

A CONTRIBUTION TO THE DYNAMIC SIMULATION OF ROBOT MANIPULATOR WITH THE SOFTWARE ROBOTDYN

NGUYEN VAN KHANG, DO THANH TRUNG
Hanoi University of Technology

ABSTRACT Robots manipulators are multibody systems with tree structures. In this paper the theoretical background of the computer program RobotDyn is presented. The program is developed using Lagrange's equations and Denavit-Hartenberg matrix. The dynamic simulation of the Robot SCARA with four degrees of freedom is considered as an example of application of the program RobotDyn.

1 Introduction

The simulation of robotic manipulators is a very important issue. Dynamic simulation is usually performed in two steps:

- (1) generation of the dynamics model and
- (2) solution of the model.

Since the dynamics equations of robot manipulator are highly nonlinear and coupled, the analytical solution of such equations is thought to be impossible. Hence numerical approaches have to be used. The development of the computer-oriented methods for the generation of dynamic robot models can roughly be divided into three groups:

1. Numerical methods,
2. Numerical-symbolic method,
3. Symbolic methods.

This is also the chronological order of development of the modeling techniques. The later the model appeared, the less computational harden it offered. Naturally, the model generation algorithms are more and more complex.

In this paper, the generation of the dynamic model of robot manipulators using the Lagrange's equations is revised. A computer program for the dynamic analysis of Robots based on Maple Software is built at the Hanoi University of Technology, and called as program RobotDyn. The RobotDyn program has often been used for analysis of the Robot manipulators.

2 Theoretical background of the software RobotDyn

2.1 The Denavit-Hartenberg Matrix

In order to compute the direct kinematics equations for an open-chain manipulator, systematically general methods are to be desired to define the relative position and orientation of two consecutive links and the problems is that to determine two frames attached to the two links and compute the coordinate transformations between them. In general, the frames can be arbitrarily chosen as long as they are attached to the link that are referred to. Nevertheless, it is convenient to set some rules also for the definition of the link frames. With reference to Fig. 1, let axis i denote the axis of the joint connecting link $i - 1$ to link i , the so-called Denavit-Hartenberg convention adopted to define link frame i [1,2]. The resulting coordinate transformation is obtained by postmultiplication of the single transformations as [1, 2, 3,10]

$${}^{i-1}\mathbf{H}_i = \mathbf{H}_i = \begin{bmatrix} \cos \theta_i & -\sin \theta_i \cos \alpha_i & \sin \theta_i \sin \alpha_i & a_i \cos \theta_i \\ \sin \theta_i & \cos \theta_i \cos \alpha_i & -\cos \theta_i \sin \alpha_i & a_i \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (2.1)$$

Notice that the transformation matrix from frame i to frame $i - 1$ is a function only of the joint variable q_i , that is θ_i for a revolution joint or d_i for a prismatic joint.

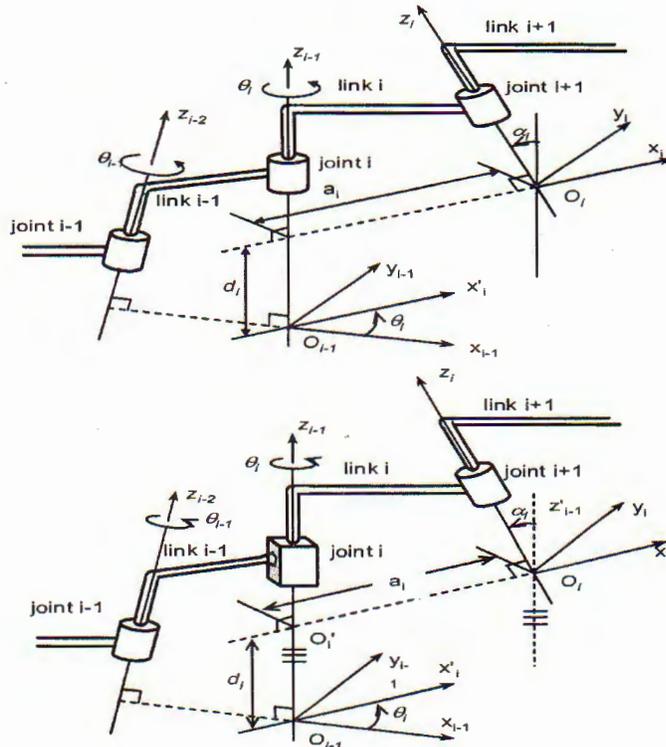


Fig 1. The DH parameters

The transformation matrix (2.1) can be rewritten in the form

$$\mathbf{H}_i = \begin{bmatrix} {}^{i-1}\mathbf{A}_i & {}^{i-1}\mathbf{p}_i \\ \mathbf{0}^T & 1 \end{bmatrix}. \quad (2.2)$$

The coordinate transformation describing the position and orientation of frame i with respect to frame 0 is given by

$$\mathbf{T}_i = \mathbf{H}_1 \mathbf{H}_2 \cdots \mathbf{H}_i = \begin{bmatrix} {}^0\mathbf{A}_i & {}^0\mathbf{p}_i \\ \mathbf{0}^T & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{A}_i & \mathbf{p}_i \\ \mathbf{0}^T & 1 \end{bmatrix}. \quad (2.3)$$

The angular velocity of the link i may be defined from components of the transformation matrix \mathbf{A}_i and its time derivative $\dot{\mathbf{A}}_i$.

$$\tilde{\omega}'_i = (\mathbf{A}_i)^T \dot{\mathbf{A}}_i. \quad (2.4)$$

The vector locates in the reference frame R_0 is given as

$$\mathbf{r}_{C_i} = \mathbf{p}_i + \mathbf{A}_i \mathbf{s}'_{C_i}. \quad (2.5)$$

The time derivative of (2.5) yields

$$\mathbf{v}_{C_i} = \dot{\mathbf{r}}_{C_i} = \dot{\mathbf{p}}_i + \dot{\mathbf{A}}_i \mathbf{s}'_{C_i}. \quad (2.6)$$

2.2 Lagrange's equations of motion of Robot Manipulators

The Lagrange's equations of motion can be written in the following form

$$\frac{d}{dt} \left(\frac{\partial T}{\partial \dot{q}_k} \right) - \frac{\partial T}{\partial q_k} = - \frac{\partial \Pi}{\partial q_k} + Q_k^* \quad (k = 1, \dots, n). \quad (2.7)$$

In the equations (2.7) T is the kinetic energy, Π is the potential energy, Q_k^* represents nonconservative forces. For the Robot manipulators, the kinetic energy has the following form [3]

$$T(q, \dot{q}) = \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n m_{ij}(q) \dot{q}_i \dot{q}_j. \quad (2.8)$$

The potential energy associated with the gravity

$$\Pi = - \sum_{i=1}^n m_i \vec{g}_i^T \vec{r}_{C_i} = \sum_{i=1}^n m_i \mathbf{g}^T \mathbf{r}_{C_i}. \quad (2.9)$$

An explicit form of the generalized forces is given as follows:

$$\begin{aligned} Q_k^* &= \sum_{i=1}^N \vec{F}_i \frac{\partial \vec{r}_i}{\partial q_k} + \sum_{i=1}^M \vec{M}_j \frac{\partial \vec{\theta}_j}{\partial q_k} \\ &= \sum_{i=1}^N \vec{F}_i \frac{\partial \vec{v}_i}{\partial \dot{q}_k} + \sum_{i=1}^M \vec{M}_j \frac{\partial \vec{\omega}_j}{\partial \dot{q}_k} \quad (k = 1, \dots, n). \end{aligned} \quad (2.10)$$

where \vec{F}_i and \vec{M}_j are respectively the force and moment vectors, while \vec{v}_i and $\vec{\omega}_j$ are the corresponding linear and angular (absolute) velocities.

Taking the derivatives required by Lagrange's equations in (2.7) and recalling that Π does not depend on \dot{q} yields.

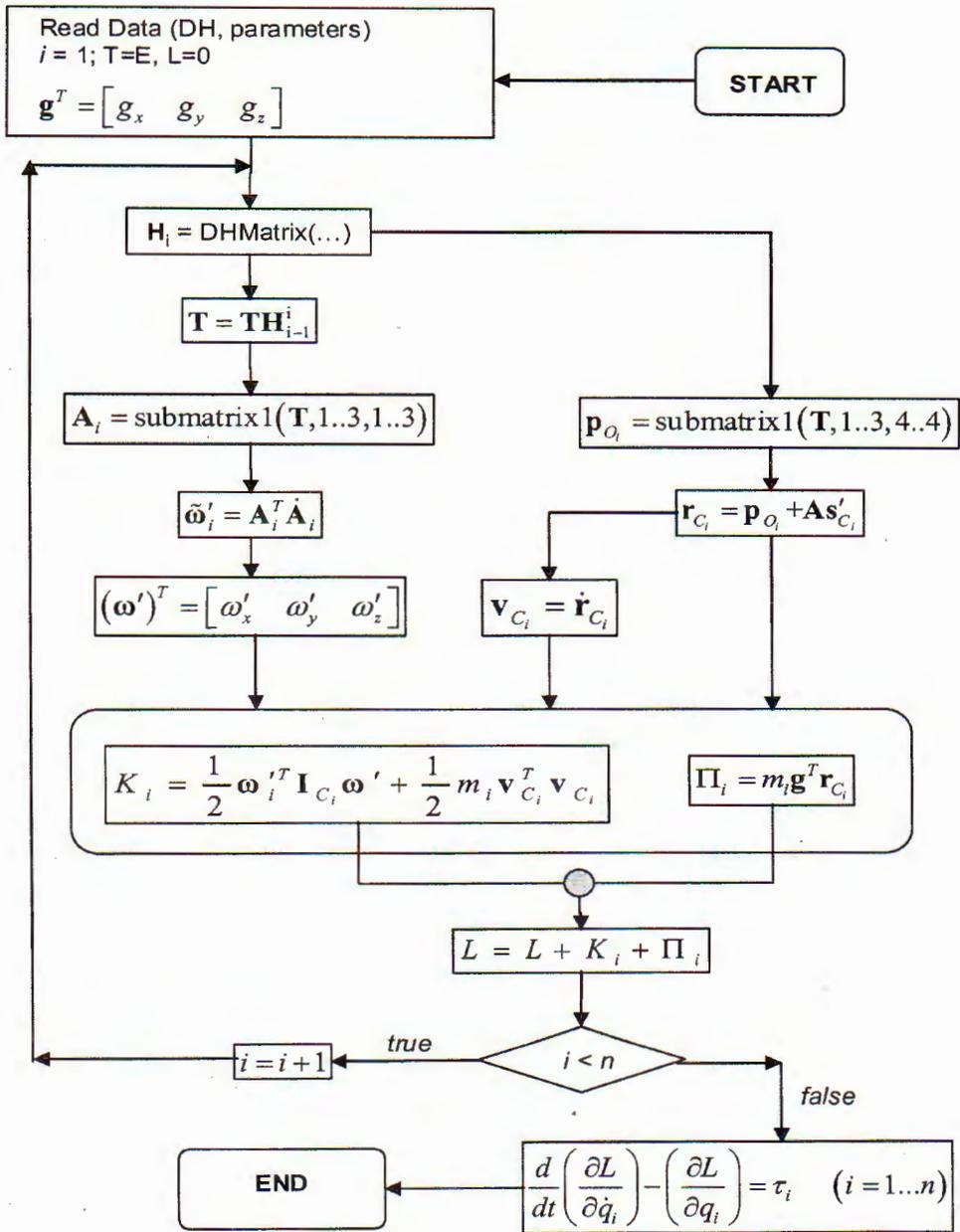


Fig 2. The algorithms structure of the program RobotDyn

$$\sum_{j=1}^n m_{kj}(q)\ddot{q}_j + \sum_{i=1}^n \sum_{j=1}^n h_{kij}(q)\dot{q}_i\dot{q}_j + g_k(q) = \tau_k, \quad (2.11)$$

where

$$h_{kij} = \frac{\partial m_{ki}}{\partial q_j} - \frac{1}{2} \frac{\partial m_{ij}}{\partial q_k}, \quad (2.12)$$

τ_k : is the generalized force associated with the generalized coordinate q_k .

In sum, the equations of motion in (2.11) can be rewritten the matrix form which represents the joint space dynamic model

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{B}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau}. \quad (2.13)$$

The computer program RobotDyn is based on the kinematical analysis with method Denavit-Hartenberg matrix and the dynamical analysis with Lagrange Equations as presented above. The algorithms structure of the program RobotDyn is show in Fig. 2. Some examples on the automatic derivation of the motion equations of industrial robots with Maple are presented in [6,7,8,9].

3 Dynamics model of the SCARA with four degrees of freedom

As an example of the application of the software RobotDyn let us consider the model of Robot SCARA with four degrees of freedom shown in Fig.3, for which the vector of generalized coordinates is $\mathbf{Q} = [\theta_1, \theta_2, \theta_3, d_4]^T = [q_1, q_2, q_3, q_4]^T$.

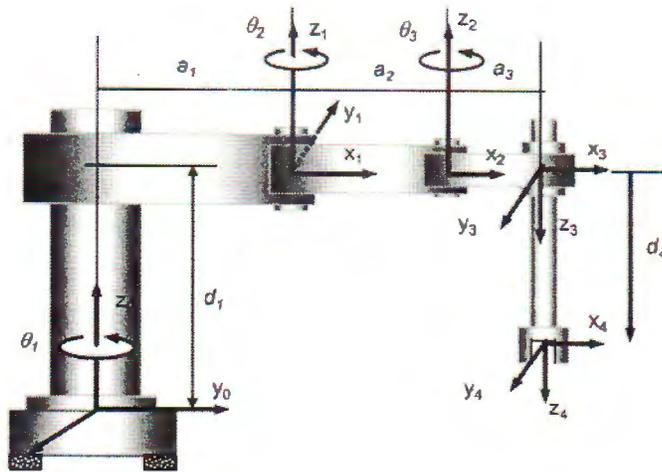


Fig. 3 . Model of Scara Robot with four DOFs

The D-H Parameters table of SCARA robot has the following form

Table 1

link	θ	d	a	α
1	q_1	d_1	a_1	0
2	q_2	d_2	a_2	0
3	q_3	0	a_3	π
4	0	q_4	0	0

Table 2

link	center of gravity			mass	inertial matrix					
	x_C	y_C	z_C		I_{xx}	I_{yy}	I_{zz}	I_{xy}	I_{xz}	I_{yz}
1	$-(a_1 - l_1)$	0	0	m_1	I_{1x}	I_{1y}	I_{1z}	0	0	0
2	$-(a_2 - l_2)$	0	0	m_2	I_{2x}	I_{2y}	I_{2z}	0	0	0
3	$-(a_3 - l_3)$	0	0	m_3	I_{3x}	I_{3y}	I_{3z}	0	0	0
4	0	0	$-l_4$	m_4	I_{4x}	I_{4y}	I_{4z}	0	0	0

Oversimplify we use following notations

$$\begin{aligned}
c_1 &= \cos q_1, & c_2 &= \cos q_2, & c_3 &= \cos q_3, & c_{12} &= \cos(q_1 + q_2), & c_{23} &= \cos(q_2 + q_3), \\
c_{123} &= \cos(q_1 + q_2 + q_3), & s_1 &= \sin q_1, & s_2 &= \sin q_2, & s_3 &= \sin q_3, & s_{12} &= \sin(q_1 + q_2), \\
s_{23} &= \sin(q_2 + q_3), & s_{123} &= \sin(q_1 + q_2 + q_3), \\
dq_i &= \frac{dq_i}{dt}; & ddq_i &= \frac{d}{dt} \left(\frac{dq_i}{dt} \right).
\end{aligned}$$

Using the software RobotDyn we obtain the differential equations of motion of the SCARA robot

$$\begin{aligned}
\tau_1 &= M_{11}ddq_1 - 2dq_2a_1dq_1l_3m_3s_{23} - 2dq_3a_1s_{23}dq_2a_3m_4 \\
&\quad - 2dq_3a_1s_{23}dq_2l_3m_3 + M_{14}ddq_4 + M_{13}ddq_3 - dq_3^2a_1s_{23}a_3m_4 \\
&\quad - dq_3^2a_1s_{23}l_3m_3 - dq_3^2s_3a_2l_3m_3 - dq_2^2a_1s_2a_2m_4 - dq_2^2a_1l_3m_3s_{23} \\
&\quad - dq_2^2a_1a_3m_4s_{23} - dq_2^2a_1s_2m_2l_2 - dq_2^2a_1s_2a_2m_3 - dq_2^2s_3a_2a_3m_4 \\
&\quad - 2dq_3s_3a_2dq_1a_3m_4 - 2dq_3s_3a_2dq_2l_3m_3 - 2dq_3s_3a_2dq_2a_3m_4 \\
&\quad - 2dq_3a_1s_{23}dq_1l_3m_3 - 2dq_3a_1s_{23}dq_1a_3m_4 - 2dq_2a_1dq_1s_2a_2m_4 \\
&\quad - 2dq_2a_1dq_1s_2a_2m_3 - 2dq_2a_1dq_1a_3m_4s_{23} - 2dq_3s_3a_2dq_1l_3m_3 \\
&\quad - 2dq_2a_1dq_1s_2m_2l_2 + M_{12}ddq_2
\end{aligned}$$

$$\begin{aligned}
\tau_2 &= M_{21}ddq_1 - M_{22}ddq_2 + M_{23}ddq_3 + M_{34}ddq_4 - dq_3^2s_3a_2a_3m_4 \\
&\quad - dq_3^2s_3a_2l_3m_3 - 2dq_3s_3a_2dq_2a_3m_4 - 2dq_3s_3a_2dq_2l_3m_3 \\
&\quad - 2dq_3s_3a_2dq_1a_3m_4 - 2dq_3s_3a_2dq_1l_3m_3 + dq_1^2a_1a_3m_4s_{23} \\
&\quad + dq_1^2a_1l_3m_3s_{23} + dq_1^2a_1s_2a_2m_3 + dq_1^2a_1s_2a_2m_4 + dq_1^2a_1s_2a_2m_4 + dq_1^2a_1s_2m_2l_2
\end{aligned}$$

$$\begin{aligned}
\tau_3 &= M_{31}ddq_1 + M_{32}ddq_2 + M_{33}ddq_3 + M_{34}ddq_4 + a_3m_4dq_1^2s_3a_2 \\
&\quad + ddq_1^2a_1a_3m_4s_{23} + 2a_3m_4s_3dq_2a_2dq_1 + a_3m_4a_2dq_2^2s_3 \\
&\quad + l_3m_3dq_1^2s_3a_2 + dq_1^2a_1l_3m_3s_{23} + 2l_3m_3s_3dq_2a_2dq_1 + l_3m_3a_2dq_2^2s_3
\end{aligned}$$

$$\tau_4 = M_{41}ddq_1 + M_{42}ddq_2 + M_{43}ddq_3 + M_{44}ddq_4 - m_4g$$

where

$$M_{11} = I_{1z} + m_1 l_1^2 + I_{2z} + m_2 a_1^2 + 2m_2 a_1 l_2 c_2 + m_2 l_2^2 + I_{3z} + m_3 a_2^2 \\ + 2m_3 a_2 a_1 c_2 + 2m_3 a_2 l_3 c_3 + m_3 a_1^2 + 2m_3 a_1 l_3 c_{23} + m_3 l_3^2 + I_{4z} + m_4 a_3^2 \\ + 2m_4 a_3 a_2 c_3 + 2m_4 a_3 a_1 c_{23} + m_4 a_2^2 + 2m_4 a_2 a_1 c_2 + m_4 a_1^2$$

$$M_{12} = I_{2z} + m_2 a_1 l_2 c_2 + m_2 l_2^2 + I_{3z} + m_3 a_2^2 + m_3 a_2 a_1 c_2 + 2m_3 a_2 l_3 c_3 \\ + m_3 a_1 l_3 c_{23} + m_3 l_3^2 + I_{4z} + m_4 a_3^2 + 2m_4 a_3 a_2 c_3 + m_4 a_3 a_1 c_{23} + m_4 a_2^2 + m_4 a_2 a_1 c_2$$

$$M_{13} = I_{3z} + m_3 a_2 l_3 c_3 + m_3 a_1 l_3 c_{23} + m_3 l_3^2 + I_{4z} + m_4 a_3^2 + m_4 a_3 a_2 c_3 + m_4 a_3 a_1 c_{23}$$

$$M_{14} = 0$$

$$M_{22} = I_{2z} + m_2 l_2^2 + I_{3z} + m_3 a_2^2 + 2m_3 a_2 l_3 c_3 + m_3 l_3^2 + I_{4z} + m_4 a_3^2 + 2m_4 a_3 a_2 c_3 + m_4 a_2^2.$$

$$M_{23} = I_{3z} + m_3 a_2 l_3 c_3 + m_3 l_3^2 + I_{4z} + m_4 a_3^2 + m_4 a_3 a_2 c_3$$

$$M_{24} = 0, \quad M_{33} = I_{3z} + m_3 l_3^2 + I_{4z} + m_4 a_3^2, \quad M_{34} = 0, \quad M_{43} = 0, \quad M_{44} = m_4$$

For numerical calculation it is necessary to specify the dynamic parameters of this robot, which are arbitrarily assigned as

$$m_1 = 5 \text{ kg}, \quad m_2 = 3.5 \text{ kg}, \quad m_3 = 3 \text{ kg}, \quad m_4 = 2 \text{ kg} \\ a_1 = 0.35, \quad a_2 = 0.25 \text{ m}, \quad a_3 = 0.1 \text{ m} \\ d_1 = 0.5 \text{ m}, \quad l_1 = \frac{a_1}{2}, \quad l_2 = \frac{a_2}{2}, \quad l_3 = \frac{a_3}{2}, \\ I_{1z} = \frac{m_1 a_1^2}{12}, \quad I_{2z} = \frac{m_2 a_2^2}{12}, \quad I_{3z} = \frac{m_3 a_3^2}{12}, \quad I_{4z} = 0.01 \text{ kgm}^2.$$

Suppose that the motion equations of the end-effector are given by:

$$\begin{cases} x_E = 0.3 + 0.15 \cos \varphi \\ y_E = 0.2 + 0.15 \sin \varphi \\ z_E = \sin\left(\frac{\pi t}{2}\right) \\ \theta = q_1 + q_2 + q_3 = 0 \end{cases} \quad (3.2)$$

where

$$\varphi(t) = \begin{cases} \frac{1}{6} a t^3 & \text{when } 0 \leq t \leq \frac{T}{4} \\ \frac{1}{192} a T^3 - \frac{1}{16} a T^2 t + \frac{1}{4} a T t^2 - \frac{1}{6} a t^3 & \text{when } \frac{T}{4} \leq t \leq \frac{3T}{4} \\ -\frac{13}{96} a T^3 - \frac{1}{2} a T^2 t - \frac{1}{2} a T t^2 + \frac{1}{6} a t^3 & \text{when } \frac{3T}{4} \leq t \leq T \end{cases} \quad (3.3)$$

$$a = 32 \frac{2\pi}{64} = \pi, \quad T = 4 \text{ s.}$$

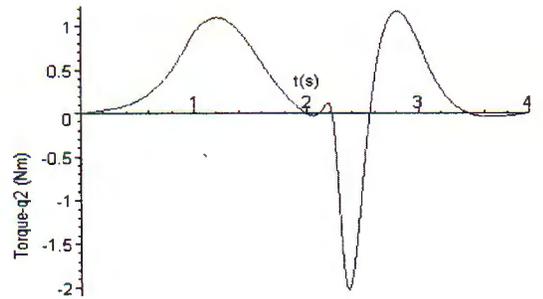
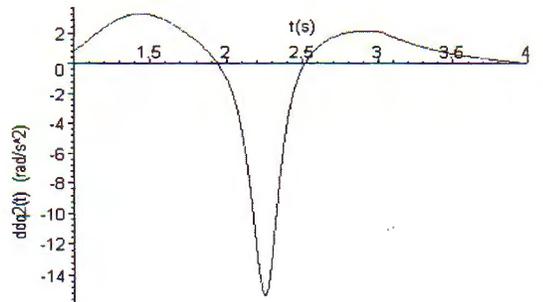
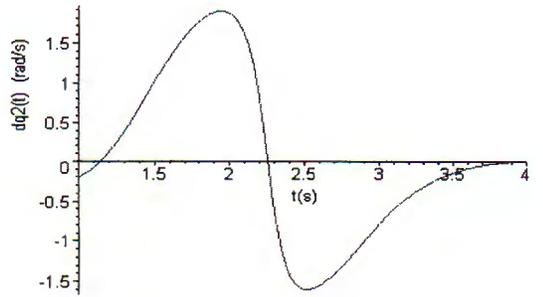
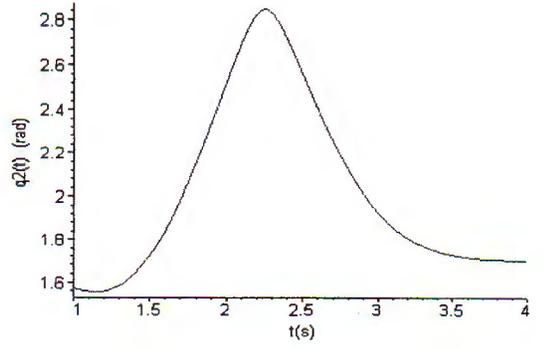
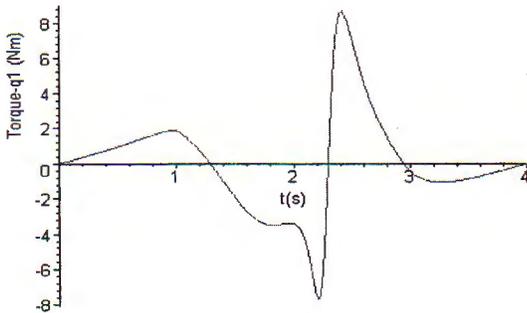
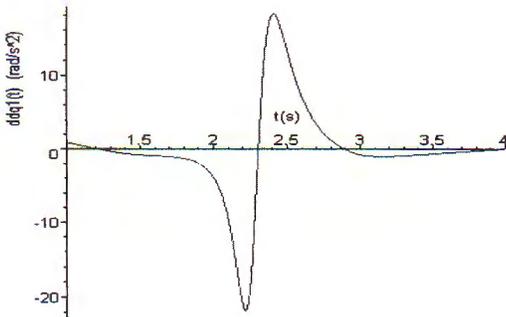
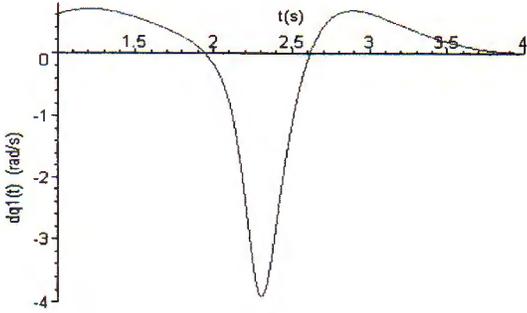
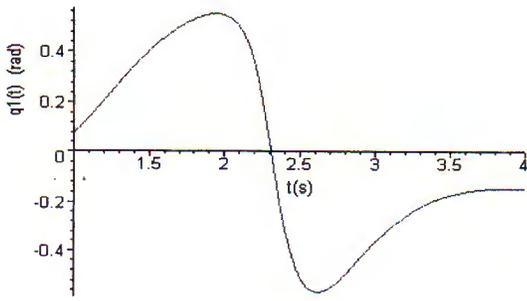


Fig. 4. Position, velocity, acceleration and driving moment curve of the joint 1

Fig. 5. Position, velocity, acceleration and driving moment curve of the joint 2

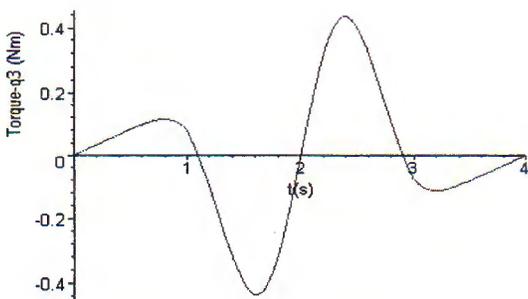
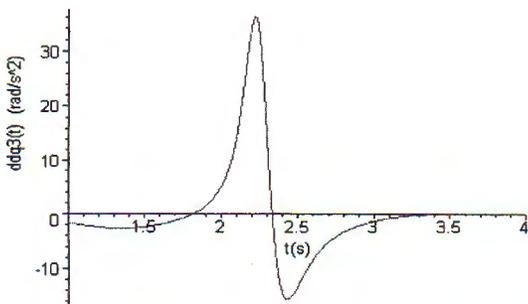
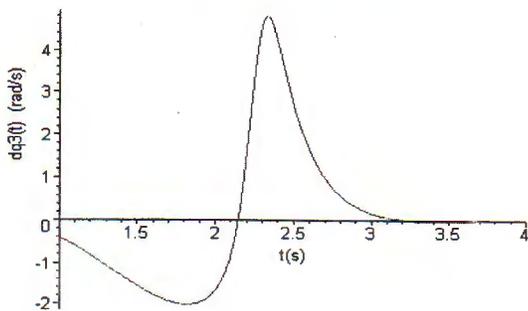
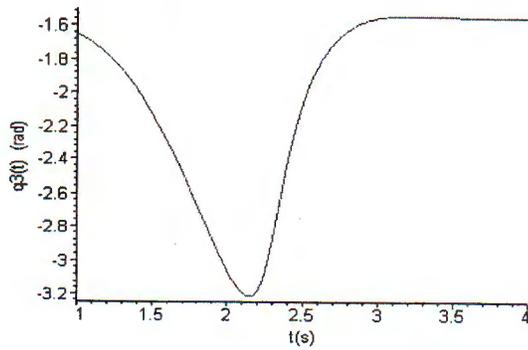


Fig. 6. Position, velocity, acceleration and driving moment curve of the joint 3

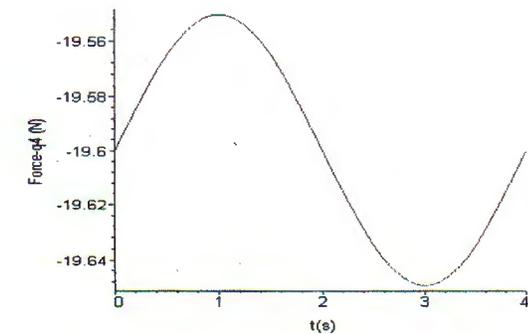
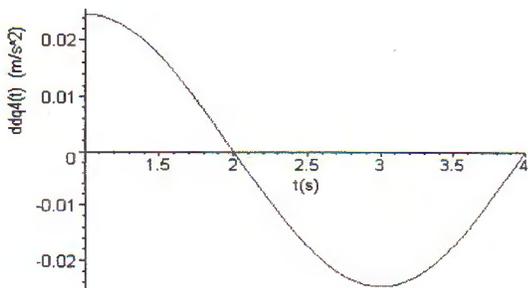
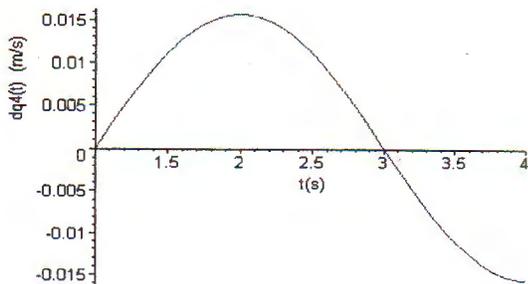
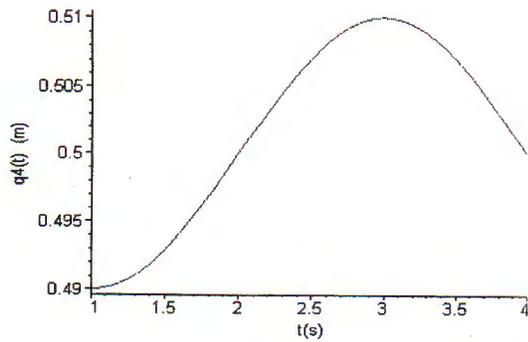


Fig. 7. Position, velocity, acceleration and force curve of the joint 4

The inverse kinematics problem consists of the determination of the joint variables (q_1, q_2, q_3, q_4) corresponding to a given end-effector position and orientation

$$q_i = q_i(x_E, y_E, z_E, \theta), \quad i = 1, 2, 3, 4. \quad (3.4)$$

The final step in solving for the inverse dynamics is to calculate torque at each joint. The torques τ_1, τ_2, τ_3 and τ_4 needed at the joints are found from equations (3.1) after considerable derivation steps.

Some calculation results are shown in Fig. 4, 5, 6 and 7. Fig. 4 shows the position, velocity, and acceleration curve and driving moment curve of the joint 1. Fig. 5 shows the position, velocity, and acceleration curve and driving moment curve of the joint 2. Fig. 6 shows the position, velocity, and acceleration curve and driving moment curve of the joint 3. In the Fig. 7 there are the similar curves of the joint 4.

4 Conclusion

Robot manipulators are multibody systems with tree structures. At the Hanoi University of Technology a computer program to dynamics analysis of robot manipulators with Maple is established. As an example of application the computer simulation of the inverse dynamics of the SCARA robot is presented.

This paper is completed with the financial support from the National Basis Research Program in Nature Sciences.

References

1. Denavit J. and Hartenberg R. S., A kinematic Notation for Lower Pair Mechanisms based on Matrix, *J. App. Mech.* **22** 215-221.
2. Schiehlen (Editor) W., *Multibody Systems Handbook*, Springer-Verlag, Berlin, 1990.
3. Sciavicco L. and Siciliano B., *Modeling and Control of Robot Manipulators*, Springer-Verlag, London, 2000.
4. Yoshikawa T., *Foundation of Robotics-Analysis and Control*, Massachusetts Institute of Technology, 1990.
5. Murray R. M., Zexiang Li. S. S. Sastry. *A Mathematical Introduction to Robotic Manipulator*, CRC Press, Boca Raton, 1994.
6. Nguyen Van Khang and Nguyen Quang Hoang, Automatic derivation of the Motion Equations of industrial Robot based on MAPLE Software, *Proceeding of the National Conference on Engineering Mechanics*, Vol 1. Vietnam National University Publishing House, Hanoi 2001.
7. Nguyen Van Khang and Nguyen Quang Hoang, On a Stability Problem of Industrial Robots. *Proceedings of the 2nd Vietnam-Japan Symposium on Advances in Applied Electromagnetic and Mechanics*, Hanoi 2000.
8. Nguyen Quang Hoang, Studying Dynamics and Stability Behavior of the Manipulator with Maple (in Vietnamese), *Master Thesis*, Hanoi University of Technology, 1999.
9. Do Thanh Trung, Modeling and Numerical Simulation of Robot Dynamics (in Vietnamese), *Master Thesis*, Hanoi University of Technology, 2003.

10. Nguyen Thien Phuc, *Industrial Robots (in Vietnamese) Science and Technical Publishing House*, Hanoi, 2002.

Received August 1, 2004

MÔ PHỎNG ĐỘNG LỰC HỌC RÔBỐT BẰNG PHẦN MỀM ROBOTDYN

Rôbốt là hệ nhiều vật có cấu trúc cây. Trong bài báo này trình bày cơ sở lý thuyết xây dựng hệ chương trình RobotDyn. Chương trình được thiết lập dựa trên phương trình Lagrange và phương pháp ma trận Denavit-Hartenberg. Rôbốt SCARA với bốn bậc tự do được lấy làm ví dụ để minh họa các thuật toán của chương trình.