Proceedings in MANUFACTURING SYSTEMS

Proceedings in Manufacturing Systems, Volume 6, Issue 1, 2011 ISSN 2067-9238

OPTIMISATION OF THE INDUSTRIAL ROBOT INVERSE KINEMATICS SOLUTIONS WITH THE PROPER NEURAL NETWORK AND LABVIEW INSTRUMENTATION

Adrian OLARU[1,*], Şerban OLARU[2], Dan PĂUNE[3]

[1)] Prof, PhD, Eng, Machines and Manufacturing Systems Department, University "Politehnica" of Bucharest, Romania
[2)] PhD, Eng, Mechatronics Department, RomSYS Mechatronics Company, Bucharest, Romania
[3)] PhD Student, Eng, Machines and Manufacturing Systems Department, University "Politehnica" of Bucharest, Romania

Abstract: *Inverse kinematics model of the industrial robot is used in the control of the end-effector trajectory. The solution of the inverse kinematics problem is very difficult to be found, when the degree of freedom increase, in many cases being impossible. In these cases, the numerical approximation or other methods with diffuse logic are used. The paper presents a new method for optimization of the inverse cinematic solution by applying the sigmoid bipolar hyperbolic tangent proper neural network with multiple time delay and recurrent links – SBHTNN (TDRL). Using this proposed method we can obtain quickly after applying the inverse kinematics method one approximated solution accompanied by a decrease of the trajectory errors.*

Key words: *trajectory optimization, direct kinematics, inverse kinematics, neural network, virtual instrumentation.*

1. INTRODUCTION

The inverse kinematics was used to control the end-effector trajectory. It is difficult to find the inverse kinematics solutions obtained by geometrical method when the robot degree of freedom increases. Inverse kinematics solutions are obtained usually by geometrical method, numerical method with known outputs, and neural network optimization [1, 2, 3, 4, and 5]. The neural network method to obtain the real solutions of the inverse kinematics in the actual research of the world does not show the simulation results and the optimization of the errors by root means square method. For optimization of the trajectory error, this paper proposes by applying the inverse kinematics control a new method based on proper neural network that uses three layers, many time delay blocks and recurrent links. All layers use 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, and 10]. The last layer is used to adapt the number of input data vector with the needed number of output. The optimal errors were obtained by applying the back propagation proper method, sigmoid hyperbolic tangent sensitive function, and multiple time delay and recurrent links.

2. DIRECT KINEMATICS MODEL

The experimental research and the theoretical cinematic analysis were realized using one arm type robot (Fig. 1) and some virtual LabVIEW instruments. The structural cinematic schema is shown in Fig. 2. Using the recurrent matrix method all joints positions of the robot structure were obtained.

The recurrent general mathematical model contains the following matrix form:

$$(r)_i^0 = (r)_{i-1}^0 + [D]_{i-1}^0 (r)_i^{i-1}, \tag{1}$$

where $\boldsymbol{r}_i^0$ is the matrix form of the absolute position vector of the joint i; $\boldsymbol{r}_{i-1}^0$ – matrix form of the absolute position of the joint i – 1; $\boldsymbol{r}_i^{i-1}$ – matrix form of the relative position vector between the joints i and i – 1; D_{i-1}^0 – coordinates transform matrix from the joint i – 1 to the base Cartesian system. The transform matrix is calculated as follows:

$$[D]_{i-1}^0 = [D]_1^0 [D]_2^1 ... [D]_{i-1}^{i-2}, \tag{2}$$

where D_{i-1}^{i-2} is the coordinates transformation matrix from the Cartesian system i – 1 to the i – 2.

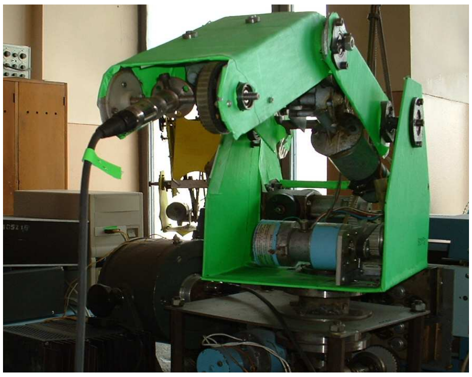

Fig. 1. The used didactical arm type robot.

* Corresponding author: Splaiul Independentei, nr. 313, sector 6, 060042, Bucharest, Romania
Tel.: 0040723852628;
Fax: 0040314013265.
E-mail addresses: *aolaru_51@yahoo.com* (A. Olaru), *serban1978@yahoo.com* (Ş. Olaru) and (D. Păune)

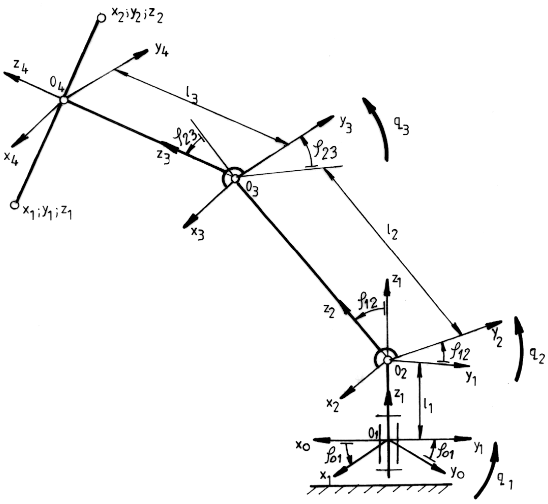

Fig. 2. The cinematic structural schema of the studied arm type robot.

After applying the recurrent mathematical model to the presented robot structure, the following position vectors of all robot joints were obtained:

$$(r)_1^0=\begin{pmatrix}0\\0\\l_1\end{pmatrix};(r)_2^0=\begin{pmatrix}0\\0\\l_1+l_2\end{pmatrix};(r)_3^0=\begin{pmatrix}c_1s_2l_3\\s_1s_2l_3\\l_1+l_2+c_2l_3\end{pmatrix};$$
$$(r)_4^0=\begin{pmatrix}c_1s_2l_3+(c_1c_2s_3+c_1s_2c_3)l_4\\s_1s_2l_3+(s_1c_2s_3+s_1s_2c_3)l_4\\l_1+l_2+c_2l_3+(-s_2s_3+c_2c_3)l_4\end{pmatrix}. \tag{3}$$

where l_i is the lengths of each robot modules; c_i, s_i – cosines and sinus trigonometric functions of the relative angle and relative robot coordinate q_i between i and $i-1$ robot bodies. The direct kinematics LabVIEW icon and the results of the direct cinematic analysis are shown in Figs. 3–5.

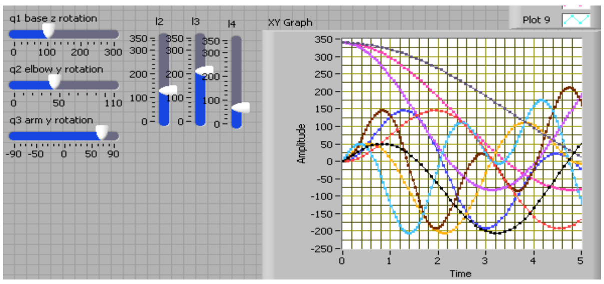

Fig. 3. The variation of the absolute coordinate functions of q_1, q_2 or q_3.

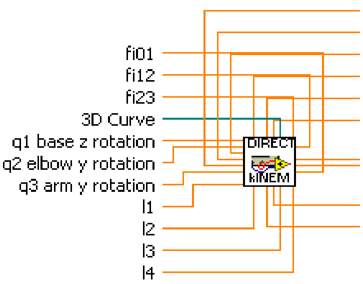

Fig. 4. The icon of the LabVIEW VI for the assisted simulation of the direct kinematics.

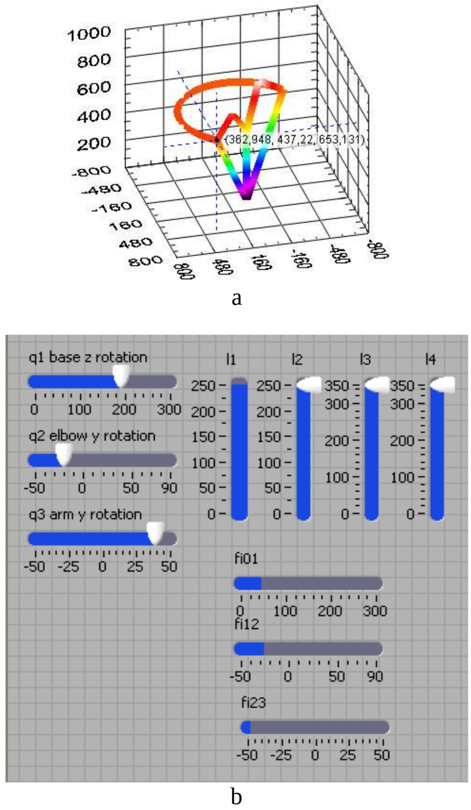

Fig. 5. The absolute variation of the position for some input data of the internal coordinates q_i; *a* – 3D trajectory and coordinates of the end point; *b* – front panel with the input data.

With the virtual LabVIEW instrumentation for direct kinematics we can show the absolute Cartesian values for different input relative coordinates q_i. We used this instrument to verify the relative coordinates obtained after applying the proper neural network method for inverse kinematics settlement.

3. NEURAL NETWORK APPLICATION

To solve the inverse kinematics problem, one proper Bipolar Sigmoid Hyperbolic Tangent Neural Network type was used having some Time Delays and Recurrent Links (BSHTNN (TDRL)) with intermediate control of the target after each layer. The used neural network is of 8-3-3 type, with three layers that can be seen in Fig. 6.

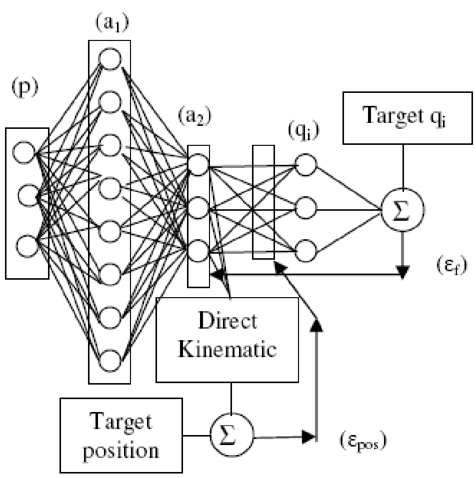

Fig. 6. The Neural Network simplified schema.

We used the neural network calculus complex schema (Fig. 7) to develop the LabVIEW virtual instrumentation to simulate the inverse kinematics settlement. In the neural network schema, we used the following notations: p is the input matrix vector; a_i – output matrix from each neural network layer; $a_i(t-1)$ – output matrix after one delay time block; w^i – weight matrix of each neural network layer; tcg_i – teaching gain; b_i – biases matrix of each layer; f – sensitive function; t – target matrix after each layer; n_i – output matrix before applying the sensitive function; ε_i – error matrix of position or of robot coordinates after each layer; k – magnifier gain to proportional control of the error.

Using the neural network schema to solve the inverse kinematics problem, we achieved the following complex matrix mathematical model:

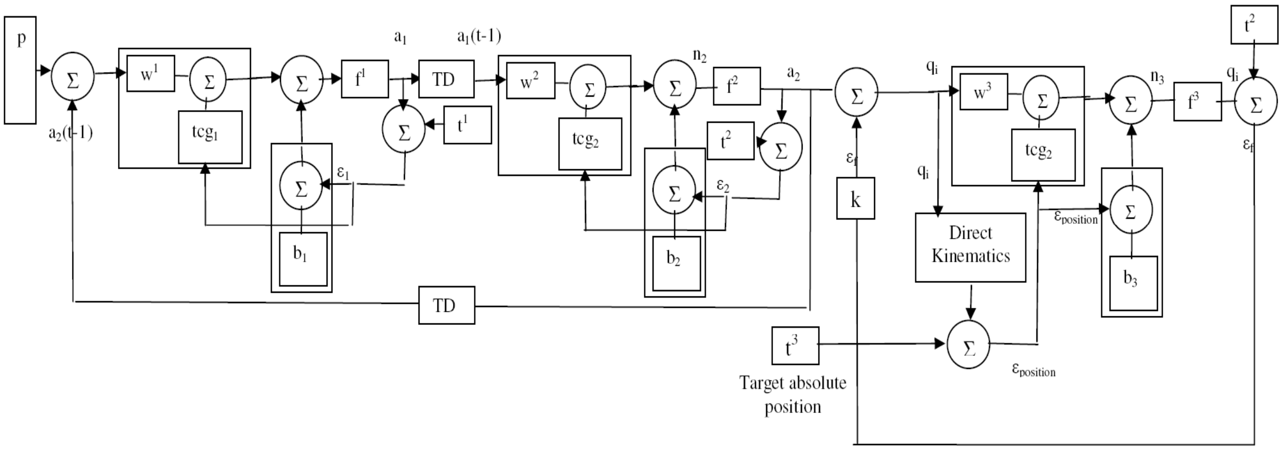

Fig. 7. The Neural Network calculus schema.

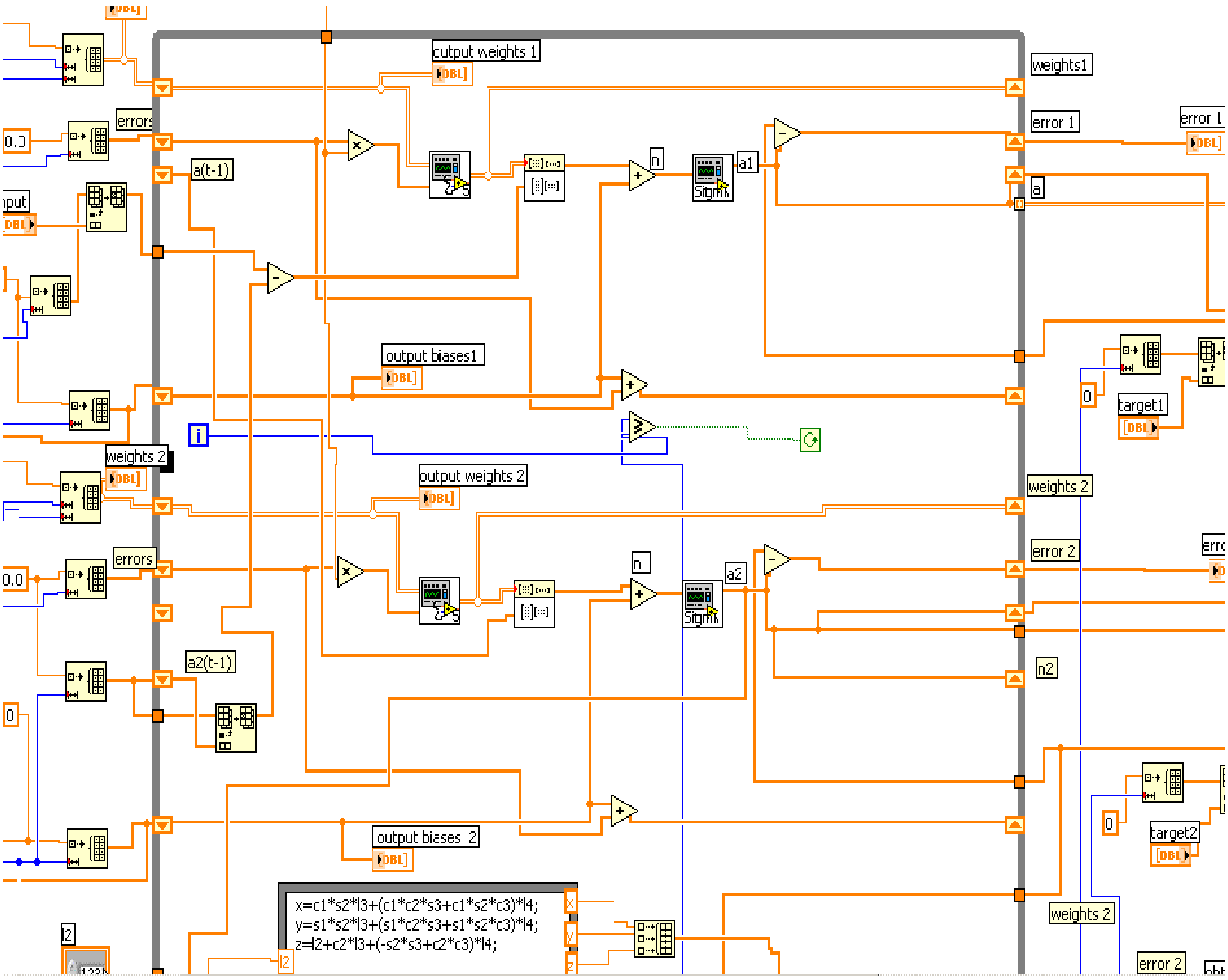

Fig. 8. Part of the Neural Network block schema.

$$
\begin{gathered}
n_1 = [w^1 + tcg_1 \cdot \varepsilon_1](p - a_2(t-1)) + (b_1 + \varepsilon_1);\\
a_1 = \frac{a(1-e^{-n_1})}{1+e^{-n_1}};\\
\varepsilon_1 = t_1 - a_1;\\
n_2 = [w^2 + tcg_2 \cdot \varepsilon_2](a_1(t-1)) + (b_2 + \varepsilon_2);\\
a_2 = \frac{a(1-e^{-n_2})}{1+e^{-n_2}};\\
\varepsilon_2 = t_2 - a_2;\\
q_i = k(a_2 - \varepsilon_f);\\
r_i = \begin{pmatrix} c_1 s_2 l_3 + (c_1 c_2 s_3 + c_1 s_2 c_3) l_4 \\ s_1 s_2 l_3 + (s_1 c_2 s_3 + s_1 s_2 c_3) l_4 \\ l_1 + l_2 + c_2 l_3 + (-s_2 s_3 + c_2 c_3) l_4 \end{pmatrix};\\
\varepsilon_{pos} = t_3 - r_i;\\
n_3 = [w^3 + tcg_2 \cdot \varepsilon_{pos}](q_i) + (b_3 + \varepsilon_{pos});\\
a_3 = \frac{a(1-e^{-n_3})}{1+e^{-n_3}};\\
\varepsilon_f = t_2 - a_3.
\end{gathered}
\tag{4}
$$

This model and the final form of the neural network schema were established after analyzing some simulation LabVIEW results. With virtual instrumentation we easily can make some new links, loops or input correction of the model.

4. EXPERIMENTAL RESULTS

The experimental results were obtained after running the neural network for some different input data and verifying the results obtained for internal coordinate q_i with the direct kinematics LabVIEW VI.

The icon of the inverse kinematics neural network is shown in Fig. 9.

All the weights and biases matrix were initialized by zero values. The magnifier gain of proportional control of the errors was imposed at the initial π value to assure on the internal robot coordinate the gain of the movement in two directions by 180 degree, because the maxim output from the sigmoid neural network was ±1.

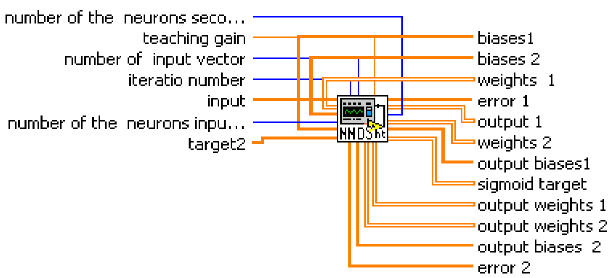

Fig. 9. Proper neural network icon for inverse kinematics.

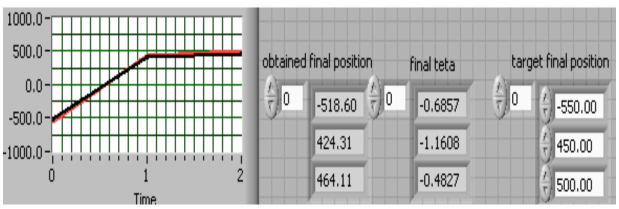

Fig. 10. The front panel with input data and some results of the LabVIEW Neural Network simulation.

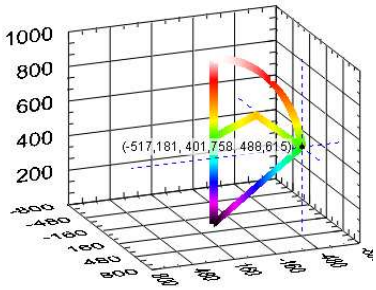

Fig. 11. The variation of the absolute coordinates function of the q_1, q_2 or q_3 determined coordinates from the neural network method.

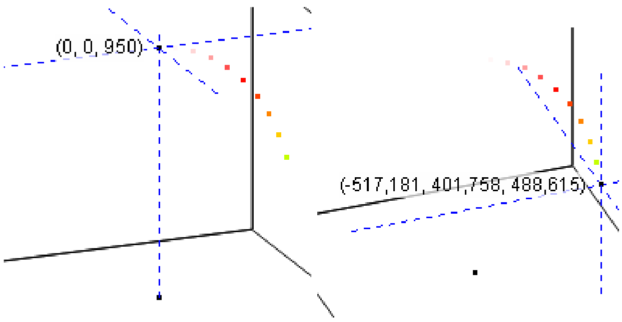

Fig. 12. First point and the last point of the commanded curve.

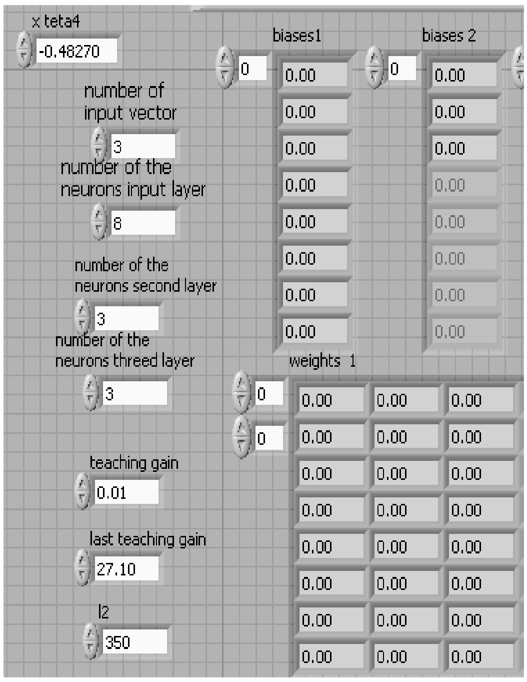

Fig. 13. Front panel with the input data concerning the number of neurons in each layer, teaching gains, initialized weights and biases matrices.

After the analysis of the results of the two study cases of the numerical simulation by using the proper neural network for the different target positions we can make the following remarks: the obtained values for internal robot coordinates verify the final imposed end effector position (target position) with 5% errors; all these errors are caused by the wrong choice the neuron numbers in each layer due to difficulty of calculation, wrong choice of the teaching gains and number of recurrent links, or position in the neural network schema of the time delay

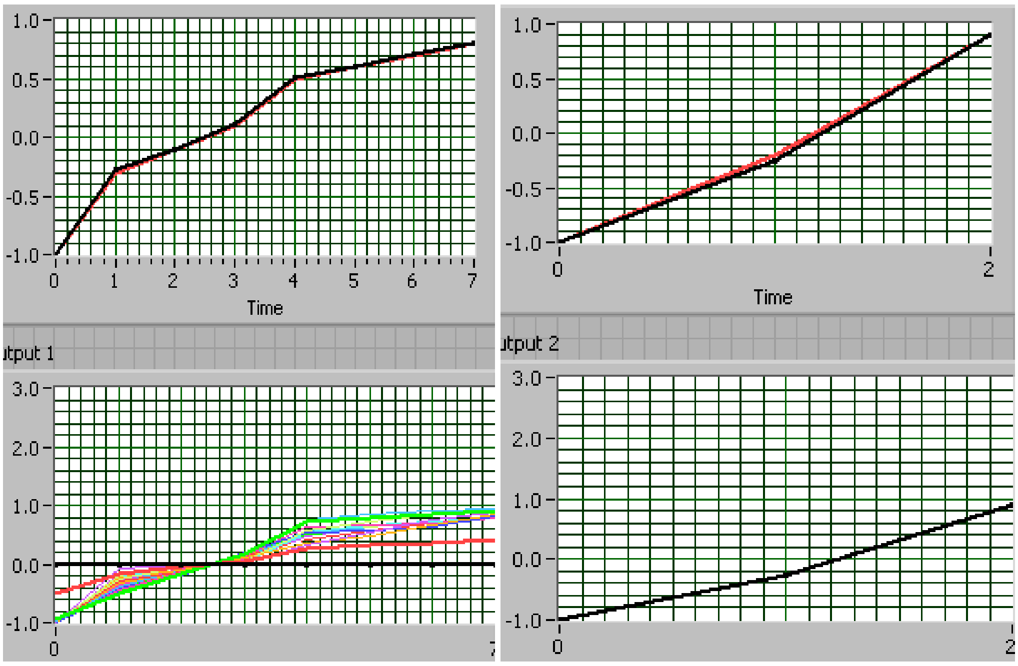

Fig. 14. The front panel of the VI with the target and obtained curves after applying the proper inverse kinematics neural network – case study 1.

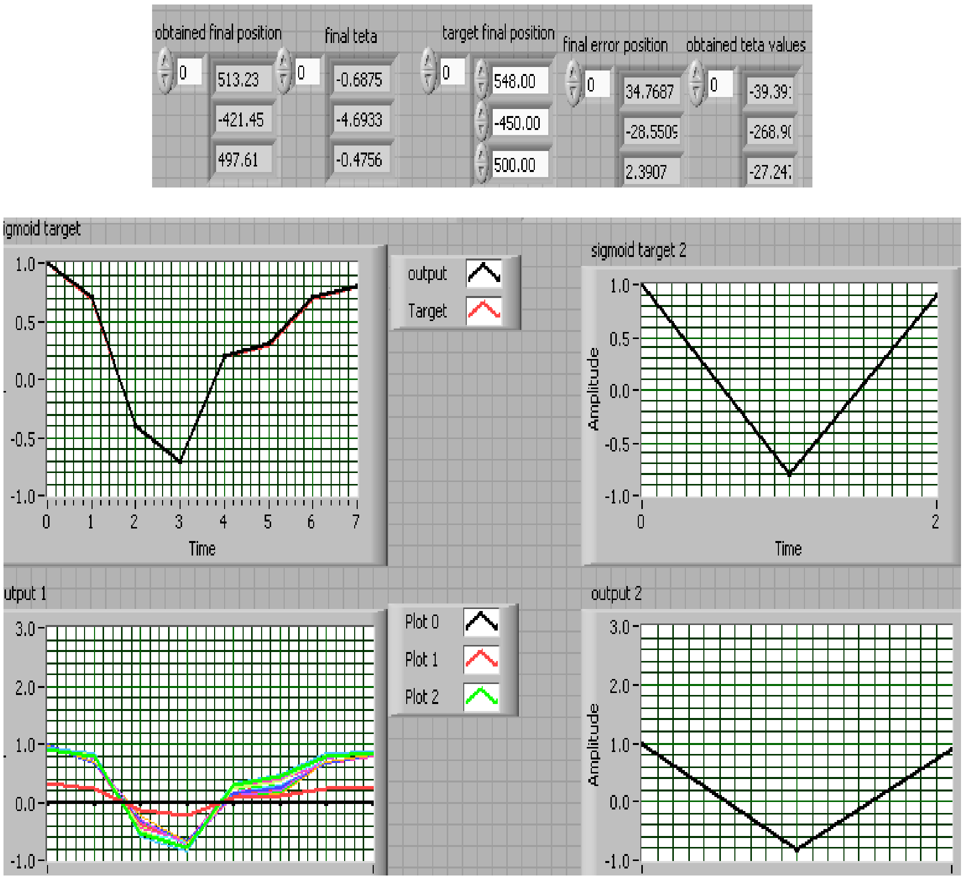

Fig. 15. The front panel of the VI with the target and obtained curves after applying the proper inverse kinematics neural network – case study 2.

and choice of the delay step. In the future work, all these influences will be studied for achieving the end-effector absolute position errors. Now, it is possible to use the obtained results in the control of the movement with inverse kinematics of the arm type robot for the rapid movement of the robot arm near the imposed final end-effector position, the extreme precision will be obtained with the incremental control in closed loop of all robot axes only for the last part of the way. By applying this method, the time of the movement will be shortened, dynamic behaviour improved, and control of the trajectory will be safer.

5. DISCUTION AND CONCLUSIONS

The paper showed one new neural method to obtain the relative robot coordinates knowing the data of the target absolute end-effector position. The research was made by applying this proposed method to one didactical arm type robot by some new virtual LabVIEW instruments. The theoretical research contents two ways:

- first, the analysis of the absolute 3D trajectory of one arm type robot, after applying the direct kinematics method with one proper LabVIEW VI and
- the second, the analysis and validation of the new proper Sigmoid Bipolar Hyperbolic Tangent Neural Network with many Time Delay and Recurrent Links, SBHTNN (TDRL). The second research way was made by comparing for two study cases the results after applying the internal robot coordinates, obtained from the neural network, in the direct kinematics VI, with the imposed absolute target of the end-effector position.

The research presented in the paper is a general approach and it can be used in many other applications when it is necessary to apply the optimization method. The future work will be the developing of the initial one using neural network to obtain the increasing of the precision more than it was obtained. The shown method will open the way to optimize the mobile and airplane robots trajectory and applying the smart materials and smart systems to the robotics field.

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, Vol. 3, 1994.

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

[3] D. Gorinevsky, T. Connolly, *Comparisons of some neural network and scattered data approximations: The inverse manipulator kinematic example,* Neural computation, Vol. 6, No. 3, pp. 521–542.

[4] L. Lee, *Training feedforward neural networks: an algorithm giving improved generalization,* Neural Networks, Vol. 10, No. 1, pp. 61–68, 1997.

[5] C. Schittenkopf, G. Deco and W. Brauner, *Two strategies to avoid overfitting in feedforward networks,* Neural Networks, Vol. 10, No. 3, pp. 505–516, 1997.

[6] A. Olaru, S. Olaru, L. Ciupitu, *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] A. Olaru, S. Olaru, D. Păune, A. Ghionea, *Assisted research of the neural network*, OPTIROB 2010 International Conference, Călimăneşti, Romania, The RPS Singapore Book, pp. 189–194, 2010.

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

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

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