NEURO-PREDICTIVE CONTROL DESIGN BASED ON GENETIC ALGORITHMS

I.Sekaj, S.Kajan, L.Körösi, Z.Dideková, L.Mrafko

Institute of Control and Industrial Informatics
Faculty of Electrical Engineering and Information Technology
Slovak University of Technology in Bratislava, Slovak Republic

Abstract

Genetic algorithm based neuro-predictive controller design of a non-linear dynamic system is described. The genetic algorithm represents an optimisation procedure, where the cost function to be minimized comprises the closed-loop simulation of the control process and a selected performance index evaluation. Using this aproach the neural model of the process has been optimised from point of view its internal architecture. Next the parameters of the predictive controller were optimised in order to become the required behaviour of the control process.

1 Introduction

For control of some clases of non-linear dynamic systems with advantage neuro-predictive controllers are used. This control structure consists from model of the controlled non-linear system and a predictive controller, wich optimises the control value in the actual time period under consideration of the controlled system behaviour prediction. In the presented project the system model is realised using artificial neural network. The optimal performance of such control structure reqires optimal parameters of the predictive controller and of the neural model. This all requires experience and expert knowledge. Our aim was to propose an aproach, which is able to find the optimal (or suboptimal) control configuration of the neuro-predictive system. For that reason an evolution-based approach has been used. Evolutionary search/optimisation approaches are able to construct new control laws and non-intuitive solutions as well. One of the most frequently used evolutionary technique is the genetic algorithm (GA). Recently, genetic algorithms have been applied in the area of process control for solving a wide spectrum of various optimisation problems in several ways and with several aims. In the proposed project the GA’s are used for search for the optimal neural model architecture and for the predictive controller parameter optimisation.

2 Controller structure

The block scheme of the used controller structure is depicted in fig.1. Without loss of generality, in case of the controlled process model a MLP (multilayer perceptron) artificial neural network is considered. The number of layers is 3 and the number of neurons in the input and hidden layer and their interconnections are the objects of the design/optimisation process. As the learning rule the off-line version of the Eror-back-propagation method with Levemberg–Marquardt modification is considered. The predictive controller in each control period *k* updates the control value *u(k)* by minimising the cost function J

$$\Delta u(k) = ?; J \rightarrow \min$$

$$J = \sum_{i=N_1}^{N_2} [r(k+i) - \hat{y}(k+i)]^2 + \rho \sum_{i=1_1}^{Nu} [\Delta u(k+i-1)]^2$$

$$u(k) = u(k-1) + \Delta u(k) .$$

Here the r is the reference signal, $\hat{y}$ is the controlled value prediction, Δu is the control value change, k is the control step, N_1 is the lower and N_2 the upper output prediction horizon, N_u is control horizon and ρ is a weight constant. The control performance depends on the values N_1, N_2, N_u and ρ. The block scheme of the neural model is in fig.2. The number of inputs $y_{k-1}, y_{k-2},...,y_{k-n}, u_k, u_{k-1}, ..., u_{k-m}$ (number of past samples of y and u) and the number of neurons in the hidden layer are normally set by the designer using exprience. But often the chosen model architecture is not optimal because of the modelling accuracy and on the other hand from point of view computation time.

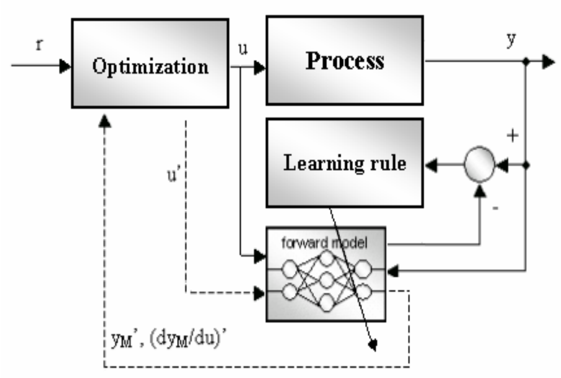

Figure 1: Neuro-predictive controller

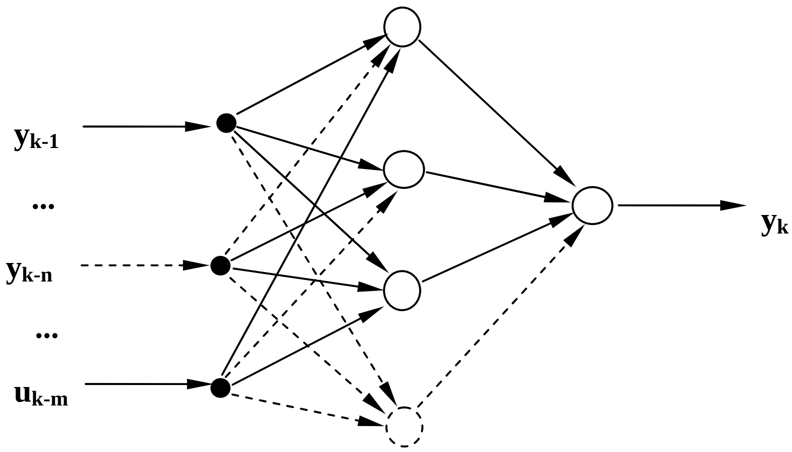

Figure 2: Feed-forward neural network

3 Genetic Algorithm

Genetic algorithm (GA) is a powerful stochastic-based search/optimisation approach, which mimics the evolution in the nature. It is described in e.g. [1],[2],[3] and others. A general scheme of a GA can be described by following steps:

1. Initialisation of the population of chromosomes (set of random potential solutions).
2. Evaluation of the cost function for all chromosomes.
3. Selection of parent chromosomes.
4. Crossover and mutation of the parents $\rightarrow$ children.
5. Completion of the new population from the new children and selected members of the old population. Jump to the step 2 or end.

The chromosomes are linear strings, whose items (genes) represent in our case the designed controller parameters. Before each simulation, the corresponding chromosome (genotype) is decoded into controller parameters of the simulation model in Simulink (phenotype) and after the simulation the selected performance index is evaluated. An example of a simple performance index is as follows

$$I_{AE} = \int_0^T |e(t)| dt \qquad (1)$$

where e is the control error. A block scheme of a GA-based design is in Fig.3.

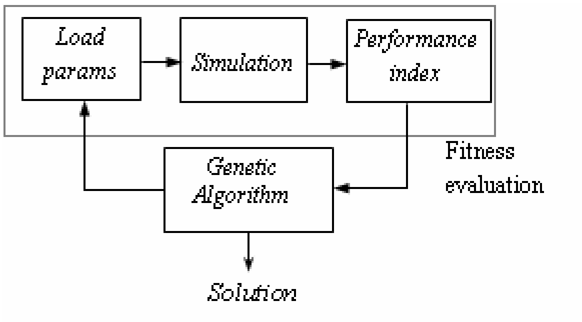

Figure 3: Block scheme of the GA-based design

4 Optimisation of the neuro-predictive controller

For the verification of the proposed methods consider the non-linear system which is described by the differential equation

$$y'' + 0{,}7y' + 0{,}2y + 0{,}3y^3 - u = 0\ .$$

In case of the neural model the number of neurons in the input and in the hidden layer and their interconnection have been searched. In the input layer neurons for the following input signals: $y_{k-1}, y_{k-2}, y_{k-3}, \ldots u_k, u_{k-1}, u_{k-2} \ldots$ are considered. The interconnection map of the net is coded into the chomosome of the GA. The used cost function (fitness) was considered is in the form

$$F = MSE + (1-\alpha) * (w/wn) * 10^{-2} + \alpha * (n/nn) * 10^{-2}$$

MSE – mean square error of the neural model (in comparison to the modelled object)
α – veight constant, which was set to 0,3
w – number of veighted interconnections between the input and hidden layer
wn – maximal number of all interconnections
n – number of neurons in the hidden layear
nn – maximal number of hidden neurons

The second design step was the search for the predictive controller cost function coeficients N_1, N_2, N_u and ρ. For this purpose another GA has been used, where the fitness consists of the closed-loop simulation and the peformance index (1) evaluation. This performance index has been minimised, and it represents the controller performance. The obtained results after these two design steps, which were performed off-line are demonstrated in fig.4. It is the time-response of the controlled system (green) after the reference signal steps (blue). The control value time-response is in fig.5. In fig.6 the trend of the fitness function is depicted, which is the graph of the current best individual of the actual population vs. generation number.

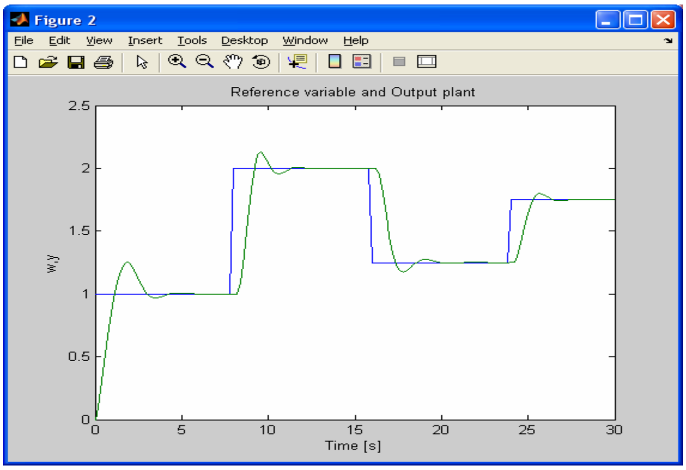

Figure 4: Time-response of the controlled system

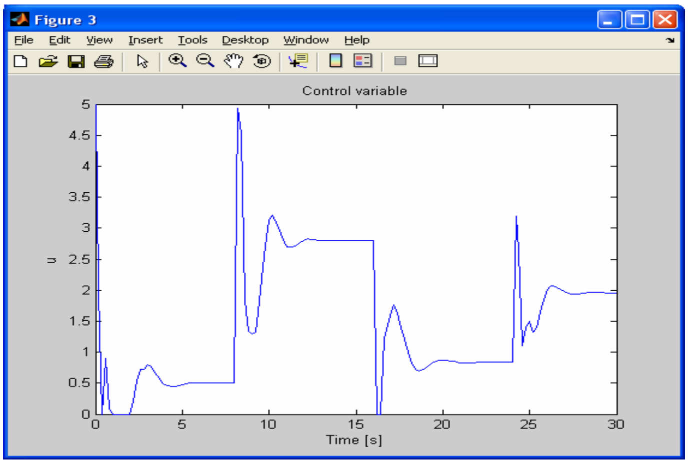

Figure 5: Control value

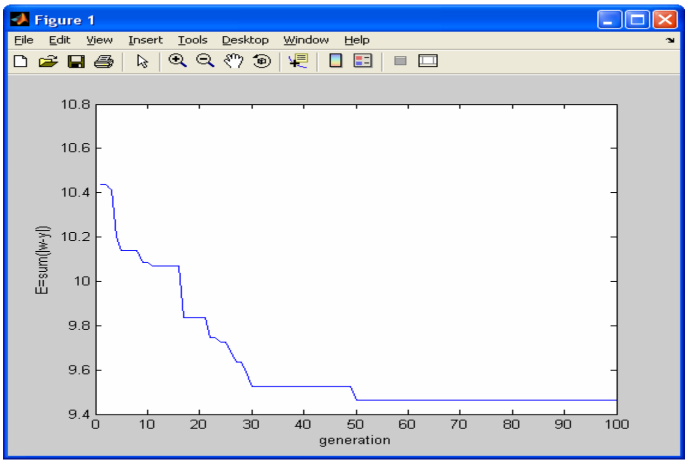

Figure 6: Evolution of the fitness function

5 Conclusion

Neuro-predictive control is an efficient mean for control of non-linear dynamic systems. To ensure the required performance, the controller has to fulfil some conditions. The first is that the neural-model has sufficient modelling accuracy in the entire working range. The second is the satisfactory parameter setting of the predictive controller. Both conditions require some knowledge or experience from the designer. The goal of the presented project is to replace the need for experience and knowledge and to design/optimize the neuro-predictive controller structure and its parameters using an automatic procedure, which is based on genetic algorithms. Evolutionary algorithms can be used for the design and optimization of various neuro-predictive controllers, which uses various types of neural networks and various types of predictive control algorithms.

Acknowledgement

This project has been supported from the grant No. VEGA 1/3100/06 of the Slovak Grant Agency.

References

[1] D.E.Goldberg, Genetic Algorithms in Search, Optimisation and Machine Learning. Addisson-Wesley, 1989
[2] Z. Michalewicz, Genetic Algorithms + Data Structures = Evolutionary Programs. Springer, 1996
[3] I.Sekaj, Evolučné výpočty a ich využitie v praxi, Iris 2005, Bratilava (in slovak)
[4] I. Sekaj, "Robust Controller Design Based on Genetic Algorithms and System Simulation", In: *Proceedings on the conference CDC-ECC'05* (on CD), Seville, Spain, December 12-15, 2005
[5] Michal Spišiak, PhD thesis, FEI STU Bratislava, 2007
[6] Peter Sartoris, Diploma thesis, FEI STU Bratislava, 2008
[7] The Mathworks, Matlab Release 2006b, documentation on CD

Doc. Ing. Ivan Sekaj, PhD, E-mail: ivan.sekaj@stuba.sk
Ing. Slavomír Kajan, PhD, E-mail: slavomir.kajan@stuba.sk
Ing. Ladislav Körösi, E-mail: ladislav.korosi@stuba.sk
Ing. Zuzana Didekova, E-mail: zuzana.didekova@stuba.sk
Ing. Leo.Mrafko, E-mail: leo.mrafko@stuba.sk

Institute of Control and Industrial Informatics, Faculty of Electrical Engineering and Information Technology, Slovak University of Technology in Bratislava,