Journal of the University of Chemical Technology and Metallurgy, 43, 1, 2008, 159-164

DEVELOPMENT OF A 3D AND VRML VIRTUAL HAND MODELS FOR DIFFERENT MECHANICAL GRIPPER

Y. Tsepkovskiy[1], L. Antonov[2], Cv. Kocev[2], F. Palis[1], N. Shoylev[2]

[1] *Otto-von-Guericke-Universität Magdeburg, Institut für Elektrische Energiesysteme 39016 Magdeburg, Germany*
[2] *University of Chemical Technology and Metallurgy, 8 Kl. Ohridski, 1756 Sofia, Bulgaria*

Received 05 Septenber 2007
Accepted 12 January 2008

ABSTRACT

In the following article development of a 3D and VRML virtual hand models for different type of mechanical gripper is shown. A 3D animation using Matlab Simulink is made. The tasks of forward and inverse kinematics are considered with their equations in analytic and matrix form.

Keywords: *3D CAD model, mechanical gripper, forward kinematics, inverse kinematics.*

INTRODUCTION

At present robotics, medicine and nanotechnologies take central places in world science researches. In the field of robotics study, the tasks for development and control of complex manipulators take central place. As an example of a such manipulator can be considered a virtual robotic hand, which can be used for development of human prosthesis, humanoid robots and monitoring systems.

EXPERIMENTS

Development of a graphic 3D model with seven degrees of freedom

The reason to start with a CAD model of the hand is that in this model can be set geometric dimensions between the axes in 3D dimension, dimensions between separate bodies and documentation of any separate detail of the model can be prepared. There is a variety of software products for development of a 3D graphic virtual hand model (Autocad, Proingeneer, Archicad, Solid works). At Fig.1 is shown a model, created by using Solid works. The main reason for using this software product is that there is an opportunity to convert the model in VRML (Virtual Reality Model Language) format. The block for virtual reality in Matlab Simulink program area is shown at Fig. 2. VRML format has a tree hierarchical structure and each component in this tree is described by a separate tree brunch. By indexing the main component of the tree model selection of the needed tree components‘ characteristics can be done. As a result of this action there is an opportunity to change the surrounding area of the model in time. It is possible, as an example, to create 3D animation in real computational time, which may not match with the present real time if the model requires vastly computational time for every loop [1].

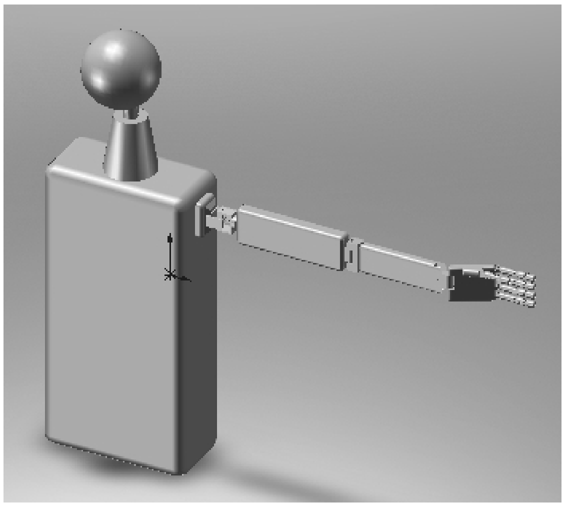

Fig. 1. Graphical 3D virtual hand model.

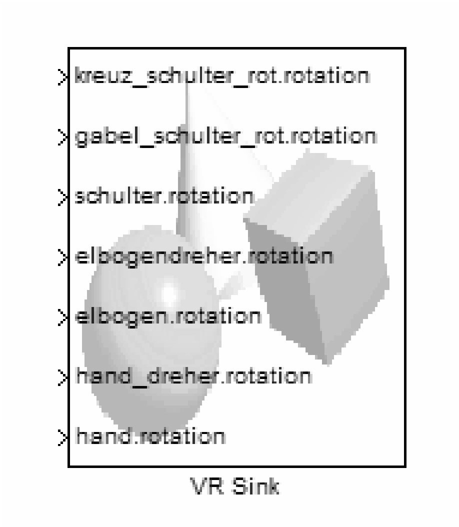

Fig. 2. Virtual Reality Block in Matlab.

The software product Matlab is suitable for development of different mechanical control systems in Simulink, which are concerned about the force robot part. In Simulink environment is intended a toolbox for modeling of dynamic systems called Sim Mechanics. This allows to the model to make animations by complex mechanical system and on the base of calculations at the same time. The whole hand with 5 fingers from the shoulder to the last finger phalange has 26 degrees of freedom – 3 in the shoulder articulation (rotating around axes OX, OY and OZ), 1 in the elbow articulation (rotating around axe OY), 3 in the hand articulation (rotating around axes OX, OY and OZ) and the rest degrees of freedom are in the wrist.

Creating a 3D animations for different mechanical hand gripper

After the 3D model is connected with Matlab program area a 3D animation of different mechanical hand gripper is made; the Simulink scheme is shown at Fig. 3. 1. All kind of hand movements are presented as a matrix and any different kind of movement is a row of this matrix. For easy demonstrations of hand movements there is a block with some mechanical switches. With them a random row from the matrix can be chosen manually and the result is realization of mechanical movement (Fig. 3. 2). After the matrix multiplexing there is a control system, including a gain, integrator and negative feedback. The gain is a proportional position regulator with coefficient 1. The value of the gain sets the value of feeding control error speed. After it there is an integrator for simple dynamic model of translation process. Next a demultiplexing device is shown, which separates the signal to 7 seven signals. These signals are set to the seven outputs of the model (kreuz_schulter_rot, gabel_schulter_rot, elbogendreher etc). There are additional blocks for every model output, setting the rotation axe in 3D space for each arm model output, shown in square brackets. The first number is set for axe OX, the second for axe OY and the third – for OZ axe. The model animations of hand mechanical gripper for catching bodies with cylindrical shape and three fingers catch are given at Fig. 4.1 and 4.2 respectively.

Rigid transformations

For the task of robots movements description the possible positions of a rigid body is needed. In most cases industrial robots are used for moving of objects with solid mass and capacity from one place to another. Based on this fact a way for description of any rigid body subsequent position is needed. The rigid transformation is a suitable computational approximation for description of any moving. Rigid body transformations are characterized by the fact, that they preserve the distance between any two points in the body [2].

Rotations are rigid transformations, which do not distort the size or shape of the body. In 3-dimentional space a rotation can be presented by the following matrixes:

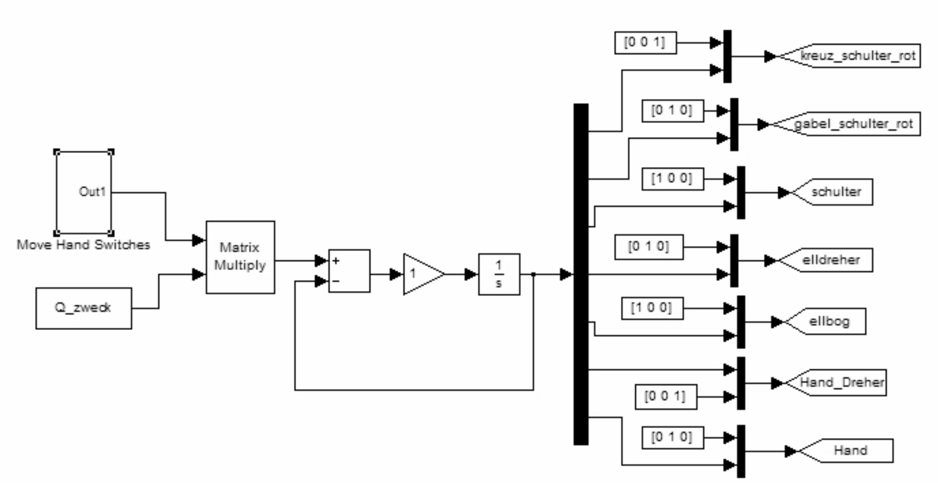

Fig. 3. 1. Simulink scheme of the 3D animation hand model.

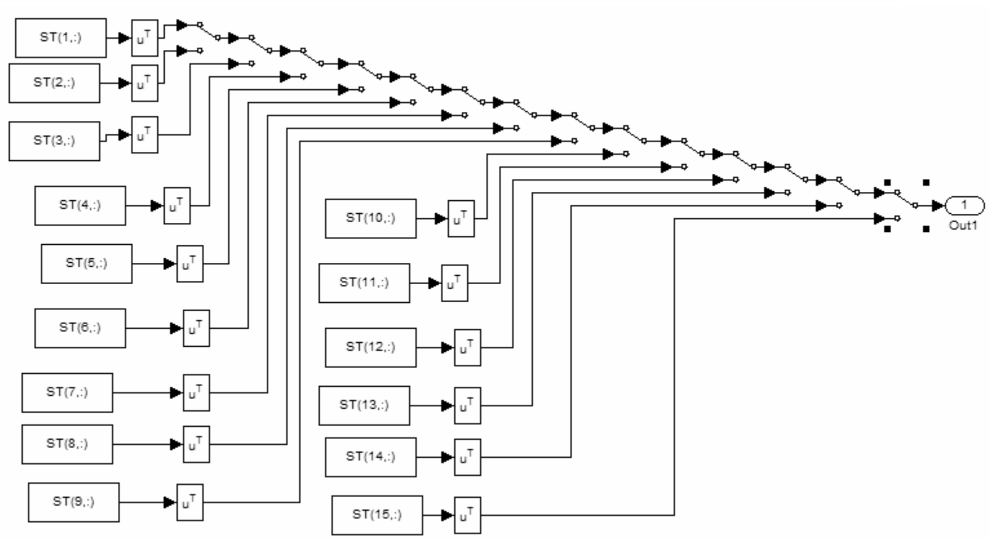

Fig. 3.2. Switches for different type of mechanical hand gripper realization.

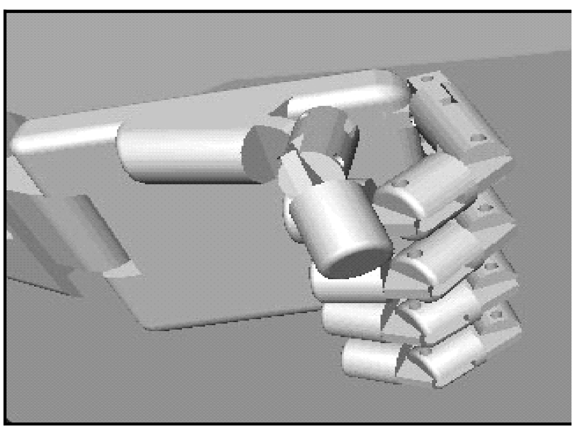

Fig. 4.1. Gripper for catching bodies with cylindrical shape.

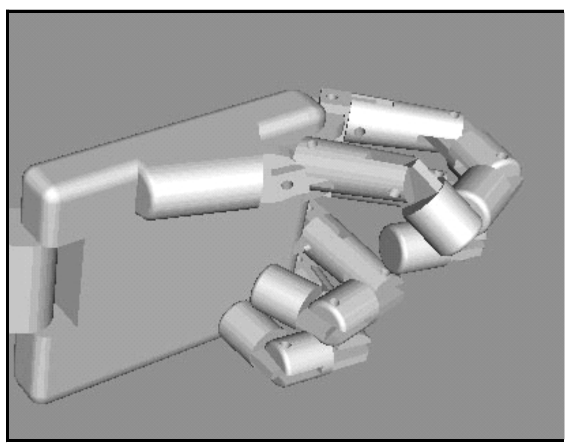

Fig. 4.2. Gripper for three fingers catch.

$$Rx_1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) & 0 \\ 0 & \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

- Rotation around axe OX.

$$Ry_1 = \begin{pmatrix} \cos(\varphi) & 0 & \sin(\varphi) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(\varphi) & 0 & \cos(\varphi) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

- Rotation around axe OY

$$Rz_1 = \begin{pmatrix} \cos(\theta) & -\sin(\theta) & 0 & 0 \\ \sin(\theta) & \cos(\theta) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

- Rotation around axe OZ,

where α is the rotation angle around axe OX, φ – around axe OY and θ – around axe OZ.

The entire rotation in 3D space can be described my multiplying the rotation matrixes around each axe: $Rrot_1=Rx_1Ry_1Rz_1$

Translations are also rigid transformations as they preserve the distance between any pair of points. Translation matrix has the following form;

$$T_1 = \begin{pmatrix} 1 & 0 & 0 & x_1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix},$$

where x_1 is the length of the first hand model element

Forward kinematics

Kinematics is the study of possible movement and configurations of a system, depending of the geometric size of the system and defining the last link coordinates dependency of the changeable degrees of freedom [3, 4, 5].

The forward kinematics concerns the position and orientation of the last link (end-effector) of the robot in the terms of the joint variables. The developed model has 7 degrees of freedom – 3 in the shoulder articulation, 1 in the elbow articulation and 3 in the wrist. In this article only 5 of these degrees of freedom are considered – 3 in the shoulder articulation, 1 in the elbow articulation and 1 in the wrist.

The matrixes, describing the movement of the mentioned above articulations, have the following form:

For the shoulder articulation:

$$R_1(\phi_z, k) = \begin{vmatrix} \cos\phi_z & -\sin\phi_z & 0 \\ \sin\phi_z & \cos\phi_z & 0 \\ 0 & 0 & 1 \end{vmatrix}$$

$$R_2(\phi_y, j) = \begin{vmatrix} \cos\phi_y & 0 & \sin\phi_y \\ 0 & 1 & 0 \\ -\sin\phi_y & 0 & \cos\phi_y \end{vmatrix}$$

$$R_3(\phi_x, i) = \begin{vmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_3 & -\sin\theta_3 \\ 0 & \sin\theta_3 & \cos\theta_3 \end{vmatrix}$$

For the elbow articulation:

$$A_1(\theta_1) = \begin{vmatrix} \cos\theta_1 & -\sin\theta_1 & (1-\cos\theta_1).l_1 \\ \sin\theta_1 & \cos\theta_1 & -\sin\theta_1.l_1 \\ 0 & 0 & 1 \end{vmatrix}$$

For the wrist:

$$A_2(\theta_2) = \begin{vmatrix} \cos\theta_2 & -\sin\theta_2 & (1-\cos\theta_2).(l_1+l_2) \\ \sin\theta_2 & \cos\theta_2 & -\sin\theta_2.(l_1+l_2) \\ 0 & 0 & 1 \end{vmatrix}$$

where l_1 is the length of the armpit and l_2 – the length of the elbow.

$$(\alpha, \beta, \gamma, x, y, z) = K(_1, _2, _3, \theta_1, \theta_2) \qquad (1)$$

where α, β, γ – axes angels in relatively coordinate system; x, y, z – gripper coordinates in global coordinate system.

The general equation for the forward kinematics can be written in the analytic form below:

$$K(_1, _2, _3, \theta_1, \theta_2) = R_1(_1,k)R_2(_2,j)R_3(_3,i)A_1(\theta_1)A_2(\theta_2) \quad (2)$$

Inverse kinematics

The inverse kinematics concerns the values of joint angels if the position and orientation of the gripper has the desired values and the instant or initial position are known [1, 4]. This task has many solutions, so finding the joint angels for each manipulator is a complex iterative procedure. When stated mathematically, the problem reduces to solving a system of multivariate equations [2]. Essentially the following matrix equation must be solved:

$$K(\phi_z, \phi_y, \phi_x, \theta_1, \theta_2) =$$
$$= R_1(\phi_z)R_2(\phi_y)R_3(\phi_x)A_1(\theta_1)A_2(\theta_2) \quad (3)$$

K is the constant matrix which specifies the position and orientation of the gripper. This constitutes a set of highly non-linear equations for the joint angels $\phi_z, \phi_y, \phi_x, \theta_1, \theta_2$.

RESULTS AND DISCUSSION

In this article the Newton-Raphson method is applied for the 3D model with mentioned above 5 degrees of freedom. The main idea of this method is solving an equation with the following form:

$$f(x) = 0 \quad (4)$$

The iterative procedure for reaching a final decision is:

$$x^{(i+1)} = x^i - \frac{f(x^{(i)})}{\frac{d}{dx} f(x^{(i)})} \quad (5)$$

In our case the following equations can be written

$$\begin{aligned} f_1(\theta_1 ... \theta_5) &= 0 \\ f_2(\theta_1 ... \theta_5) &= 0 \\ f_3(\theta_1 ... \theta_5) &= 0 \end{aligned} \quad \Rightarrow \quad f(\theta) = 0 \quad (6)$$

Using the Tailor's theorem we have:

$$f(\theta + h) \approx f(\theta) + J(\theta).h$$
$$h \approx J^{-1}(\theta).f(\theta + h)$$
$$\theta^{(i+1)} = \theta^{(i)} - J^{-1}(\theta^{(i)}).f(\theta^{(i)}) \quad (7)$$

where J is the Jacobean of the manipulator kinematics matrix; h - error; θ - joint angle (vector of angels) representing the solution of the inverse kinematics task about the manipulators.

In this article for stop criteria is used the error between the desired coordinates of end-effector and current coordinates after joint angles calculation.

For desired coordinates is set the following vector:

$$\begin{bmatrix} 4 \\ 0 \\ -5 \end{bmatrix}$$

The error is set to a value of 0.1 m for 132 iterations. The lengths of the arm links have the following values:

7 m for the link between shoulder articulation and elbow articulation;

5 m for the link between elbow articulation and the wrist;

3 m for the wrist link.

The following joint angles are calculated as a final result [0; 24,7; 116,6; 147,3; 75,8] , where

0^o – for the first degree of freedom of the shoulder articulation rotating around axe OZ;

$24,7^o$ - for the second degree of freedom of the shoulder articulation rotating around axe OY;

$116,6^o$ - for the third degree of freedom of the shoulder articulation rotating around axe OX;

$147,3^o$ - for the only degree of freedom of the elbow articulation rotating around axe OY;

$75,8^o$ - for the degree of freedom of the wrist rotating around axe OY.

For these values of joint angles the following coordinates of hand end-effector are calculated:

$$\begin{bmatrix} 4.0745 \\ 0 \\ -4.936 \end{bmatrix}$$

The error is calculated by the following way:

$$Er = |Cd - Cc| \tag{8}$$

where
Er – error in relative dimension
Cd – desired coordinate around the respectively axe
Cc - current coordinate around the same axe

So, in this case the errors around axes OX, OY and OZ have the following values:

Around axe OX: $Er = |4 - 4.0745| = 0.0745m$
Around axe OY: $Er = |0 - 0| = 0m$
Around axe OZ: $Er = |-5 - (-4.936)| = 0.064m$

The only disadvantage of this method is that it concerns no joint angles restrictions.

CONCLUSIONS

The question of development of three-dimensional robot hand model with 5 degrees of freedom from the shoulder to the wrist and 19 degrees of freedom in the wrist is considered with it animations in the environment of Matlab. The attention is paid as to features of importing CAD models in Matlab and their connection with Simulink environment, also to kinematics problems and a way of their solutions. All this gives opportunity to use Matlab for modeling of mechanical devices, creating control systems by similar systems and to modeling the mechanical behavior of the model.

Acknowledgements

This work is supported by the DAAD PPP-program 06/2007 between Otto-von-Guericke-University Magdeburg, Germany, and University of Chemical Technology and Metallurgy, Sofia, Bulgaria.

REFERENCES

1. Virtual Reality Toolbox Users Guide.
2. J. M. Selig, “Introductory robotics”, Prentice Hall International (UK), 1992.
3. D. Manocha, J. Canny, “Efficient Inverse Kinematics for General 6R Manipulators”, IEEE Transactions on robotics and automation, **10**, 5, October 1994.
4. D. Tolani, A. Goswami, N. Badler, “Real Time Inverse Kinematics Techniques for Anthropomorphic Limbs”, Computer and information science department, University of Pennsylvania, Philadelphia, Pennsylvania, 19104-6389, 15 August 2000.
5. W. Griffin, R. Findley, M. Turner, R. Cutkosky, “Calibration and Mapping of a Human Hand for Dexterous Telemanipulation”, Haptic Interfaces for Virtual Environments and Teleoperator System Symposium, ACME IMEGE 2000.