KINETOSTATIC ANALYSIS OF A REDUNDANT SERIAL ROBOT

Mihai CRENGANIS[1], Anca Lucia CHICEA[1] and Octavian BOLOGA[1]

ABSTRACT: This paper presents aspects concerning the static forces analysis for a seven degrees of freedom robotic arm. This type of structure or robot can be used in military applications and in such operations, that require high degree of mobility for a robotic arm. For example, manipulation of objects in thigh spaces and when collision avoidance is needed. In the first part of the paper the mathematical model and the homogenous transformation matrixes that characterizes the robotic arm and the robotic structure are presented. Based on the transformation matrixes the entire homogenous matrix that characterizes the robot arm is computed and then the direct kinematics of the robot is solved. After obtaining the absolute position and orientation of the end effector the Jacobian matrix is computed. Using the Jacobian transpose matrix, the static analysis of forces is developed and then the mathematical model is validated using virtual kinematic models in MATLAB-Simulink, SimMechanics. The main advantage of the robot described in this paper is that it has in its kinematic structure an extra degree of freedom to position and orientate an object in the three-dimensional Euclidean space. In the end the virtual 3D robotic arm is presented.

KEY WORDS: redundancy, static forces, SimMechanics.

1 INTRODUCTION

With the continuous evolution of robotics and industrial automatization various mechanical systems that mimic living systems have been developed. Different structures of anthropomorphic robotic arms have been developed so far and they are becoming more and more technological advanced. A redundant robotic arm with more degrees of freedom (DOF) than necessary shows great potential because it gains a vast working space and a high dexterity and therefore it can be used in different industrial fields like automotive, aerospace industry and so on. Firstly, the paper presents the mechanical structure of the serial robot and then it is highlighted the modeling and the simulation of the direct kinematics of a robotic arm with seven DOF. The Jacobian matrix is then obtained and using the Jacobian transpose the static forces are computed. Finally, the results of the simulations are compared with the numerical analysis and then the conclusions are expressed.

2 THE M.I.E DEP. ROBOTIC ARM

To characterize certain properties of a robotic arm, there can be considered different parameters that can express the degrees of mobility, the magnitude and dimensions of robot movements.

[1] "Lucian Blaga" University from Sibiu, Engineering Faculty, Str. Emil Cioran, No. 4, 550025 Sibiu, România
E-mail:mihai.crenganis@ulbsibiu.ro;
anca.chicea@ulbsibiu.ro, octavian.bologa@ulbsibiu.ro

We can consider that the robot used in this paper represented in simplified form, acquires three DOF for the shoulder joint, one DOF for the elbow joint and three DOF for the hand wrist. A 3D CAD model of the robot structure is shown in Figure1.

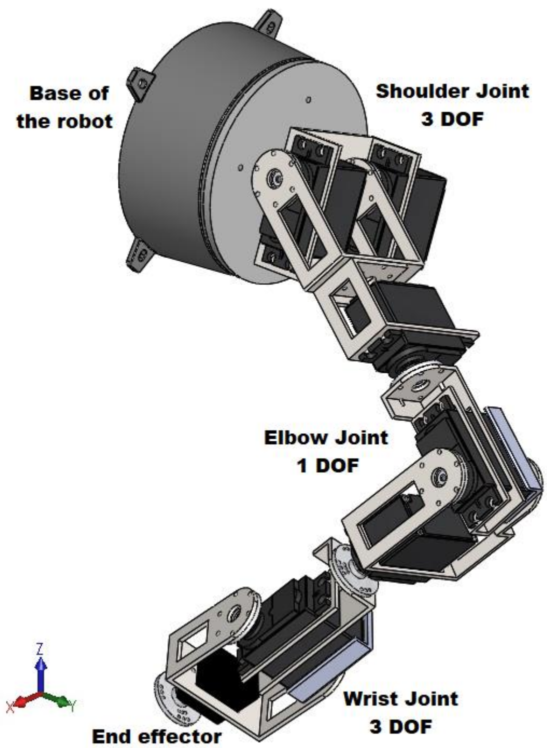

Figure 1. M.I.E Dep. robot arm

3 THE KINEMATICS OF THE ROBOT

If we attach to each element "i", (i = 0…7) of the structure, one fixed coordinate system $k_i(O_i, x_i, y_i, z_i)$, then we can express the homogeneous transfer matrices A_i which characterize the relative movements between each element of the mechanic structure. If we know the relative parameters θ_i (i = 1 …7) and the homogeneous transfer matrix form between two elements or the homogeneous transfer matrix between the coordinate systems attached to each element, we can determine the total transfer matrix between the system $k_7(O_7, x_7, y_7, z_7)$ and system $k_0(O_0, x_0, y_0, z_0)$.

$$H_{07} = A_1 \cdot A_2 \cdot A_3 \cdot A_4 \cdot A_5 \cdot A_6 \cdot A_7 \quad (1)$$

were:

$A_1 = R_y(\theta_1), A_2 = R_x(\theta_2), A_3 = R_z(\theta_3) \cdot T_z(L1),$

$A_4 = R_y(\theta_4) \cdot T_z(L2), A_5 = R_z(\theta_5), A_6 = R_x(\theta_6),$

$A_7 = R_y(\theta_7) \cdot T_y(L3)$

In figure 2 the kinematic structure of the robotic arm is presented.

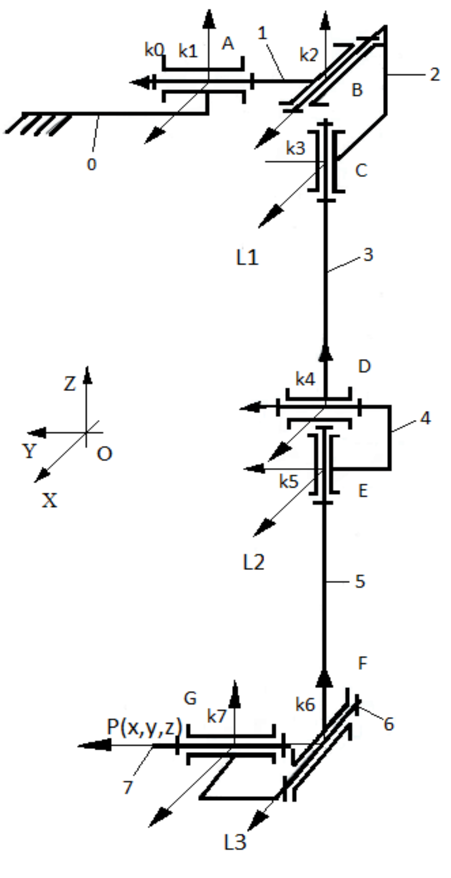

Figure 2. Kinematic structure of the robot arm

$$A_1 = \begin{pmatrix} cos(\theta_1) & 0 & sin(\theta_1) & 0 \\ 0 & 1 & 0 & 0 \\ -sin(\theta_1) & 0 & cos(\theta_1) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (2)$$

$$A_2 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & cos(\theta_2) & -sin(\theta_2) & 0 \\ 0 & sin(\theta_2) & cos(\theta_2) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (3)$$

$$A_3 = \begin{pmatrix} cos(\theta_3) & sin(\theta_3) & 0 & 0 \\ -sin(\theta_3) & cos(\theta_3) & 0 & 0 \\ 0 & 0 & 1 & L1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (4)$$

$$A_4 = \begin{pmatrix} cos(\theta_4) & 0 & sin(\theta_4) & sin(\theta_4) \cdot L2 \\ 0 & 1 & 0 & 0 \\ -sin(\theta_4) & 0 & cos(\theta_4) & cos(\theta_4) \cdot L2 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (5)$$

$$A_5 = \begin{pmatrix} cos(\theta_5) & sin(\theta_5) & 0 & 0 \\ -sin(\theta_5) & cos(\theta_5) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (6)$$

$$A_6 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & cos(\theta_6) & -sin(\theta_6) & 0 \\ 0 & sin(\theta_6) & cos(\theta_6) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (7)$$

$$A_7 = \begin{pmatrix} cos(\theta_7) & 0 & sin(\theta_7) & 0 \\ 0 & 1 & 0 & L3 \\ -sin(\theta_7) & 0 & cos(\theta_7) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (8)$$

We will note the elements of H_{07} like this:

$$H_{07} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{pmatrix} \quad (9)$$

3.1 Direct kinematics equations

The direct kinematic analysis of positions is based on the relationship between kinematic joints, the relative movements from the kinematic joints, or the robotic arm joint space, and the position and orientation of the tool or end effector. From a formal point of view, the direct kinematic problem deals with the determination of the position and orientation of the end effector if the variation of relative displacements of the kinematic joints is known. In the present case, for the robotic arm assembly, the relative displacements in the kinematic joints are angular movements or rotations, each kinematic coupling acquires a single degree of freedom. The direct kinematics task is to determine the cumulative effect of all joint movements. If the H_{07} matrix is computed, then the position and orientation of the end effector are determined using the next equations:

$$X = a_{14} \quad Y = a_{24} \quad Z = a_{34} \qquad (10)$$

$$\varphi_x = \arctan2(a_{23}, a_{33}) \quad \varphi_y = -\arcsin(a_{13}) \quad \varphi_z = \arctan2(a_{12}, a_{11}) \qquad (11)$$

where: *X*, *Y*, *Z* represents the position of the end effector and $\varphi_x, \varphi_y, \varphi_z$, represents the orientation of the end effector. The direct problem of velocities is formulated as follows: considering the laws of variation of the generalized velocities $\dot{q} = (\dot{q}_1 \ldots \dot{q}_n)^T$or the relative movements between the elements $q = (q_1 \ldots q_n)^T$, we want to determine the absolute velocities of the characteristic point or the final effector velocities $\dot{X} = (\dot{x},\ \dot{y},\ \dot{z}, \dot{\varphi}_x, \dot{\varphi}_y, \dot{\varphi}_z)^T$. The relation between the relative and the absolute velocities is linear, unlike the one between the relative and absolute positions which is non-linear. If the relative speed of two elements increases twice then the absolute speed will be twice as high [3]. Thus, it can be expressed by the relationship:

$$\dot{X} = J(q) \cdot \dot{q} \qquad (12)$$

where: $\dot{X}$ represents the absolute velocity of the end effector, *J(q)* is the Jacobian matrix, *q* represents the joint space.

$$\bar{J} = \frac{\partial f(q)}{\partial q_i} \qquad (13)$$

In this case, the manipulator is redundant, so *J* is not of the square size but is of the 6x7 matrix.

$$J = \begin{bmatrix} J_{11} & J_{12} & J_{13} & J_{14} & J_{15} & J_{16} & J_{17} \\ J_{21} & J_{22} & J_{23} & J_{24} & J_{25} & J_{26} & J_{27} \\ J_{31} & J_{32} & J_{33} & J_{34} & J_{35} & J_{36} & J_{37} \\ J_{41} & J_{42} & J_{43} & J_{44} & J_{45} & J_{46} & J_{47} \\ J_{51} & J_{52} & J_{53} & J_{54} & J_{55} & J_{56} & J_{57} \\ J_{61} & J_{62} & J_{63} & J_{64} & J_{65} & J_{66} & J_{67} \end{bmatrix} \qquad (14)$$

The Jacobian matrix is determined using the MATLAB function: J=jacobian([X, Y, Z, φ_x, φ_y, φ_z],[*q1, q2, q3, q4, q5, q6, q7*]); The Jacobian transpose matrix is determined using the MATLAB function JT=transpose(J);

3.2 Static forces analysis

The kinetostatic analysis studies the relationships between the forces and torques acting in the robot joints and the corresponding momentum that acts on the end effector. Knowing the forces and torques that act on the end effector one should determine the forces and torques that need to be developed in the kinematic joints. The mathematical equation that connects the motor torque needed at the joint level and the force acting the end effector level is:

$$M = J^T(q) \cdot F \qquad (13)$$

where: $M=(M1, M2, M3, M4, M5, M6, M7)^T$, are joint torques;

$F=(Fx, Fy, Fz, Mx, My, Mz)^T$ are the forces and torques acting at the end effector.

3.3 Testing the mathematical equations

To obtain numerical results for the kinetostatic analysis the mathematical equations presented above, were written in MATLAB ® as cod.m then implemented in Simulink® figure 3. The implementation was done in an Embedded MATLAB Function block. Numerical results for kinetostatic analysis were obtained for the following:

- the values of joint space:

$$q_1 = 10^\circ, q_2 = 20^\circ, q_3 = 30^\circ, q_4 = 10^\circ,$$
$$q_5 = 20^\circ, q_6 = 10^\circ, q_7 = 10^\circ$$

- the values for the forces and torques that acts on the end effector:

Fx = 10N, *Fy* = 10N, *Fz* = 10N,

Mx=0Nm; *My*=0Nm; *Mz*=0Nm.

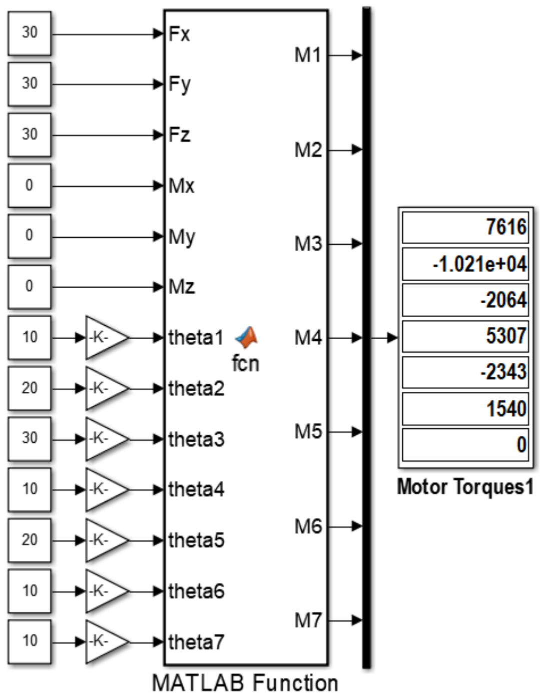

Figure 3. Mathematical equations numerical results

3.4 Forces analysis using virtual models

During the analysis, to validate the mathematical equations, a three-dimensional model of the robotic arm was created. Thus, with Simulink - SimMechanics environment, three-dimensional virtual models of the robotic arm were realized. These were first created in the SolidWorks CAD environment and then imported into the SimMechanics environment. At the same time, using SimMechanics environment, another method of solving the kinetostatic problem of the robotic arm was used, namely, by actuating the virtual kinematic elements of the robotic arm with the virtual actuators placed in the kinematic joints, various information related to the position and orientation using virtual sensors placed either in kinematic joints or attached to the final effector were extracted. The SimMechanics-Simulink environment operating mechanism is based on vector equation systems and on homogeneous transformation matrices, to which the user has access only to certain control parameters. The SimMechanics environment is also based on a numerical, geometric analysis to solve the kinetostatic problem of the robotic arm. In figure 4 the kinetostatic analysis model block based on SimMechanics is presented.

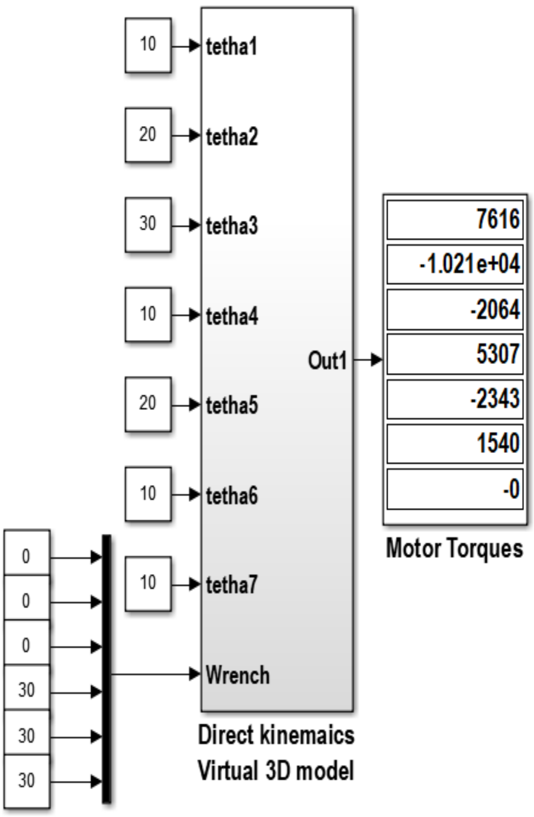

Figure 4. Results for SimMechanics analysis

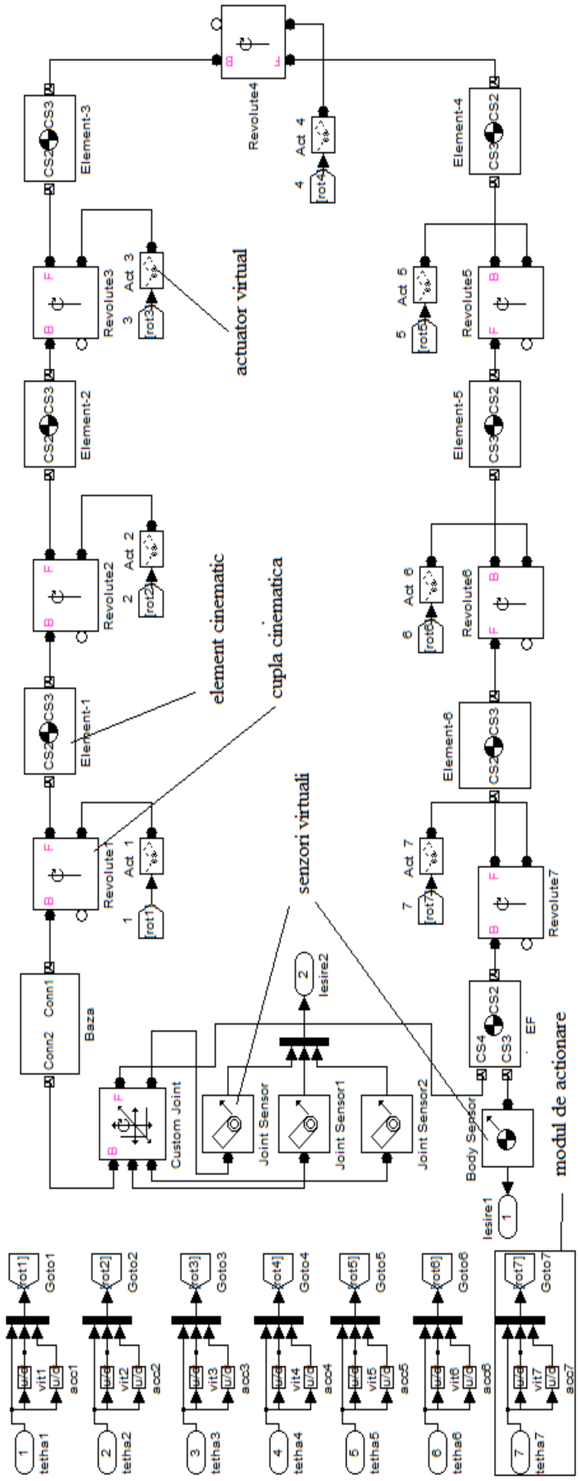

Figure 5. SimMechanics block diagram

To build the SimMechanics forces analysis corresponding model we have used a Body Block to actuate the end effector with the desired forces and torque. The Body Actuator block actuates a Body block with a generalized force signal, representing a force/torque applied to the body:

- Force for translational motion;
- Torque for rotational motion;

The generalized force is a function of time specified by a Simulink® input signal. This signal can be any Simulink signal, including a signal feedback from a Sensor block. The Body Actuator applies the actuation signal in the reference coordinate system (CS) specified in the block dialog. The import is the Simulink input signal. The output is the connector port you connect to the Body block you want to actuate. In figure 6 a body actuator is presented. To obtain numerical results in every kinematic joint we used a joint sensor to read the computed torques. A joint sensor is presented in figure 8.

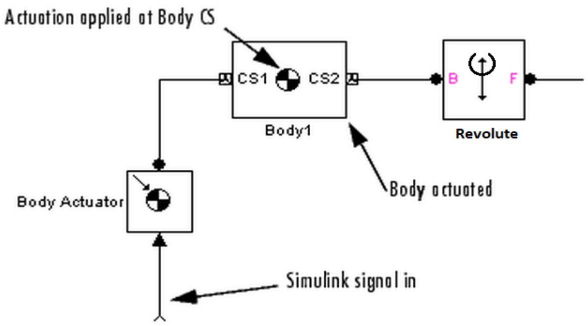

Figure 6. Body Actuator connected to a Body

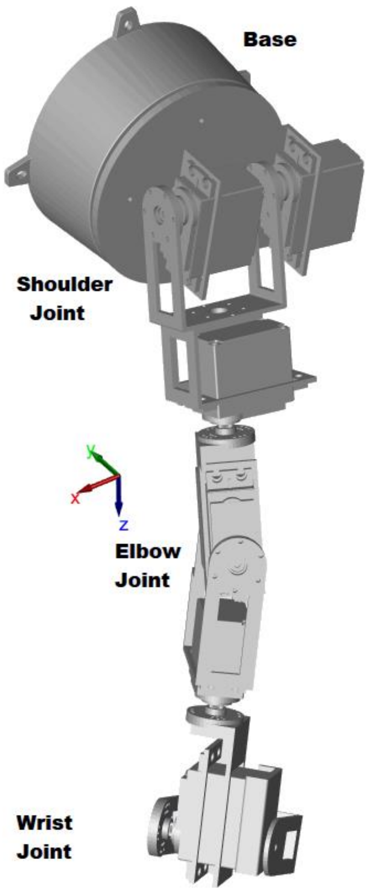

Figure 7. SimMechanics 3D virtual model

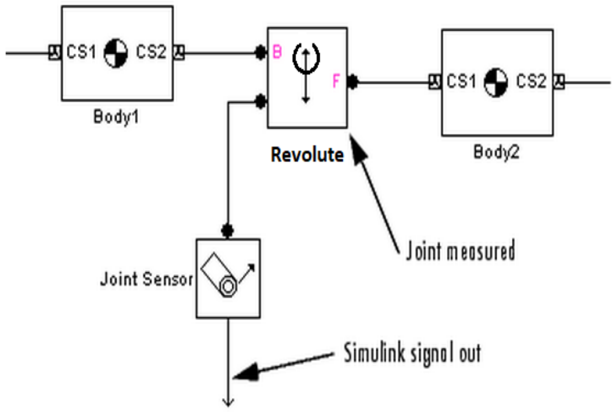

Figure 8. A joint sensor connected to a joint

The Joint Sensor block measures the position, velocity, and/or acceleration of a joint primitive in a Joint block. It also measures the reaction force and torque across the Joint. The Joint Sensor measures the motion along/about the joint axis (or about the pivot point for a spherical primitive) in the reference coordinate system (CS) specified for that joint primitive in the Joint's dialog. The Joint connects a base and a follower Body at one body coordinate system on each body. The base-follower sequence determines the sense of the motion, which is defined as follower relative to base. The reaction force and torque are three-component vectors of the force and torque that the joint transfers from the base Body to the follower Body. The computed force is the component of the reaction force projected along the prismatic primitive axis. It is also the force along the prismatic axis that reproduces the follower motion with respect to the base.

The computed torque is the component of the reaction torque projected along the revolute primitive axis. It is also the torque about the revolute axis that reproduces the follower motion with respect to the base. After performing robot kinematic simulations and static analysis of forces based on mathematical equations and on virtual 3D models, it has been established that both models are correct, efficient and easy to integrate into real-time simulation models. By realizing the simulations based on the SimMechanics three-dimensional robotic model, it was concluded that kinetostaic resolution is possible even if the kinematic model equations are not known. As a result, using the SimMechanics work environment, it is possible to solve the kinematics and kinetostatics to make even more complex mechanical assemblies, whose mathematical model would be very difficult to be developed. The results obtained from kinetostatic simulations based on three-dimensional virtual models are identical to those resulting direct from the kinetostatic simulations based on mathematical equations. The main advantage of using theSimMechanics model block is that one can even analyze the dynamics of an entire robot. In the case of dynamics, the problem is more complex in terms of modeling the whole assembly mathematically. The above presented analysis does not consider the weight of kinematic elements. The analysis presented, and the developed models illustrates only the effect of the forces acting in the kinematic joints onto the forces that occur at the end effector. In figure 9 the results obtained from considering the weight of the robotic arm elements is presented.

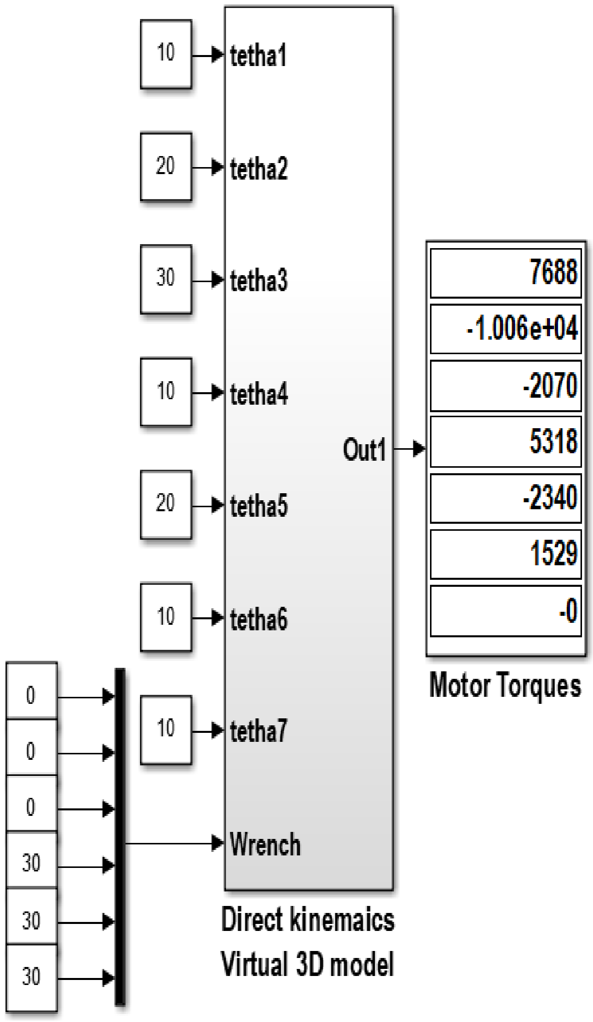

Figure 9. Results for SimMechanics analysis with weight of the elements taken into account

4 CONCLUDING REMARKS

The developed kinematics equations were validated using virtual simulations. The robotic arm did not show the impossibility of moving for any joint motion.

Simulations have demonstrated the effectiveness of the proposed method for solving the kinetostatic problem for an anthropomorphic robotic arm.

During the analysis, to validate the mathematical equations, a three-dimensional model of the robotic arm was created. Thus, with Simulink - SimMechanics environment, three-dimensional virtual models of the robotic arm were realised. These were first created in the SolidWorks CAD environment and then imported into the SimMechanics environment. After performing robot kinematic simulations and static analysis of forces based on mathematical equations and on virtual 3D models, it has been established that both models are correct, efficient and easy to integrate into real-time simulation models. By realizing the simulations based on the SimMechanics three-dimensional robotic model, it was concluded that kinetostaic resolution is possible even if the kinematic model equations are not known. As a result, using the SimMechanics work environment, it is possible to solve the kinematics and kinetostatics to make even more complex mechanical assemblies, whose mathematical model would be very difficult to be developed. The results obtained from kinetostatic simulations based on three-dimensional virtual models are identical to those resulting direct from the kinetostatic simulations based on mathematical equations.

5 REFERENCES

►Angeles, J., *Fundamentals of Robotic Mechanical Systems: Theory, Methods, and Algorithms*, 3rd Edition, Springer- Verlag, New York, 2006.

►Antonelli, G. Indiveri, G. and Chiaverini, S., *Prioritized closed loop inverse kinematic algorithms for redundant robotic systems with velocity saturations*, Proceedings of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 5892–5897, St. Louis, USA, Oct 2009.

►Carignan, C., *Trajectory optimization for kinematically redundant arms*, Journal of Robotic Systems, Vol. 6, No. 5, 221–248, 1993.

►Craig, J., *Introduction to robotics*, Reading, MA: Addison-Wesley, 1989.

►Deepak, T., *Real-Time Inverse Kinematics of the Human Arm*, Presence. Vol. 5. No.4., 393-401 by the Massachusetts Institute of Technology, 1996.

►Howerd, D., *Application of Fuzzy Logic for the Solution of Inverse Kinematics and Hierarchical Controls of Robotic Manipulators*, Journal of

Intelligent and Robotic Systems 23: 217–247, 1998.
►Klein, C., Hung, C. H., *Review of pseudoinverse control for use with kinematically redundant manipulators*, IEEE Trans. on Systems, Man, and Cybernetics, Vol. 13, 245–250,
►Sciavicco L. and Siciliano, B., *A solution algorithm to the inverse kinematic problem for redundant manipulators*, IEEE J. of Robotics and Automation, Vol. 4, No. 4, 303–310, 1998.
►Siciliano, B. and Bastin, G., *Theory of Robot Control*, Springer Verlag, London, U.K., 1996.
►MathWorks Suport-Fuzzy Logic,
►http://www.mathworks.com/help/html.