Design of Cascaded Controllers for DC Drives using Evolutionary Algorithms

Francesco Cupertino, David Naso, Luigi Salvatore, and Biagio Turchiano

Dipartimento di Elettrotecnica ed Elettronica
Politecnico di Bari
Via Re David, 200 – 70125 Bari (IT)
E-mail: {cupertino, naso, salvatore, turchiano}@poliba.it
Fax: (+39) 080 5963 410

Abstract: **In this paper, we describe a new design procedure for cascaded controller for electrical drives based on evolutionary algorithms. Most electric drives have two separate controllers for current and speed control, which are in general designed in two consecutive steps (firstly the current controller and then the speed one). Using a hybrid evolutionary algorithm designed to test and compare controllers of different orders, we search simultaneously for the couple of discrete anti-windup controllers achieving the optimal compromise of cost and performance indices related to both current and speed responses. Simulation comparison with conventional design techniques on a non-linear DC drive with variable load are presented to show the effectiveness of the approach.**

I. INTRODUCTION

Genetic and Evolutionary Algorithms are stochastic search techniques that have been successfully applied to various control engineering problems, including sliding mode [14], adaptive [5, 11], optimal [13], and robust control [10-12].

Recently, Genetic Algorithms (GAs) have been successfully applied also to electric drives control [9]. Namely, in [6] a "fuzzy-like" Luenberger observer is used to estimate the stator flux and the speed of an induction motor. A GA determines the best pole placement for the observer minimising a cost function related to the time response of the error between estimated and actual state variables. In [15], a GA is used for the optimal design of a fuzzy controller. The GA chooses number and position of the membership functions and modifies the rule table. The method does not require for professional expertise or mathematical analysis of the plant model. Unfortunately fuzzy controllers are still difficult to implement on commercial low-cost micro-controller, and may not offer significant improvements in terms of robustness and performances: the use of classical PID controllers is preferred whenever non-linear techniques are not strictly required. In [1], the authors utilise a GA-based algorithm to tune the speed controller of a brushless DC drive. In their work the authors do not consider the tuning of the inner current control loop. Moreover large and small speed steps are separately considered leading to two different speed regulator designs that are only locally optimal.

In general, optimal control problems in the field of electric drives require finding a trade off between conflicting design objectives. This task is frequently prohibitive in industrial applications where offline modeling and design techniques capable of accounting the actual nonlinearities of the system (e.g. saturation of the current controller, drive frictions) are either too complex or involve oversimplification [1]). On the contrary, the availability of a detailed nonlinear model of the controlled drive allows the computation of any index combining cost and performance criteria through simulation. In fact, simulation well lends itself to measure the 'fitness' of a control solution in a genetic optimization process, where alternative design solutions are iteratively created, compared and selected by the GA. In this respect, the potentialities of evolutionary computation are often not fully exploited, since the preponderance of technical literature uses single indices as integral time errors in system's step response.

This paper uses a GA to optimize both the structure and the associated parameters of two cascaded controllers for a DC electric drive. The approach has two main peculiarities with respect to existing literature. Firstly, in contrast with the conventional GAs that work on populations of homogeneous structures (i.e. controllers having the same number of zeros and poles), the evolutionary algorithm hereby proposed has been designed to test and compare simultaneously controllers having different orders. In other words, the GA must find both the numbers and the locations of controller's zeros and poles providing the maximum fitness. Secondly, while conventional design strategies firstly tune the current controller and then the speed controller (thus leading to a potentially sub-optimal couple of controllers), our algorithm simultaneously optimizes the two controllers, to obtain the best overall two-cascaded-loops control systems. Simulation results show the effectiveness of the proposed approach for the controller design of nonlinear drive systems. A comparison with traditional linear design method is also given.

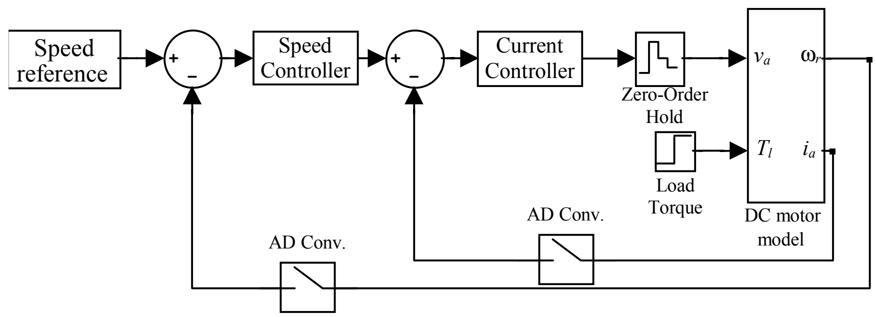

FIGURE1. DC DRIVE BLOCK DIAGRAM.

TABLE I - DC MOTOR PARAMETERS

Rated current	i_{an} = 1000 A
Rated voltage	V_n = 220 V
Rated speed	ω_n = 50 rad/s
Torque constant	$k\phi$ = 3.8 Nm/A
Rotor inertia	J = 100 kg m^2
Armature resistance	R_a= 0.03 Ω
Armature inductance	L_a=0.0006 H

II. DC DRIVE CONTROLLERS DESIGN.

The most common linear approximation of DC motor current and voltage dynamics can be summarized as follows:

$$v_a = R_a i_a + L_a \frac{di_a}{dt} + k\Phi\omega \tag{1}$$

$$T_e = k\phi i_a \tag{2}$$

where v_a, i_a, R_a, and L_a are the armature voltage, current, resistance and inductance respectively, $k\Phi$ is the torque constant, ω the rotor speed and T_e the electromagnetic torque developed by the motor. DC motors are usually controlled using a cascaded scheme with an inner current PI controller and an outer speed PI controller, as shown in figure 3. This control scheme is also employed in high performance AC drives such as vector controlled asynchronous and brushless drives. Therefore, the design procedure proposed in this paper can be extended to these cases without significant modifications.

Traditionally PI-type regulators are employed in both control loops due to their high performance/simplicity trade-off. Very often, proportional and integral gains are chosen using linear analysis methods. The non-linearities and delays introduced by the power converter, sensors, stiction, current and voltage limits and the load characteristic are not considered. Therefore, an on-line fine-tuning procedure using trial and error methods is necessary. Manual tuning requires experienced operators, is time-consuming, and does not guarantee to find the optimal controllers. Moreover PI regulators are suitable to control second order linear systems but their performances may deteriorate with higher-order non-linear systems such as electric drives. Higher-order controllers can offer considerable enhancements in this case but the tuning difficulties may become prohibitive.

This paper formulates the global design of a two loop discrete control system for electric drives as a search problem. As in optimal control design problems, we aim to find the couple of controllers (including orders) achieving the maximum satisfaction of a cost-to-performance merit function. Typically, optimal control theory for linear systems requires to optimize the sum of quadratic norms of the (actuation) cost and output error with respect to set-point. In this paper, to quantify the effectiveness of the couple of controllers, we extend the conventional objective function to take into account further desirable characteristics of the current and speed responses.

The reference input for control loop simulation also plays a fundamental role in the success of the evolutionary design. Controllers for DC drives must provide satisfactory responses when operating within the linearity ranges of the actuators, and at the same time they must limit the effects of saturations in presence of abrupt set-point changes. These aspects are often contrasting, i.e. good controllers in linear conditions have poor performance in saturation and vice-versa. In order to take both aspects into account in our problems, the system is excited with a square wave of variable amplitude (first small changes to operate in linear conditions and then higher changes to induce saturation). To optimize disturbance rejection, we also apply a change in load torque during steady state conditions.

Our objective function is a weighted sum of several performance indices that are directly measured on system's response to the input signal described above. Namely, we define the following fitness (to be minimized):

$$f = \sum_{j=1}^{6} \alpha_j \cdot f_j \tag{3}$$

where α_j represent positive weights, and f_j are six performance indices defined as follows. Weights α_j allow the user to emphasize or reduce the contribution of each single performance index in the final value of the fitness. In this paper, the α_j are set heuristically, i.e. performing preliminary runs of the GA with changed weights until the desired trade-off between indices is achieved. Symbols ω and ω_{ref} indicate motor and reference (i.e. setpoint) speed, respectively. Analogously, i and i_{ref} indicate the actual current and its reference value provided by the speed controller. All the signals are normalized with respect to their nominal value. The integer n indicates the number of samples in the simulated interval.

0-7803-7282-4/02/$10.00 ©2002 IEEE

n_{zi}	n_{pi}	$n_{z\omega}$	$n_{p\omega}$	K_i	z_{0i}	z_{1i}	p_{0i}	p_{1i}	p_{2i}	…	K_ω	$z_{0\omega}$	$z_{1\omega}$	$p_{0\omega}$	$p_{1\omega}$	$p_{2\omega}$	…
←Structure chromosomes→				←Parameters of current control→							←Parameters of speed controller→						

FIGURE 2: STRUCTURE OF THE GENERIC CHROMOSOME

1. Steady state speed response

$$f_1 = \sum_{j=1}^{n} \left|\omega(j) - \omega_{ref}(j)\right| \cdot g(j) \tag{4}$$

where

$$g(j) = \begin{cases} 1 \text{ if } \left|\omega(j) - \omega_{ref}(j)\right| \le \sigma_0 \\ 0 \quad \text{otherwise} \end{cases} \tag{5}$$

This index measures the speed absolute error only along segments of the speed response settling around the steady state. Parameter σ_0 defines the settling band.

2. Disturbance rejection

$$f_2 = \left|\omega(n) - \omega_{ref}(n)\right| \tag{6}$$

This index measures the absolute error in the last time sample. It is used to evaluate the ability to reject the change in load torque, which is applied in the final part of the simulation.

3. Speed transient response duration

$$f_3 = \frac{n - n_t}{n} \cdot l(n_t) \tag{7}$$

where

$$n_t = \sum_{j=1}^{n} g(j)\,;\; l(n_t) = \begin{cases} 0 \text{ if } n_t \ge \rho \cdot n \\ 1 \quad \text{otherwise} \end{cases} \tag{8}$$

This index measures the duration of transient condition, i.e. estimates the sum of all settling times. The parameter ρ defines the desired steady-state time of speed response in per-unit.

4. Current response

$$f_4 = \sum_{j=1}^{n} \left| i(j) - i_{ref}(j) \right| \tag{9}$$

This index measures the sum of absolute current errors.

5. Current overshoot

$$f_5 = (i_{max} - i_s) \cdot m(i_{max}) \tag{10}$$

with $i_{max} = \max\limits_{j=1\ldots n} \left|i(j)\right|$, and $m(i_{max}) = \begin{cases} 1 \text{ if } i_{max} > i_s \\ 0 \text{ otherwise} \end{cases}$ (11)

This index measures the highest current peak i_{max} exceeding the maximum current supply i_s.

6. Current oscillations for null speed reference

$$f_6 = \frac{n_{+/-}}{n} \tag{12}$$

This index accounts for small ripples and oscillations of the current induced by typical dead band nonlinearities when the motor speed tends to zero. The integer $n_{+/-}$ counts the number of change of sign in current response.

III. GAs FOR STRUCTURE AND PARAMETER OPTIMIZATION

The performance of any GA in terms of speed of convergence, reliability of the search and accuracy of the final solution is strictly related to the selection and recombination strategies. In our research, crossover and mutation operators have been redefined to deal with variable order controllers, and their optimal occurrence rate has been modified to ensure a simultaneous convergence of controller structure and parameters. Due to the ability to work on heterogeneous solutions, the proposed GA shares interesting similarities with another large class of evolutionary algorithms known as Genetic Programs (GP). In a recent research [7-8], GP have been used to automatically discover both the topology and the parameters of robust controllers for simulated plants. In contrast with [7-8], in this paper the topology of the control system is pre-selected in the standard form, having the two feedback loops with cascade current and speed controllers described in Figure 1, while the search algorithm focuses on the optimization of the discrete transfer functions.

Figure 2 describes the encoded structure of a generic solution. A first part of the chromosome contains the integer parameters describing the orders of the polynomials of the two controllers, and the remaining part contains the transfer functions gains and the positions of their zeros and poles. Both controllers are described in discrete-time domain using z-transform. To obtain minimum phase controllers, all the zeros and poles must lay within the unit circle. Poles and zeros are encoded in the chromosomes using the mapping strategy described in [2], which ensures a correct manipulation of conjugate roots. Furthermore, both the controllers are implemented using anti-windup algorithm to improve the response of the system in presence of saturations of the actuators.

The crossover and mutation operators have been redesigned to deal with controllers of variable structure. The mutation of a solution consists in a random perturbation of one (or more) components of the solution (see figure 2). A mutation of the structure consists in the random elimination or addition of

0-7803-7282-4/02/$10.00 ©2002 IEEE

one or more random zeros and poles, located within the search bounds. The probability of adding or removing more roots is decreased with the number of roots, i.e. the elimination/addition of a single root has higher probability than two or more roots. The operations performed during the mutation phase of a solution are summarized in figure 3.

By analogy with standard operators, the crossover between two solutions having different orders should produce new solution inheriting part of the characteristics of both parents. Several known operators, including a crossover recently proposed for analogous problems [2], were compared in preliminary runs to set up and optimize the search algorithm. The highest success rates were obtained by letting only solutions with the same orders to cross with each other, thus obtaining new zeros and poles prevalently placed in intermediate positions between those of the two parents. Therefore, the GA proposed in this paper performs the standard crossover using three well-known variants (simple, arithmetical and heuristic), only when two homogeneous solutions are selected from the mating pool.

The ability to work on heterogeneous solutions makes the proposed algorithm similar to other evolutionary computation techniques, such as GP and Evolutionary Programming (EP). A significant difference between EP and GAs lays in the rate of application of the mutation. EP is explicitly based on mutation, which is the preponderant strategy to determine new solutions, while GAs are based on crossover, and the mutation is an auxiliary operator used to prevent inherent drawbacks as premature convergence or genetic drift. In the preliminary investigation for optimal occurrence rate of operators, a much faster convergence was obtained with higher mutation rates. Related research [3], suggests several intuitive justifications of these results. In fact, conventional crossover tends to perform similarly to random mutation when complex coding structures are employed. Furthermore, as the population size grows, smaller crossover probability are in general more effective [4]. In conclusion, due to our final choice of emphasizing the rate of mutation (80%) with respect to the crossover rate (20%), the proposed algorithm can be viewed as a hybrid GA-EP evolutionary algorithm.

IV. SIMULATION RESULTS

As term of comparison of the effectiveness of the proposed approach, we consider a cascaded loop of two PI controllers designed with optimal techniques for linear systems. The design of the PI controllers is briefly described in the following. The current control loop has been reduced to a first order system having time constant equal to

$$\tau_{ia} = 0.1\frac{L_a}{R_a} \tag{13}$$

by placing the current PI time constant

$$\tau_{ii} = \frac{JR_a}{(k\Phi)^2} \tag{14}$$

and its proportional gain

$$k_{pi} = \frac{L_a}{\tau_{ia}} - R_a \tag{15}$$

For the speed control loop the symmetric optimum criterion [9] has been used. Accordingly, the speed PI time constant and proportional gain have been selected as follows:

$$\tau_{i\omega} = 4\tau_{ia}\,,\; k_{p\omega} = \frac{J}{2k\Phi\tau_{ia}\left(1-\tau_{ia}R_a/L_a\right)}\,. \tag{16}$$

The maximum order of both the controllers to be designed by the GA was chosen equal to four. The bounds of the controller gains were chosen so to ensure stable behavior. An initial population of 100 random individuals is then evolved throughout 200 generations. The GA was launched 40 times with different initial population to analyze the reliability of final results.

The simulation model is a detailed replica of a laboratory DC drive featuring both actuator saturation and a dead-zone with an amplitude amounting to 5% of the rated voltage. Figure 4-7 compare the results obtained with the cascaded PI controllers and the GA-designed double loop. In all figures, the part (a) shows the response of the double PI loop, while part (b) shows the response of the GA-designed (GA-DES) system.

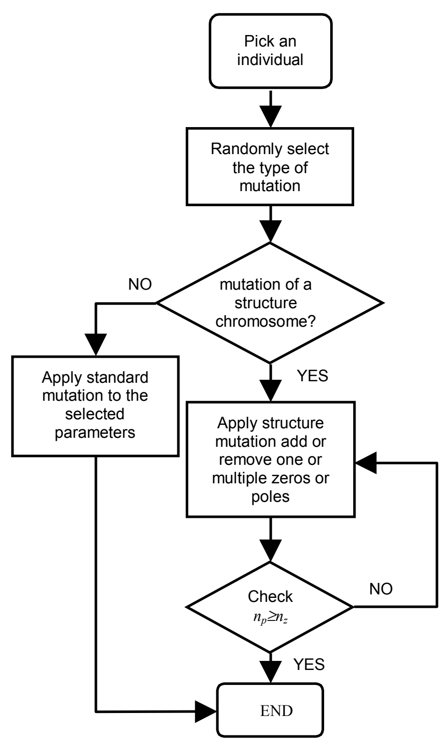

FIGURE 3: THE PROPOSED HYBRID MUTATION

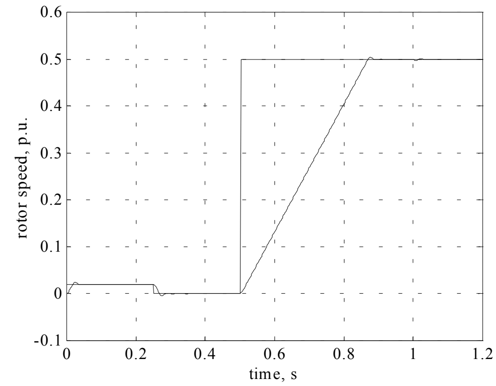

FIGURE 4(A): SPEED OUTPUT VS SET-POINT- DOUBLE PI

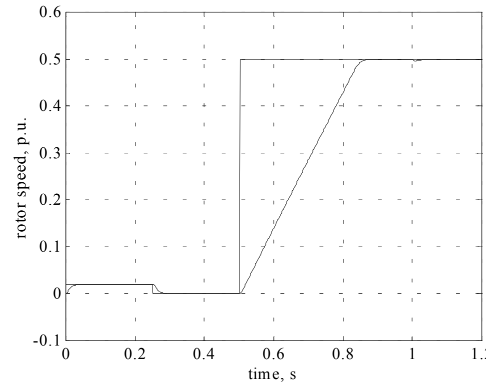

FIGURE 4(B): SPEED OUTPUT VS SET-POINT- GA-DES. CONTROL

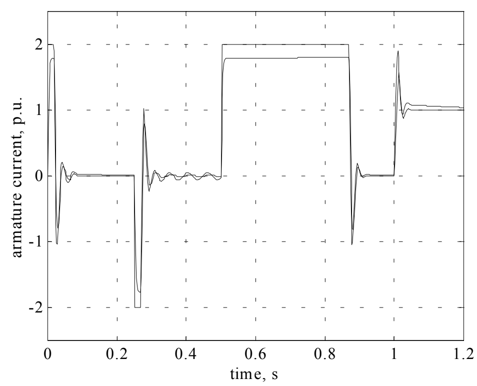

FIGURE 5(A): CURRENT OUTPUT VS SET-POINT- DOUBLE PI

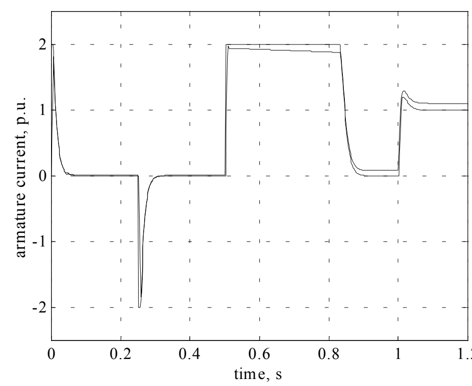

FIGURE 5(B): CURRENT OUTPUT VS SET-POINT- GA-DES. CONTROL

All figures show both the reference set-point signal (either the speed or the current) and the corresponding output. Due to the cascaded double loop structure, the current set-point is the output of the speed controller.

In particular, figures 4(a) and 5(a) show that using two PI regulators, the motor speed has still a satisfying behavior, but the current response in the low-speed region suffers from excessive oscillation (ranging from t=0.3s to t=0.5s, see the expanded figure 7(a)). No particular tuning expedient appears feasible in this case, since slower current PI regulators introduce higher current errors during speed transients and, consequently, lower dynamic performance. On the other hand, faster current PI regulators introduce unacceptable current and voltage oscillations.

The best control loop obtained by the GA encompasses a first order controller for the current and a second order controller for the speed. For sake of completeness, the current and speed controllers have the following zeros, poles and gain, respectively: z_{i1}=-0.3509, p_{i1}= 0.7181, k_i=0.5684; $z_{\omega1}$=-0.0328, $z_{\omega2}$= 0.8987, $p_{\omega1}$= 0.2452, $p_{\omega2}$= 0.9967, k_ω=3785. Figures 4(b) and 5(b) show that the GA-designed double loop enhances both the speed and the current response. The differences between the two design strategies are emphasized in figure 6-7, tracing closer views of the speed response to the load torque (applied at time t=1s) and the current along the time interval with null speed set-point. Figure 6 shows that the GA-designed loop avoids the overshoot but suffers from a residual steady state error due to the small offset of the pole $p_{\omega2}$=0.9967 with respect to the pure integrator z=1. The current response is also significantly improved, since there are no current oscillations in the low-speed region (figure 7) in spite of the presence of the dead-band. This behavior is not obtainable with a lower order speed controller.

CONCLUSIONS

In this paper, we used an hybrid Evolutionary Algorithm to search for the optimal cascaded two loop control system for non-linear DC electrical drives. Both control loops are based on discrete linear controllers with anti-windup algorithms. In digital micro-controllers the increased order of a transfer function does not significantly increase the computational burden, since only a few more samples of the controlled variables have to be stored in the processor memory Thus the algorithm was devised to search for the structure and the associated parameters. Encouraging results have been obtained in simulation. Using a slightly higher order for the speed controller, the GA is able to design a cascaded loop with an optimal trade-off of merit figures. Namely, the effects of nonlinearities as saturations and dead-zone, which degrade the results of linear design techniques, are fully compensated by the optimized combination of controllers. The proposed design technique well lends itself to be directly applied to real drives, rather than to simulation models. By applying GAs to real drives, the a priori knowledge of the detailed model is not indispensable, while it was fundamental in this paper both to define the simulation model and to design the PI regulators for benchmarking. In principle, the genetic design of higher order controllers can be more profitable in

0-7803-7282-4/02/$10.00 ©2002 IEEE

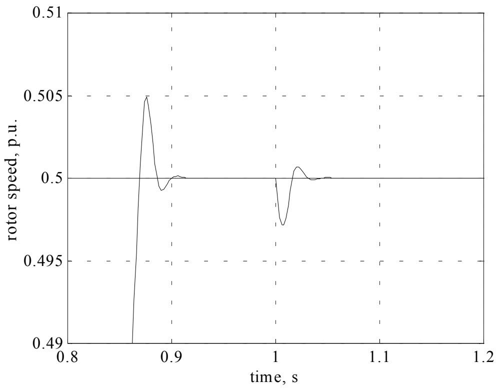

FIGURE 6(A): SPEED OUTPUT VS SET-POINT- DOUBLE PI

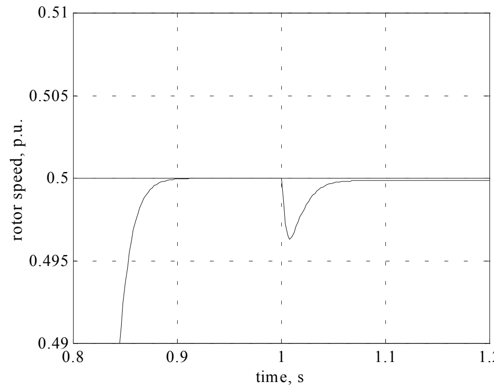

FIGURE 6(B): SPEED OUTPUT VS SET-POINT- GA-DES. CONTROL

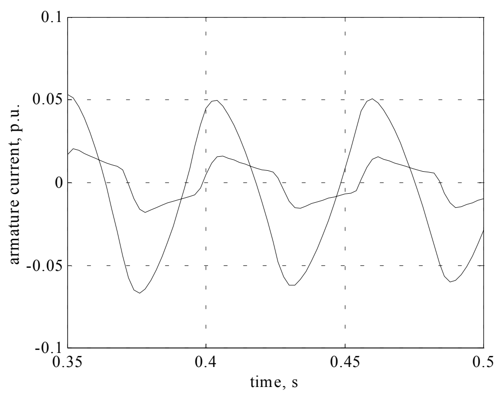

FIGURE 7(A): SPEED OUTPUT VS SET-POINT- DOUBLE PI

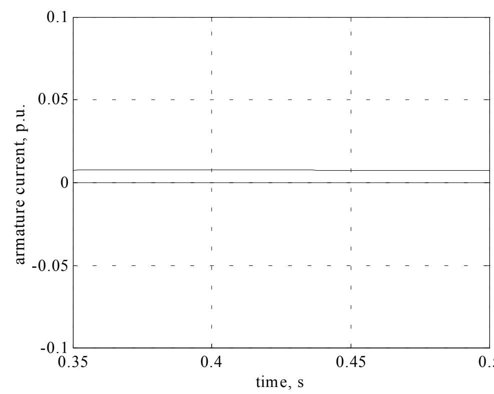

FIGURE 7(B): SPEED OUTPUT VS SET-POINT- GA-DES. CONTROL

the case of direct application to drives, whereas actual and unknown a priori high order terms and nonlinearities can be fully compensated. Also the set up of the weighted aggregation of performance indices was a relatively easy task in the considered case, but this cannot be guaranteed in general. Therefore, a promising enhancement of the proposed strategy is the use of a truly multi-objective evolutionary approach with pareto-set optimization instead than an aggregating approach. These aspects, along with the extension of the proposed approach to motors with different control structures, as asynchronous and brushless drives, are the focus of the current research.

REFERENCES

[1] W.G. da Silva, P.P. Acarnley, and J.W. Finch: "Application of Genetic Algorithms to the Online Tuning of Electric Drive Speed Controllers", IEEE Transactions on Industrial Electronics, Vol.47, no.1, February 2000, pp217-219.

[2] M. Dotoli, G. Maione, D. Naso, e B. Turchiano, "Genetic Identification of Dynamical Systems with Static Nonlinearities", IEEE SMCia/01,Mountain Workshop on Soft Computing in Industrial Applications, Virginia Tech, Blacksburg, Virginia, June 25–27, 2001.

[3] D. B. Fogel, System Identification Through Simulated Evolution – A Machine Learning Approach to Modeling, Ginn Press, Needham Heights, MA, 1991.

[4] J. J. Grefenstette. "Optimization of Control Parameters for Genetic Algorithms", IEEE Trans. on Sys, Man, and Cyb., Vol. 16, no.1,pp. 122-128, 1986.

[5] J. R. Gremling and K. M. Passino, "Genetic Adaptive State Estimation for a Jet Engine Compressor", Proc. Of the 12th IEEE Int. sym on Intelligent Control, Istanbul, Turkey, pp. 131-136, 16-18 July 1997.

[6] G. Griva, F. Profumo, L. Rossel, R. Bojoi: "Optimization of Fuzzy-Like Luenberger Observer for High Speed Sensorless Induction Motor Drives Using Genetic Algorithms", Proceedings of IAS 2000.

[7] Koza, J.R., Keane, M.A., Yu, J.; Bennett, F.H., III., Mydlowec, W., Stiffelman, O., Automatic synthesis of both the topology and parameters for a robust controller for a nonminimal phase plant and a three-lag plant by means of genetic programming, Proceedings of the 38th IEEE Conference on Decision and Control, 1999 , Vol. 5 , pp. 5292 –5300.

[8] Koza, J.R., Keane, M.A., Yu, J., Mydlowec, W., Bennett, F.H., Automatic synthesis of both the control law and parameters for a controller for a three-lag plant with five-second delay using genetic programming and simulation techniques Proceedings of the 2000 American Control Conference, Vol.1, pp. 453 –459.

[9] Leonhard, W., Control of Electrical Drives. Springer-Verlag Berlin, 1985.

[10] C. H. Marrison, and R. F. Stengel, "Robust Control system Design Using Random Search and Genetic Algorithms", IEEE Trans. Automatic Control, vol. 42, no. 6, pp. 835-839, 1997.

[11] M. L. Moore, J. T . Musacchio, and K. M. Passino, "Genetic Adaptive Control for an Inverted Wedge", Proc. of the American Control Conference, San Diego, California, pp. 400-404, June 1999.

[12] M. Sami Fadali, Y. Zhang, and S. J Louis, "Robust Stability Analysis of Discrete-Time systems Using Genetic Algorithms", IEEE Trans. System, Man, and Cybernetics, Part A: System and Humans, vol. 29, no.5, pp. 503-508, 1999.

[13] Y.C. Sim, S. B. Leng, and V. Subramaniam, "A combined genetic algorithms-shooting method approach to solving optimal control problems", Int. J. of System Science, vol. 31, no. 1, pp.83-89, 2000.

[14] A. Trebi-Ollennu, and B. A. White, "Multiobjective fuzzy genetic algorithm optimisation approach to nonlinear control system design", IEE Proc.-Control Theory Appl. Vol. 144, No. 2, pp. 137-142, 1997.

[15] Y.S. Zhou, L.Y. Lai: "Optimal Design for Fuzzy Controllers by Genetic Algorithms", IEEE Transactions on Industry Applications, Vol.36, no.1, January/February 2000, pp.93-97.

0-7803-7282-4/02/$10.00 ©2002 IEEE