group of individuals), based on the principle of the selective pressure, that is part of the evolutionary process. A brief comparison between a multivariable deterministic identification method MOESP-VAR and the proposed coevolutionary method is shown. From the results it is possible to notice that the proposed method presents an accuracy higher than the obtained with the deterministic identification method.

1 INTRODUCTION

System identification consists on finding the mathematical models that can describe dynamic systems behavior, from observed input-output data (Ljung, 1999). A significant part of activities and researches in system identification focuses on time-invariant dynamic systems. However, there are innumerable systems in nature that are multivariable with nonlinear and time-varying behavior. To deal with last problem, many of the time-varying systems are approximated by linear time-invariant systems, as long as these systems vary slowly (Tamariz et al., 2005). In some cases these approximations cannot describe the real variant system behavior, so that the magnitude of the error between output data and output model is considerable.

The main focus of this paper is to present an evolutionary method to identify multivariable time-varying system (MTVS). For decades evolutionary algorithms (EAs) have been applied in optimization problems with successful results. Moreover, system identification area had one of its first applications in (Rodrigues-Vasquez and Fleming, 2004) when a classical genetic algorithm is applied to identify the parameters of the transfer function of a two masses resonant system. In (Wakizono and Uosaki, 2006) an approach for identifying linear and non-linear systems based on CLONALG algorithm (De Castro and Von Zuben, 2002) was proposed. Other recent work is (Giesbrecht and Bottura, 2015), in which an immuno-inspired algorithm was develop to identify MTVS. In that reference, the MTVS identification problem is defined as an optimization problem, where a search space is created by all possible matrices quadruples A,B,C,D that represent the MTVS in state-space model for a given time interval. Furthermore, each state-space model is seen as a point that minimizes the error between the system and the model outputs to the same input signal. In this paper, those principles were used to develop the coevolutionary algorithm (COEA) that can be applied to MTVS identification problem.

In this work, some comparisons are made between implemented COEA and deterministic method called MOESP-VAR (Tamariz et al., 2005), based on Multivariable Output-Error State sPace (MOESP) (Verhaegen and Dewilde, 1992). A genetic algorithm (GA) also was implement using the same parameters and operators used for COEA, but the populations do not evolve together. Trying to show the effectiveness of evolutionary processes, it was developed a random search (RS), that performs a searching without any evolutionary intelligence behind.

The paper is organized as follows. In the next section, subspace methods for system identification are introduced. In the section 3, the coevolutionary algorithms are briefly detailed. Section 4 describes COEA to identify MTVS, where the coevolution is the main topic. In the section 5 the results of the application

Robles, A. and Giesbrecht, M.
Coevolutionary Algorithm for Multivariable Discrete Linear Time-variant System Identification.
DOI: 10.5220/0006907602900295
In *Proceedings of the 15th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2018) - Volume 1*, pages 290-295
ISBN: 978-989-758-321-6

of the proposed method to identify a discrete slowly time-varying benchmark are shown. In the same section we compare COEA with other approaches such as GA, RS, and MOESP-VAR method. Finally, in the section 6, the conclusions of this paper and future works are exposed.

2 SUBSPACE METHODS FOR SYSTEM IDENTIFICATION

2.1 State-space Models

A multivariable system can be modeled in state-space representation. The discrete multivariable time-invariant system is defined in state-space model as follows:

$$\begin{cases} x(k+1) = Ax(k) + Bu(k) \\ y(k) = Cx(k) + Du(k) \end{cases} \tag{1}$$

Where $x(k) \in \mathbb{R}^n$ is defined as the system state at instant k, $A \in \mathbb{R}^{nXn}$ is the state transition or system matrix, $B \in \mathbb{R}^{nXm}$ is a matrix that relates the input $u(k) \in \mathbb{R}^m$ to the state, $C \in \mathbb{R}^{lXn}$ is the matrix that relates the output $y(k) \in \mathbb{R}^l$ to the state and $D \in \mathbb{R}^{lXm}$ is the matrix that relates the outputs to the inputs (Katayama, 2005), (Ljung, 1999).

2.2 Subspace Methods for System Identification

Multivariable discrete system identification using subspace methods allows finding a causal time-invariant model (1). The model is estimated just from system outputs and inputs. The main advantage of this approach is that a MIMO system can be modeled even if the system order is unknown a priori. There are well known subspace methods to address the time-invariant MIMO system identification such as MOESP and N4SID. The MOESP method (Verhaegen and Dewilde, 1992) is based on the LQ decomposition of Hankel matrix formed from input-output data, where L is lower triangular and Q is orthogonal. A Singular Value Decomposition (SVD) can be performed on a block from the L matrix to obtain the system order and the extended observability matrix. From this matrix it is possible to estimate the matrices C and A of model (1). The final step is to solve overdetermined linear equations using the least-squares method to estimate matrices B and D, for more details see (Giesbrecht and Bottura, 2015).

Another method is N4SID (Numerical Algorithms for Subspace State Space System Identification) (Overschee and Moor., 1994). The method stars with the oblique projection of the future outputs to past inputs and outputs into the future inputs direction. The second step is to apply the LQ decomposition and then the state vector can be computed by the SVD. Finally, it is possible to compute the matrices A,B,C and D by using the least-squares method.

2.3 Time-varying Systems

Most of the systems in real environments are nonlinear and time-variant, so the complexity of the system identification problem increases. Time-varying systems can be modeled in state-space as shown in (2), where the matrices A,B,C,D depend on k, that represents the time.

$$\begin{cases} x(k+1) = A(k)x(k) + B(k)u(k) \\ y(k) = C(k)x(k) + D(k)u(k) \end{cases} \tag{2}$$

There is a MOESP type method called MOESP-VAR. This method was develop in (Tamariz et al., 2005). Given a slowly time-varying system, the algorithm starts splitting up the output-input data into intervals defined as windows. Then, the MOESP algorithm is applied to each window and the matrices A,B,C,D are estimated. If the system presents a fast variation, the MOESP-VAR algorithm has poor accuracy. Another disadvantage is that the matrix decompositions used in the subspace methods is limited by the size of window data, in other words a greater amount of data is needed in order to make new methods are computationally feasible. For this case, techniques based on EAs can be used as an alternative, as proposed in (Giesbrecht and Bottura, 2015).

3 SYSTEM IDENTIFICATION PROBLEM AS A OPTIMIZATION PROBLEM

The identification of time variant systems can be defined as an optimization problem if a space of state space models is defined. The space contains all possible quadruples of matrices, the dimensions of matrices being known a priori. Each model in the state space is seen as a point in this space, also called search space. The main idea is to find the point that minimizes the error between the model and system outputs generated from the same input. This point also changes over time, that is, another quadrupling of matrices must be estimated.

EAs can adapt well to this optimization problem. For larger system variations over time, the algorithm should look for a new, four-point point that minimizes the estimation error. Based on this principle the COEAs were developed in this paper. In an illustrative way, the optimization problem can be defined as follows:

Exists $X_i \in \mathbb{R}^{(n+p)X(n+m)}$:

$$X_i = \begin{bmatrix} A_i & B_i \\ C_i & D_i \end{bmatrix} \tag{3}$$

The estimation error $E(X_i)$ is calculated in equation 4 and the fitness function $f(X_i)$ is defined in equation 5. Where $y(k)$ is the system output, $\hat{y}(k)$ is the model output (estimated by evolutionary algorithm) and N is the data size.

$$E(X_i) = \sum_{k=1}^{N} ||(y(k) - \hat{y}(k))|| \tag{4}$$

$$f(X_i) = \frac{1}{1 + E(X_i)} \tag{5}$$

There is a x_i^* which minimizes error, this concept taken to the fitness function, means to maximize the fitness function and is defined: $x_i^* \rightarrow max(f(x_i^*))$. The equation 5, can be modified or changed, this is another advantage of the EAs, which does not restrict the ways of creating a new fitness function.

4 COEVOLUTIONARY ALGORITHMS

Coevolutionary algorithms (COEAs) are heuristic methods based on cooperation and competition. Moreover, coevolution is strongly associated with a mutual evolution between two species, so that one exerts selective evolutionary pressure on the other, thus creating a paradigm: *there is no isolated evolution*. More detailed information regarding this subject can be found in the reference (Wiegand, 2003).

There are three categories that might benefit from coevolution: problems with large (infinite) Cartesian-product spaces, problems with no intrinsic objective measure, and problems with complex structures. The intent of coevolution is to produce a typical dynamic refereed as an *arms race*. The idea is that continued small adaptations in some individuals (or populations) will force competitive adaptations in others. Consider the predator-prey systems: if the prey evolves to run faster; then the predator population is forced to develop better attacking strategies (such as powerful claws). Therefore, the expectation is that, as the processes evolves, both populations will present exceptional attributes (Wiegand, 2003).

Another concept, used in this work, is the cooperative. In a context in which a large problem is very complex to solve, it may be possible to divide the problem into subcomponents and to solve each one of them independently. The subcomponents are also defined as subpopulations and the only interaction between subpopulations is in the cooperative evaluation of each individual of this subpopulations. This is done by concatenating the current individual with the best individuals from the rest of the other subpopulations, as described by (Potter and Jong, 2000).

Indeed, the advantages of the coevolution led this kind of technique to be applied in a large variety of complex problems. Optimization applications include the coevolution of fast, complete sorting networks (Hillis, 1991) and the coevolution of maximal arguments for complex functions (Potter and Jong, 1994). Machine learning applications include COEAs to search for useful game-playing strategies (Rosin and Belew, 1996).

5 COEVOLUTIONARY ALGORITHM TO SYSTEM IDENTIFICATION

Based on the principles of cooperative coevolution, in this work a COEA was develop to identify MTVS. To initialize the algorithm four populations were created: P_A, P_B, P_C and P_D, that are related to the matrices A, B, C and D from the state-space model (2), respectively. Notice that these matrices are not constant.

The individual objective of the populations is to find the configuration of their associated state matrices, that minimizes the estimation error between the model outputs $\hat{Y}$ and the actual system outputs Y to the same given input set U. The search space for each state-space matrix (2) is $M_A \in \mathbb{R}^{nXn}$,$M_B \in \mathbb{R}^{nXm}$,$M_C \in \mathbb{R}^{lXn}$, $M_D \in \mathbb{R}^{lXm}$. The populations belong to those search spaces.

In the following subsections the adopted coding, operators, the fitness function and the structure of algorithm are introduced.

5.1 Adopted Coding

The candidate solutions of each population is represented as follows: P_A has nXn elements, P_B has nXm elements, P_C has lXn elements P_D has lXm elements. The columns of each state-space matrix are transformed to rows and concatenated. This coding is shown for P_A in the table 1. The other matrices follow the same coding.

Table 1: Configuration of P_A chromosome.

	$a_{1\to n,1}$	$a_{1\to n,2}$	$\cdots$	$a_{1\to n,n}$
P_A	$(a_{1,1}\to a_{n,1})$	$(a_{1,2}\to a_{n,2})$	$\cdots$	$(a_{1,n}\to a_{n,n})$

5.2 Operators

The four populations P_A, P_B, P_C and P_D used the same operators, which are shown in the table 2.

Table 2: Operators.

Recombination	Arithmetic
Mutation	Gaussian
Selection	Tournament

In the mutation, the value of the variance depends on the number of generations, see 3. This approach was inspired by the simulated annealing (Kirkpatrick et al., 1983), where it is desired that individuals have greater freedom to explore the search space during the first generations. As the generations (g) pass, the freedom of exploration should be reduced, to encourage individuals to refine the candidate solutions previously encountered. In the original simulated annealing, the decay of this exploration factor (called temperature T) acts on the selection of individuals, whereas in this work it acts on the mutation of individuals, generating a reduction in the intensity of mutations over time.

The reduced value of individuals for tournament selection, generates low selective pressure in the population, as a consequence, allows low fitness individuals have a high chance to survive.

5.3 Fitness Function

The fitness function is defined as follows:

$$f(p_{(A,i)}) = \frac{1}{1+E(p_{(A,i)},p_t(B,r_1),p_{(C,r_2)},p_{(D,r_3)})}, \qquad (6)$$

Where individual $p_{(x,i)}$ of population x, in last equation (6) corresponds to P_A. Also, r_1, r_2 and r_3 are random individuals extracted from the others populations. The function E represents the total error between the model outputs $\hat{y}(k)$ estimated from these individuals and the system outputs $y(k)$, where n is the total number of data samples set to be considered. This function is defined as follows:

$$E(p_{(A,i)},p_{(B,r_1)},p_{(C,r_2)},p_{(D,r_3)}) = \sum_{k=1}^{n} ||(y(k)-\hat{y}(k)|| \qquad (7)$$

5.4 Structure

The first step is to initialize randomly the individuals from each population. The number of candidate solutions that are selected for mutation and recombination has deterministic behavior, in which 75% of the population is selected for the crossover, while 25% of the individuals undergo mutation. Then, the fitness of the individuals is calculated. Finally, the step of selection is made. All steps are shown in Figure 1.

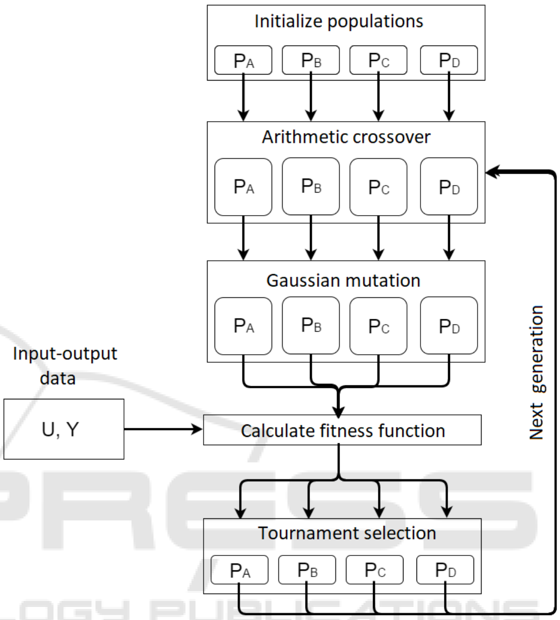

Figure 1: Algorithms steps.

6 RESULTS

In the following example the algorithm proposed in this paper was applied to identify a slowly time-varying system. A bi-dimensional white noise with $N = 1000$ samples was given as input. The parameters of the algorithm were configured as shown in the Table 3. The benchmark is modeled by the following model taken from (Tamariz et al., 2005):

$$A_k = \begin{bmatrix} a(k) & b(k) \\ 1 & -1 \end{bmatrix} \qquad (8)$$

where

$$a(k) = -\frac{1}{2}(k/2500)^2 + \frac{1}{2}(k/2500)$$

$$\begin{aligned} b(k) \quad = \quad & -\frac{1}{16}(k/2500)^4 - \frac{1}{8}(k/2500)^3 - \frac{13}{16}(k/2500)^2 \\ & -\frac{3}{4}(k/2500) - \frac{1}{2} \end{aligned}$$

The others matrix are invariant

$$B_k = \begin{bmatrix} -2 & 1 \\ 1 & 1 \end{bmatrix}; \qquad C_k = \begin{bmatrix} 1 & 3 \\ 1 & 2 \end{bmatrix}$$

$$D_k = \begin{bmatrix} 1 & 3 \\ 1 & 1 \end{bmatrix} \tag{9}$$

The time variation of the parameters $a(k)$ and $b(k)$ are shown in Figure 2.

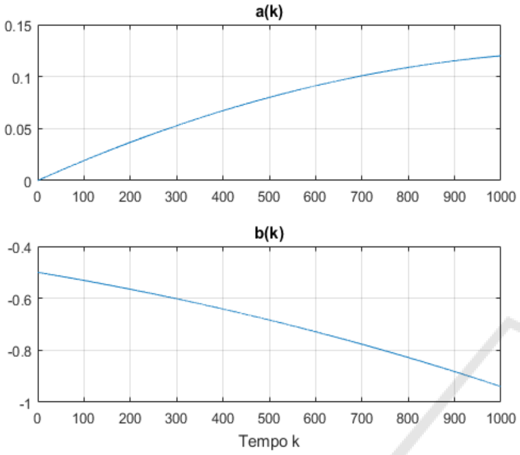

Figure 2: Time variations in the parameters of A_k.

Table 3: Parameters configuration.

Parameters	
Number of populations	4
Generations (g)	4000
Size of population	200
Individuals to crossover	150
Selection probability (α)	0.6
Individuals to mutation	50
Variance (σ)	$1-\frac{g_i}{2*g}$
Individuals to tournament (q)	2

In the process of identifying systems using coevolution, diversity is a key factor to ensure the evolution progress. Often the increase in fitness is not associated to the best fitness individual. For instance, the chromosome with the highest fitness population P_A will not necessarily get better fitness when interacting with the best individual of the population P_B. Increased fitness can happen through interactions between a high fitness individual and another from below or even between two low fitness individuals. Thus, ensuring the population diversity through regular mutations is crucial to its evolution.

Interdependence among populations is a limiting factor affecting the evolutionary progress of a population. For example, if a population has already found the optimal solution within their search space, it will not help if the other populations do not evolve together. For this reason, the progress of a population usually triggers the progress of other populations. This can be seen in Figure 3, in which the fitness evolution of the four populations along the iterations is presented. From the figure it is possible to perceive that the interdependence of the populations occurs during the generations of 2000 to 3400, where the advance of a population, that affects the others happens. The number of generations was determined by empirical analysis. It has been realized that the populations need at least 4000 generations to obtain relevant fitness values.

To evaluate the proposed method, the data generated by white noise ($N = 1000$) was divided into 5 sets: $(U_1,Y_1),(U_2,Y_2),(U_3,Y_3),(U_4,Y_4)$ and (U_5,Y_5). In those sets, the system variation is higher for the higher index data. For instance, the variation for (U_2,Y_2) is greater than those of (U_1,Y_1) and less than (U_3,Y_3). In each data set, three additional methods were applied: MOESP-VAR, GA and RS. To ensure a fair evaluation, the number of iterations (generations) made for each method, except for MOESP-VAR, was the same. In addition, GA uses the same operators and parameters applied to COEA.

The results are presented in the Table 4. It is possible to infer that MOESP-VAR and COEA have similar performance, but in most cases the coevolution has a slight advantage. This may be a promising result which indicates that more elaborate refinement may lead to coevolution to find the solution even better for time-varying system identification problem.

Both GA and RS have fitness ten times lower than MOESP-VAR and COEA, this occurs because GA and RS try to find the best configuration of the subspaces at once, that is, they work with the evolution of individuals with dimension 16. On the other hand, coevolution uses the idea of "divide for conquer ", where different populations try to find the sub-solutions that interact better with the other individuals of other populations.

7 CONCLUSIONS

In this paper the steps to identify time-varying systems with COEA are defined. The results have demonstrated that this type of algorithm has better performance (fitness) than deterministic method MOESP-VAR in the majority of the examples studied. Moreover, this approach estimates models of time-varying systems, which represent the system in a window or time interval. In addition, the coevolution allows to parallelize the evolutionary process, which can make it competitive in terms of execution time.

Table 4: Fitness value of the best individual for the data sets used in each method.

	Fitness $(x10^{-3})$				
	(U_1,Y_1)	(U_2,Y_2)	(U_3,Y_3)	(U_4,Y_4)	(U_5,Y_5)
COEA	10.1	6.2	5.4	5.1	2.2
MOESP-VAR	8.23	6.51	5.33	4.27	2.13
GA	0.97	0.78	0.85	0.83	0.67
RS	0.82	0.65	0.81	0.61	0.52

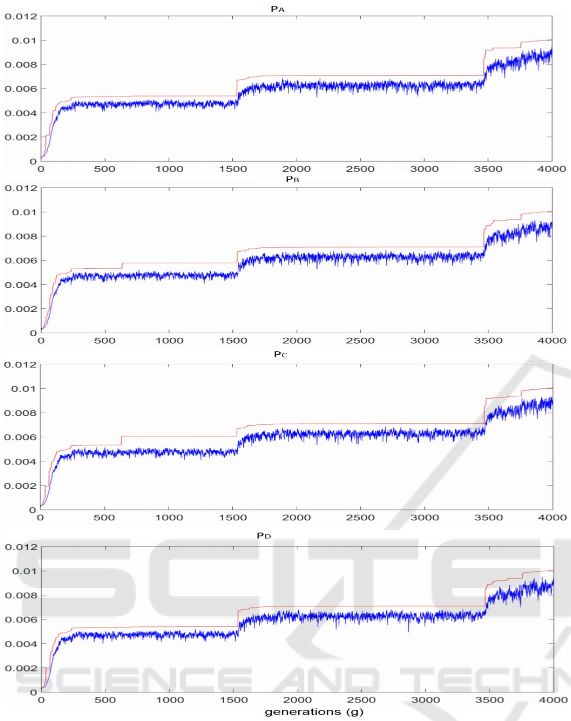

Figure 3: Fitness of the four populations for (U_1,Y_1) set . The red curve represents the fitness of the best individual (best fitness) and the blue curve is the mean fitness of the population.

The main objective was to find the matrices $\hat{A},\hat{B},\hat{C},\hat{D}$ (state-space model), that minimize the estimation error by solving an optimization problem. A new approach for future work would be to develop a recursive coevolution method, to estimate the state-space model at each instant of time. Other evolutionary techniques can be implemented to find the best alternative to solve the defined optimization problems. Finally, the proposed method can be applied to model time series and compare results with immuno-inspired algorithm (Giesbrecht and Bottura, 2015).

REFERENCES

De Castro, L. N. and Von Zuben, F. J. (2002). Learning and optimization using the clonal selection principle.

Giesbrecht, M. and Bottura, C. P. (2015). *Recursive immuno-inspired algorithm for time variant discrete multivariable dynamic system state space identification*. International Journal of Natural Computing Research,5(2), April-June 69-100.

Hillis, D. (1991). *Co-evolving parasites improve simulated evolution as an optimization procedure*. Artificial Life II, SFI Studies in the Sciences of Complexity 10, 313324.

Katayama, T. (2005). *Subspace Methods for System Identification*. Springer.

Kirkpatrick, S., Gerlatt, C. D. J., and Vecchi, M. P. (1983). *Optimization by Simulated Annealing*. Science.

Ljung, L. (1999). *System Identification-Theory for the User.* Prentice-Hall, Upper Saddle River, N.J., 2nd edition edition.

Overschee, P. V. and Moor., B. D. (1994). *N4SID: Supspace Algorithms for the Identification of Combined Deterministic-Stochastic Systems*. Automatica, 30(1):75-93.

Potter, M. and Jong, K. D. (1994). *A cooperative coevolutionary approach to function optimization*. See Davidor and Schwefel, pp. 249257.

Potter, M. and Jong, K. D. (2000). Cooperative coevolution: an architecture for evolving coadapted subcomponents.

Rodrigues-Vasquez, K., F. C. and Fleming, P. (2004). Identifying the structure of nonlinear dynamic systems using multiobjective genetic programming.

Rosin, C. and Belew, R. (1996). *New methods for competitive coevolution*. Evolutionary Computation 5(1), 129.

Tamariz, A. D. R., Bottura, C. P., and Barreto, G. (2005). *Iterative MOESP Type Algorithm for Discrete Time Variant System Identification*. Proceedings of the 13th Mediterranean Conference on Control and Automation.

Verhaegen, M. and Dewilde, P. (1992). *Subspace model identification - part 1 : The output-error state-space model identification class of algorithms*. International Journal of Control 56, 5, 11871210.

Wakizono, M., H. T. and Uosaki, K. (2006). Time varying system identification immune based evolutionary computation.

Wiegand, R. P. (2003). *An analysis of cooperative coevolutionary algorithms*. Ph.D. Dissertation Department of Computer Science, George Mason University, Fairfax, VA, USA.