ISBN 978-1-84626-xxx-x
Proceedings of 2011 International Conference on Optimization of the Robots and Manipulators (OPTIROB 2011)
Sinaia, Romania, 26-28 Mai, 2011, pp. xxx-xxx

Assisted Research and Optimization of the proper Neural Network Solving the Inverse Kinematics Problem

Adrian Olaru [1+], Serban Olaru [2] , Dan Paune [1] and Oprean Aurel [3]

[1] University Politehnica of Bucharest

[2] Romsys Mechatronics Systems Company

[3] Romanian Academy of Technical Sciences

Abstract. Finding the better solution of the neural network design to solve the inverse kinematics problem with the minimum of the trajectory errors is very difficult, because there are many variable parameters and many redundant solutions. The presented paper show the assisted research of the influences of some more important parameters to the final end-effector trajectory errors of the proposed neural network model solving the inverse kinematics problem. We were been studied the number of neurons in each layers, the sensitive function for the first and second layer, the magnifier coefficient of the trajectory error, the variable step of the time delay and the position of this block, the different cases of target data and the case when the hidden target data were adjusted. All obtained results were been verified by applying the proper direct kinematics virtual LabVIEW instrumentation. Finally we were obtained one optimal Sigmoid Bipolar Hyperbolic Tangent Neural Network with Time Delay and Recurrent Links (SBHTNN(TDRL)) type, what can be used to solve the inverse kinematics problem with maximum 4% of trajectory errors.

Keywords: inverse kinematics, neural network, trajectory error, LabVIEW instrumentation

1. Introduction

The inverse kinematics was used to control the end-effector trajectory. The inverse kinematics solutions obtained by geometrical method are more difficult to find, when the robot degree of freedom increase. Inverse kinematics solutions are obtained usually by geometrical method, numerical method with knowing outputs and with neural network optimization [1, 2, 3, 4, 5]. The neural network method to obtain the real solutions of the inverse kinematics in the actual research doesn't show the simulation results and the optimization of the errors. In the paper was proposed for optimization of the trajectory error, after apply the inverse kinematics control, one new method with proper neural network what used three layers, many time delay blocks and recurrent links. All layers used the sensitive sigmoid bipolar hyperbolic tangent function types to take in consideration the influences of the input data to the internal coordinates q_i in all two directions of the movement [6, 7, 8, 9, 10]. The last layer is used to adapt the number of data vector with the needed number of output. The optimal errors were been obtained by applying the back propagation proper method, the bipolar sigmoid hyperbolic tangent sensitive function, the multiple time delay and the recurrent links and some of the results of the presented research.

2. Generality of the Neural Networks and the Studied Robot

Neural network are composed of simple elements operating in parallel, like a biological nervous systems. As in nature, the connections between elements largely determine the network function. You can train a neural network to perform a particular function by adjusting the value of the connections (weights) between elements, the hidden targets and the biases. Neural network have been trained to perform complex functions

+ Corresponding author. Tel.: +0040723852628; fax: +0040314013265.
E-mail address: olaru_robotics@yahoo.com.

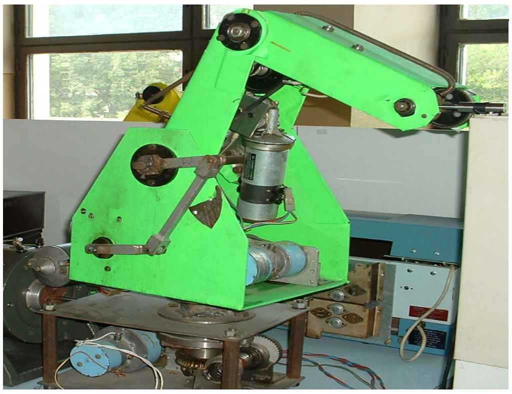

Fig.1: The didactical arm type studied robot

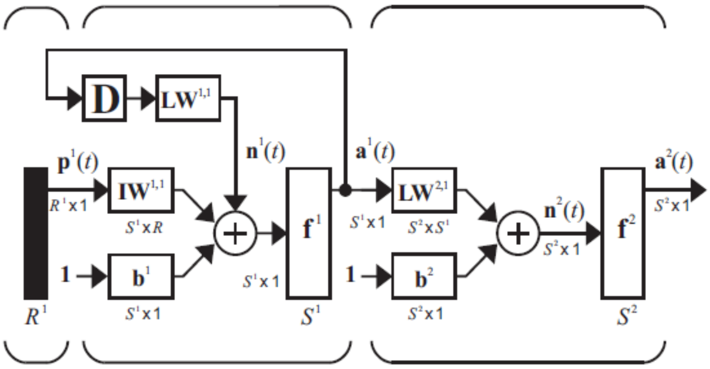

Fig.2: The Layer Neural Network (LNN)

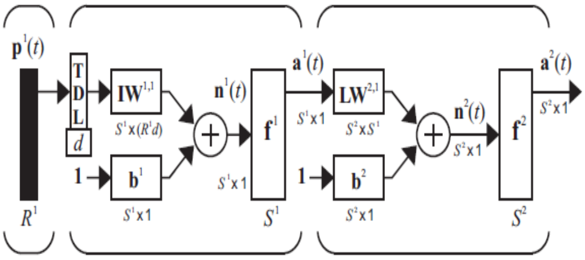

Fig.3: Focused Time-Delay Neural Network (FTDNN)

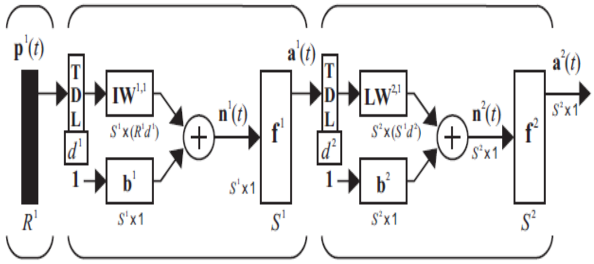

Fig.4: Distributed Time-Delay Neural Network (DTDNN)

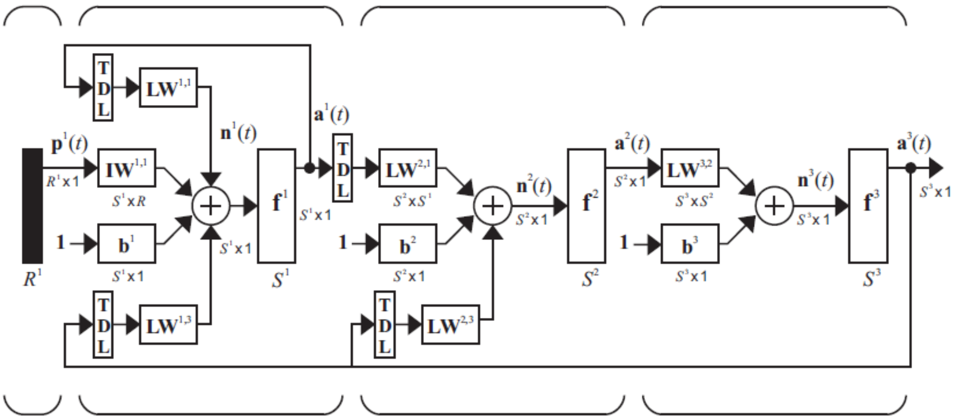

Fig.5: Layered Digital Dynamic Network (LDDN)

in various fields including pattern recognition, identification, classification, speech, vision and control systems [11-16]. We consider neural network as an alternative computational scheme rather that anything else. The artificial neural networks which we described in this paper are all variation on parallel distributed processing (PDP) idea. An artificial network consists of a pool of simple processing units which communicate by sending signals to each other a large number of weighted connections. Typically, neural network are adjusted, or trained, so that a particular input leads to specific target output. There, the network is adjusted, based on comparison of the output and the target, until the network output matches the target. Typically, many such input/target pairs are needed to train a network. The more important neural networks used in the recognition the voice, the form, to optimize the guidance trajectory and the researched robot are presented in the figs.1-5 [6-10]. For the assisted solving the inverse cinematic problem was used one neural network with three layers with the following configuration: 3-8-3-3 with bipolar sigmoid hyperbolic tangent neural network type. This configuration was choose because the input must be 3 to control three internal coordinates (relative angles), 8 because more neurons in hidden layer not influence positive the target errors like you can see in the presented research, 3 because must be used the direct kinematics to obtain the output, 3 because the output must be the same number with target (target are the space position of the end- effector).

3. Mathematical Model and LabVIEW Instrument of the used Neural Network

To solve the inverse kinematics problem was used one proper Bipolar Sigmoid Hyperbolic Tangent Neural Network type with some Time Delays and Recurrent Links (BSHTNN (TDRL)) with intermediate control of the target after each layer. The used neural network was 3-8-3-3 type, with three layers. The simplified schema we can see on the fig. 6. The mathematical model of the used Neural Network is shown in relation (1), where were notated all researched parameters, p_1-p_9.

$$
\begin{aligned}
n_1 &= [\underbrace{w^1}_{p_1} + \underbrace{tcg_1}_{p_2} \cdot \varepsilon_1](p - a_2(t - p_3 + 1)) + (b_1 + \varepsilon_1) \\
a_1 &= \frac{p_4(1 - e^{-n_1})}{1 + e^{-n_1}} \\
\varepsilon_1 &= t_1 - a_1 \\
n_2 &= [w^2 + \underbrace{tcg_2}_{p5} \cdot \varepsilon_2](a_1(t - p_6 + 1)) + (b_2 + \varepsilon_2) \\
a_2 &= \frac{p_7(1 - e^{-n_2})}{1 + e^{-n_2}} \\
\varepsilon_2 &= t_2 - a_2 \\
q_i &= p_8(a_2 - \varepsilon_f) \\
r_i &= \begin{pmatrix} c_1 s_2 l_3 + (c_1 c_2 s_3 + c_1 s_2 c_3) l_4 \\ s_1 s_2 l_3 + (s_1 c_2 s_3 + s_1 s_2 c_3) l_4 \\ l_1 + l_2 + c_2 l_3 + (-s_2 s_3 + c_2 c_3) l_4 \end{pmatrix} \\
\varepsilon_{pos} &= t_3 - r_i \\
n_3 &= [w^3 + \underbrace{tcg_2}_{p_5} \cdot \varepsilon_{pos}](q_i) + (b_3 + \varepsilon_{pos}) \\
a_3 &= \frac{p_9(1 - e^{-n_3})}{1 + e^{-n_3}} \\
\varepsilon_f &= t_2 - a_3
\end{aligned}
\tag{1}
$$

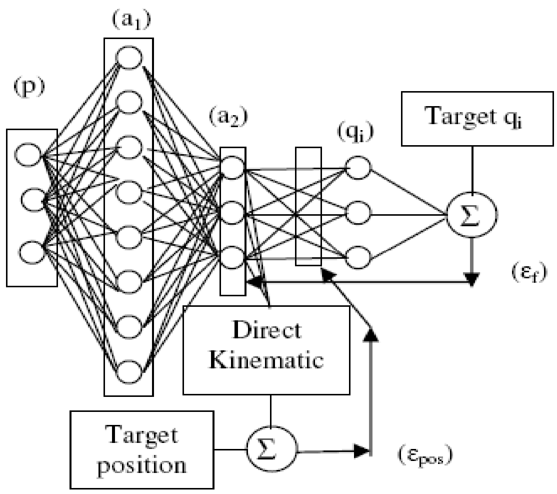

Fig.6: The proposed Neural Network. Simplified schema

Parameters are: p_i are the assisted research proposed parameters about p_1- the number of neurons; p_2 – the first teaching gain; p_3- step of the first time delay; p_4- the first sensitive function gain; p_5- the second teaching gain; p_6- the step of the second time delay; p_7- the second sensitive function gain; p_8- the magnify gain of the proportional error control; p_9- the third sensitive function gain.

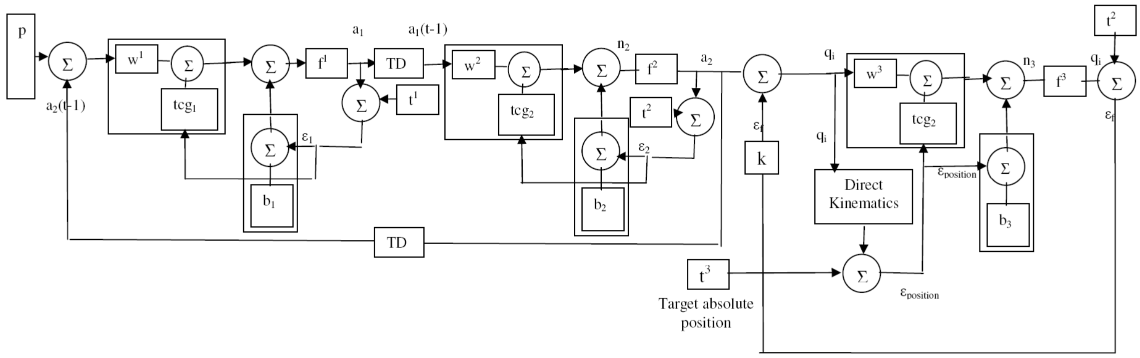

Fig.7: Detailed schema of the researched Neural Network

This mathematical model and the final form of the neural network schema were been established after analyzing of some simulation LabVIEW results. With virtual instrumentation easily can take some new links, loops or input correction of the model and established the optimal hidden target to obtain the minimum of the end-effector errors after assisted determination of the internal coordinates by using the inverse kinematics.

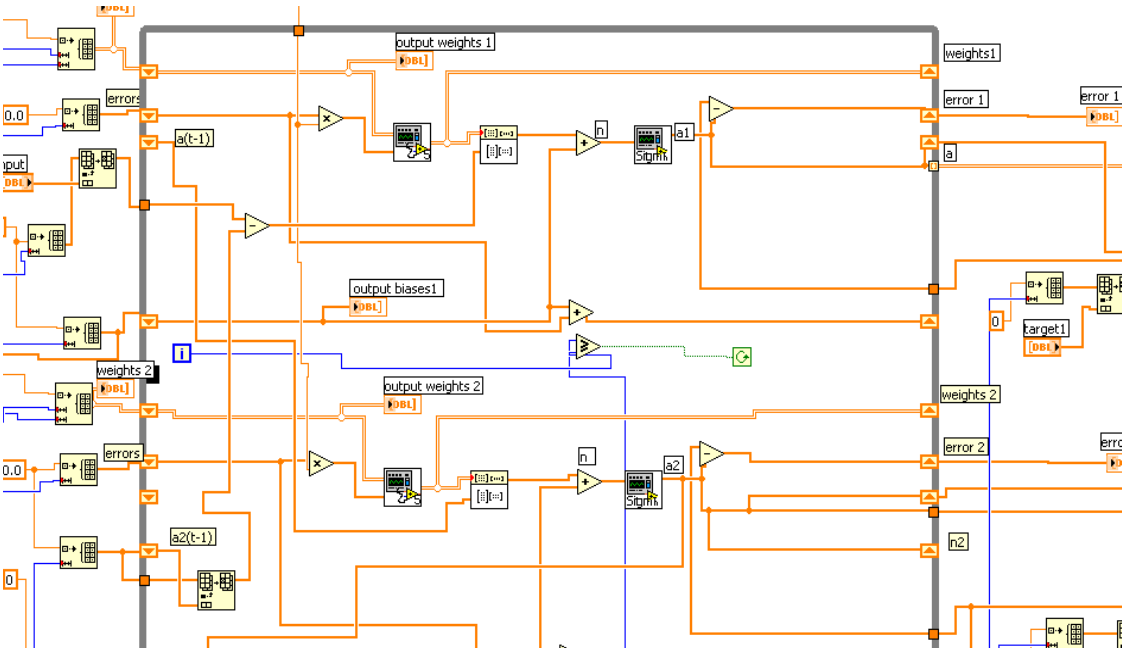

Fig.8: Part of the block schema of the researched Neural Network

4. Assisted Research of some Parameters of proposed Neural Network

The assisted research shown in the paper studied the influences to the errors between the neural network output and the target (the space position of the end-effector) of the following neural network parameters: number of neurons in each layers, the sensitive function for the first and second layer, the magnifier coefficient of the trajectory error, the variable step of the time delay and the position of these blocks, the different cases of target data and the case when the hidden target data were on-line adjusted.

Some of the studied cases are shown in the figs.9-14 for same different parameters. The results are synthetic shown in the table 1.

Table 1 The synthetic results of the assisted research of the proposed neural network

Fig. number	Config.	Amplif. gain	Teaching gain	l_i	Trget data	Obt. Pos.	Obtained q_i	Target hidden	Iteration number	Relative errors
9	3-8-3-3	1.8	0.2	350	550 300 450	447.65 244.17 430.09	28.610 50.420 61.306	1 0.7 0.8	132	18%
10	3-7-3-3	1.8	0.2	350	-‘-	447.65 244.17 430.09	28.610 50.420 61.306	1 0.7 0.8	132	18%
11	3-8-3-3	1.6	0.5	350	-‘-	525.55 286.66 471.85	28.610 49.274 58.441	1 0.7 0.7	132	4%
12	3-8-3-3	0.24	0.5	350	-‘-	578.42 315.50 453.53	28.610 63.394 35.351	0.4 0.4 0.5	32	4%
13	3-8-3-3	1.6	0.5	350	-‘-	502.04 273.84 425.81	28.610 47.947 69.003	1 0.7 1	8	9%
14	3-8-3-3	1.6	0.5	350	-‘-	526.92 287.41 424.86	28.610 52.668 60.445	0.4 0.6 0.4	32	4%

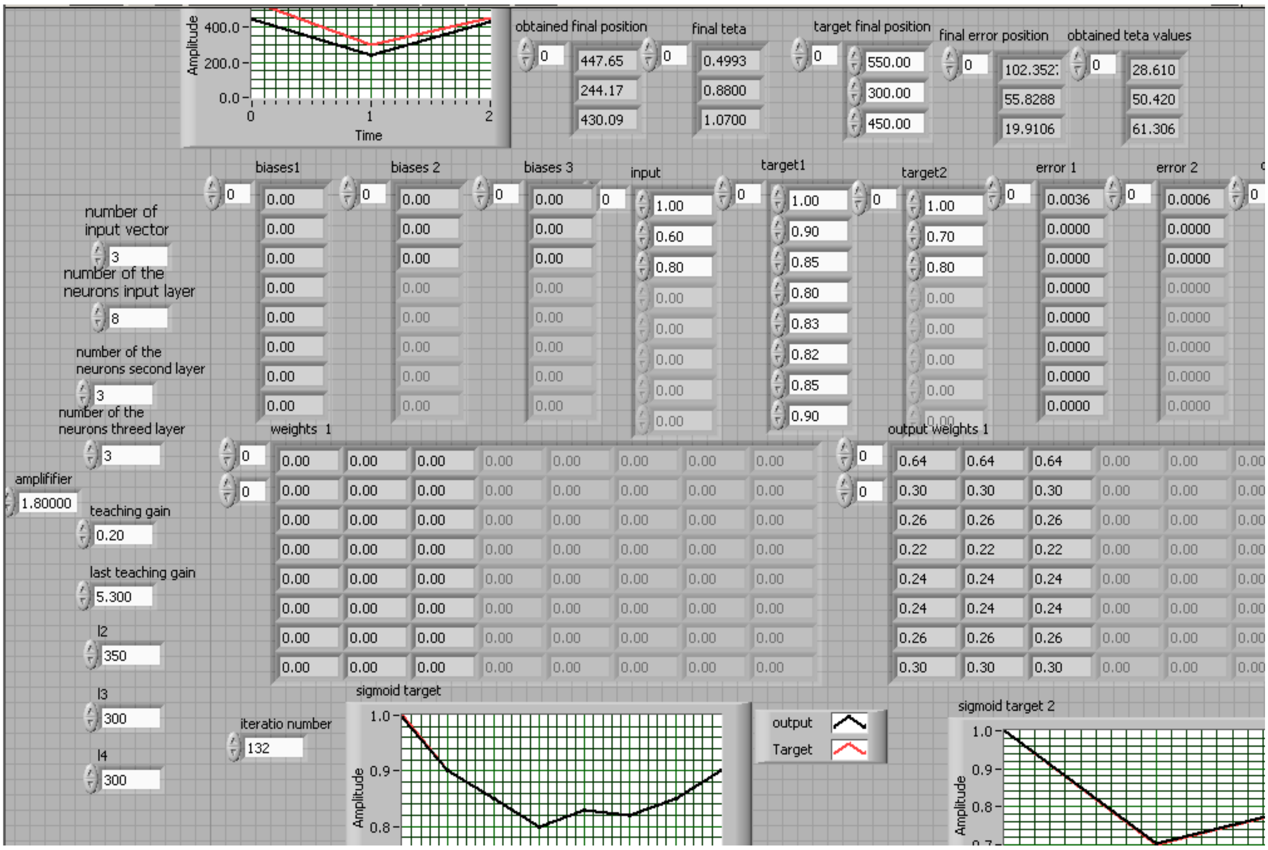

Fig.9: The front panel with the input, target data, errors and weights and biases matrices for the first data

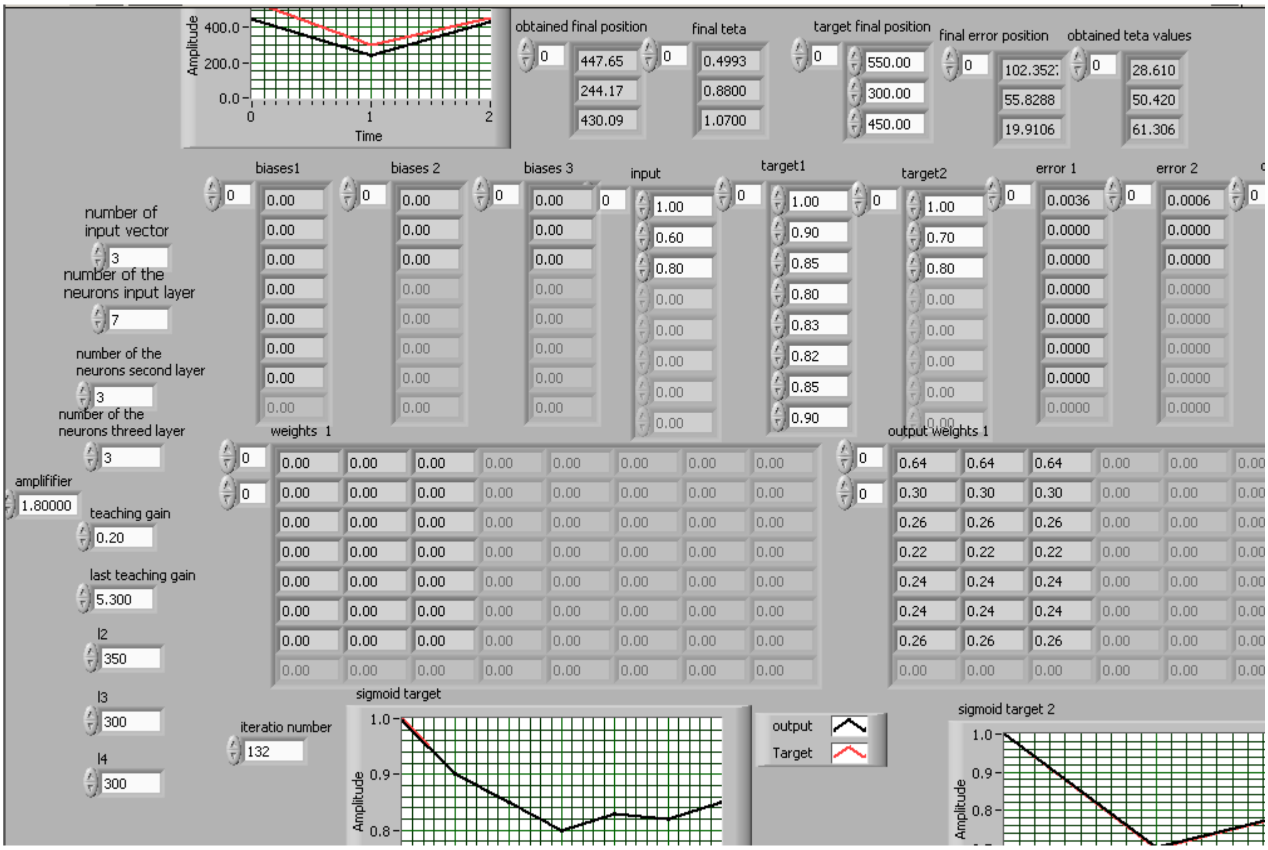

Fig.10: The front panel with the input and neural network target data, errors and weights and biases matrices when was changed the neuron number of the first layer

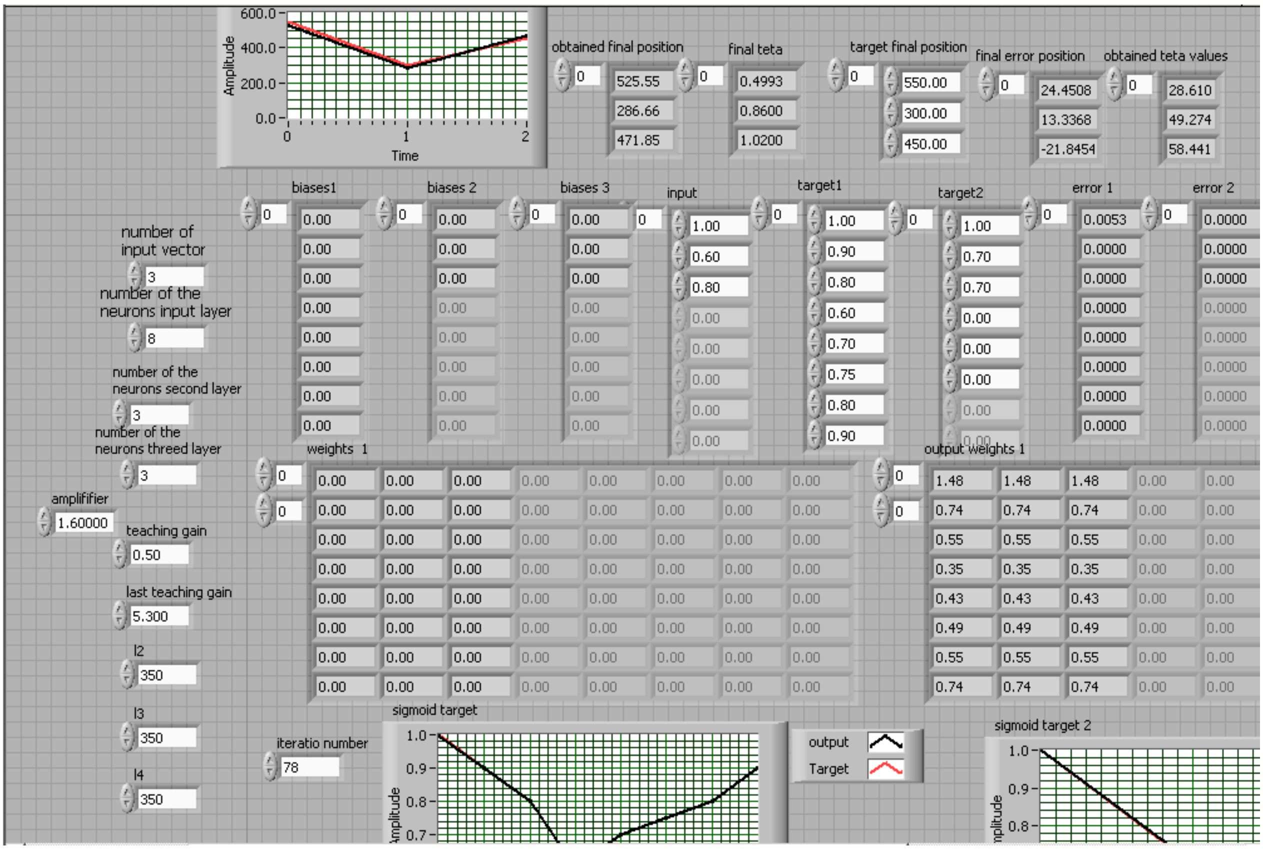

Fig.11: The front panel with the input and neural network target data, errors and weights and biases matrices when were changed the hidden target data, amplifier and teaching gain

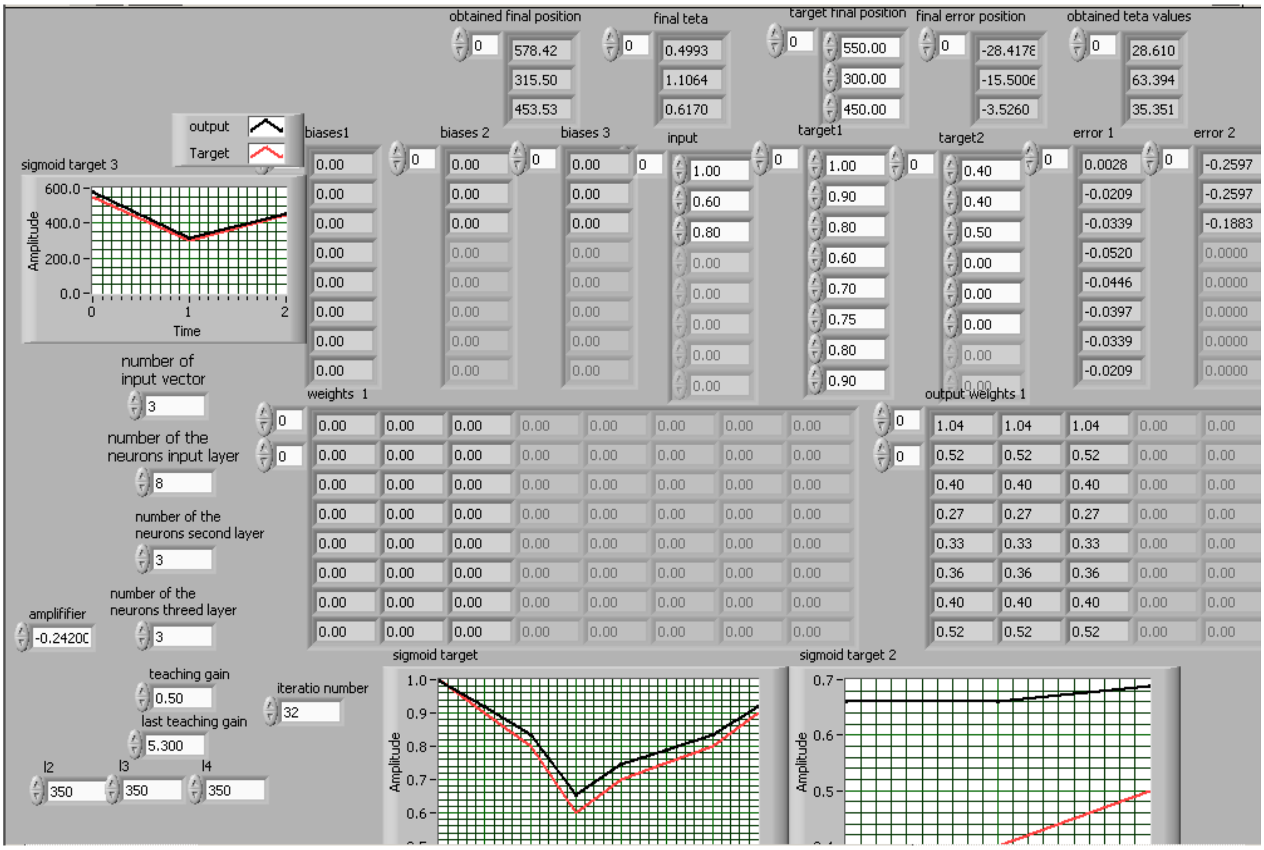

Fig.12: The front panel with input and neural network target data, errors, weights and biases matrices when were changed the hidden layer target, the teaching gain and amplifier gain

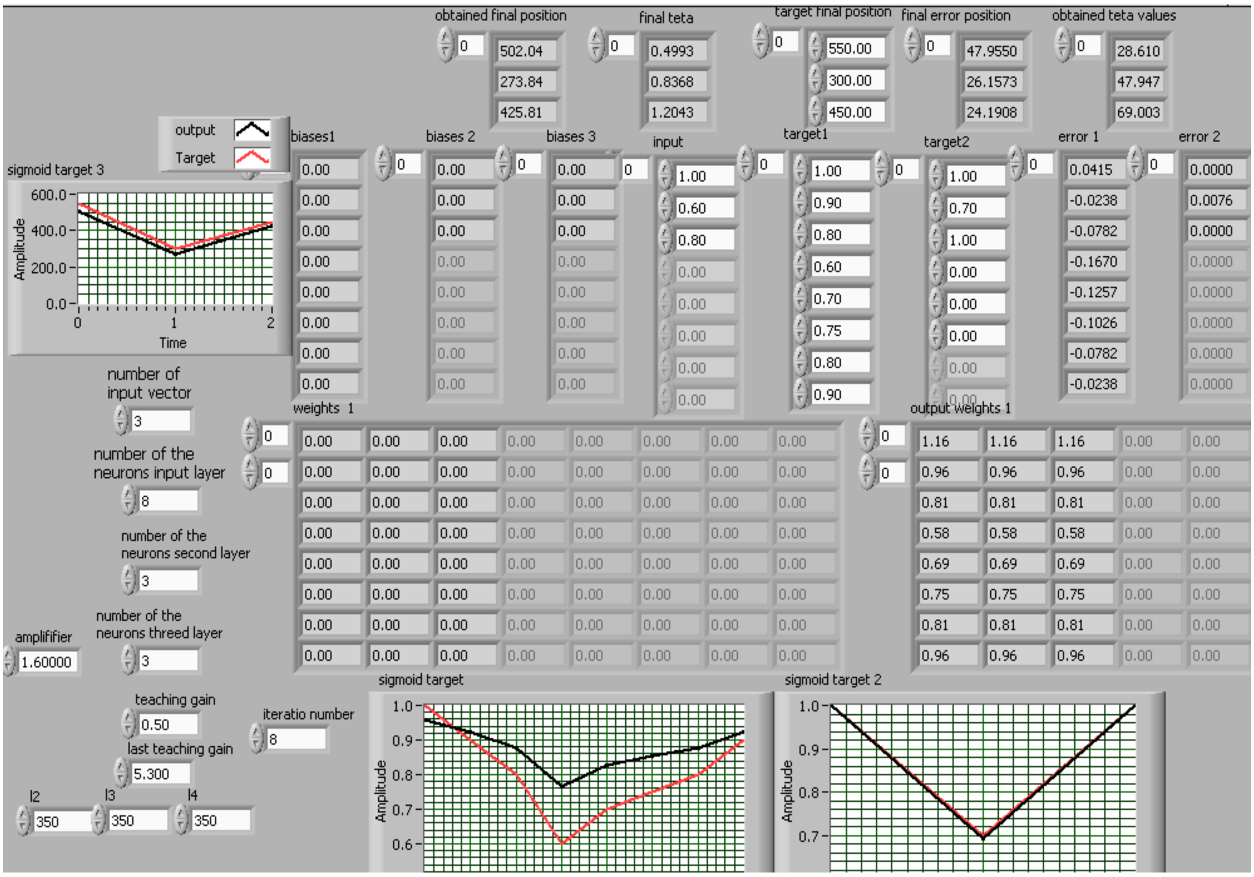

Fig.13: The front panel with input and neural network target data, errors, weights and biases matrices when were changed the hidden target data, teaching gain

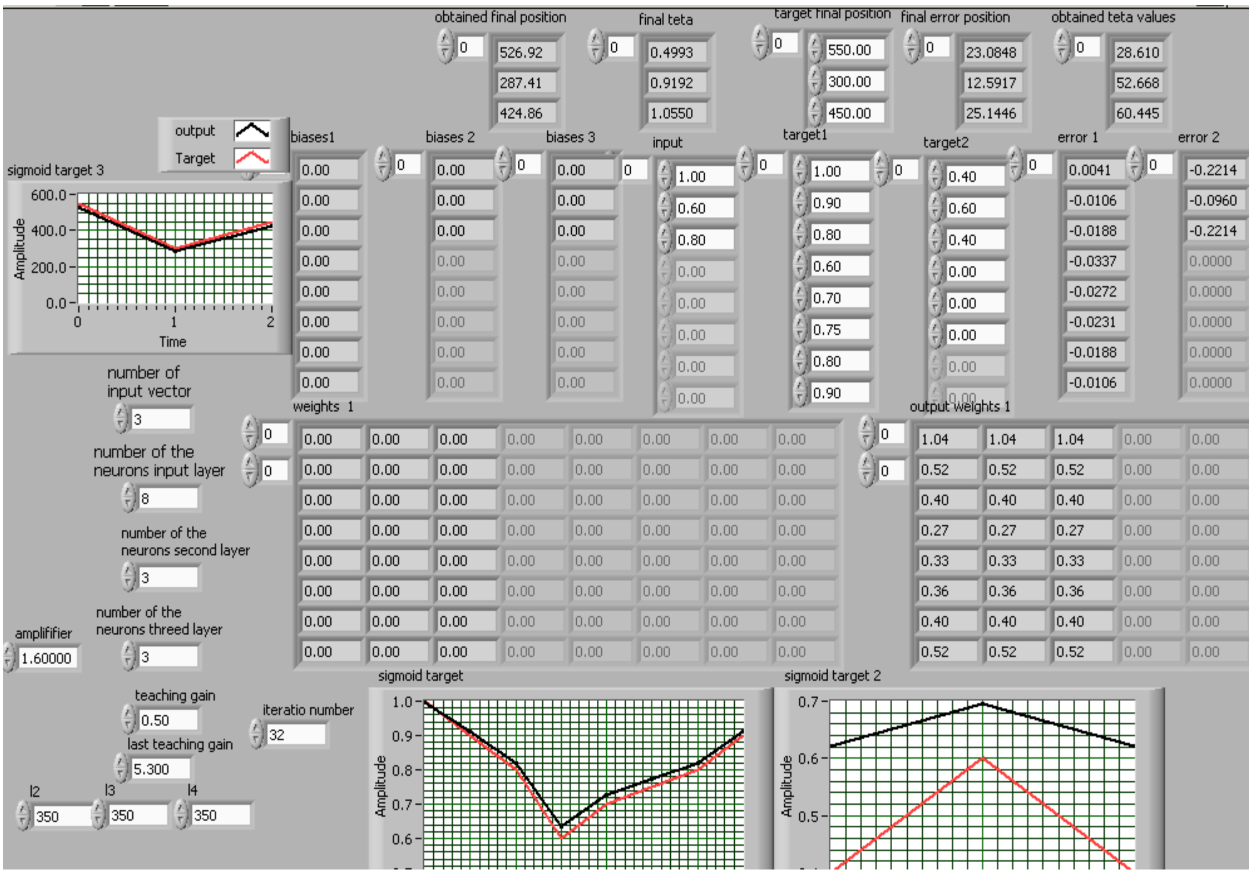

Fig.14: The front panel with input, neural network target, weights and biases matrices when were changed the amplifier gain, teaching gain, the hidden layer target data

5. Discussion and Conclusions

After analysing the figs.9-14 and the results obtained after the assisted research, synthetic presented in the table 1, we can do the following remarks: the change of the number of the neurons in the first layer don't change the errors; the change of the amplifier gain and the teaching gain assured the decrease of the error from 18% to 4% for the 132 number of iteration; one substantial decreasing of the errors and the decreasing of the number of iteration was obtained by on-line changing of the hidden layer target data, 18% to 4% for 32 iteration. With this method, by applying the control of the inverse kinematics and by using the proposed neural network type, will be possible to obtain one optimization of the robot end- effector position in the space.

The applying method, the proposed neural network, the assisted research with the virtual LabVIEW instrumentation opens the way to apply in to the robot control, the intelligent systems.

6. References

[1] P.J.Alsina, N.S.Gehlot, Direct and inverse kinematics of robot manipulator based on modular neural networks, *ICARCV*, *IEEE*, pp. 1743-7, **3**, 1994.

[2] R. Manseur, D.Keith, A fast algorithm for reverse kinematics analysis of robot manipulator, *International Journal of Robotics Research,* **7** (3), 1998, pp. 622-648.

[3] D. Gorinevsky, T.Connoly, Comparations of some neural network and scattered data approximations: The inverse manipulator kinematics example, *Neural computation*, **3** (6), pp.521-542.

[4] L.Lee, Training feedforward neural networks: An algorithm givin improved generalization, *Neural Networks*, **10** (1), pp.61-68, 1997.

[5] C. Schittenkopf, G.Deco and W. Brauner, Two strategies to avoid overfitting in feedforward networks, *Neural Networks*, **10** (3), pp.505-516, 1997.

[6] Olaru, A., Olaru S., Ciupitu, L. Assisted research of the neural network by bach propagation algorithm, OPTIROB 2010 International Conference, Calimanesti, Romania, *The RPS Singapore Book*, pp.194-200 , 2010.

[7] Olaru, A., Olaru, S., Paune D., Ghionea A. Assisted research of the neural network, OPTIROB 2010 International Conference, Calimanesti, Romania, *The Research Publishing Services Singapore Book*, pp.189-194 , 2010.

[8] Olaru, A., Olaru, S. Assisted research of the neural network with LabVIEW instrumentation, *IEEE ICMENS-2010 Proceedings*, Changsha, China, pp.1-8, 2010.

[9] Olaru, A., Oprean, A., Olaru, S., Paune, D. Optimization of the neural network by using the LabVIEW instrumentation, *IEEE ICMERA 2010 Proceedings*, ISBN 978-1-4244-8867-4, IEEE catalog number CFP1057L-ART, pp.40-44, 2010.

[10] Olaru, A., Olaru, S., Paune, D. Assisted research dynamic neural network with time delay and recurrent links, *IEEE ICMERA 2010 Proceedings*, ISBN 978-1-4244-8867-4, IEEE catalog number CFP1057L-ART, pp.284-288, 2010.

[11] M. Minsky and S. Papert, *Perceptrons: an introduction to computational geometry,* The MIT Press, Cambridge MA, 1969.

[12] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities", *Proc. of the National Academy of Sciences,* 81, 1984, pp.3088-3092.

[13] Hajduk, M., Semjon, J., Vagaš, M.: *Design of the Welding Fixture for the Robotic Station for Spot Weldind Based on the Modular Concept.* In: Acta Mechanica Slovaca No. 3/2009, volume 13, p. 30-35, ISSN 1335-2393.

[14] Hajduk, M., Sukop, M., Baláž, V., Semjon, J., Vagaš, M.: *Improving the performance of manufacturing systems based reconfiguring and computer integration.* Košice, 2006.

[15] Páchniková L., Šalátová, M.: *Ciele a efekty pružných výrob*. In: Acta Mechanica Slovaca, roč. 12, č. 2-A, 2008, p. 451- 454, ISSN 1335-2393.

[16] Páchniková L.: *Profiling production cells based on simulation*. In: Modelling and designing in production engineering, Lublin 2009, p. 132 -137, ISBN 978-83-87833-91-6.