Comparison 11: SCARA Robot

C11 - SCARA Robot, is the 11th comparison on simulation software and modelling techniques. It is the 6th comparison of continuous type and deals with the handling of implicit systems.

Background: Mechanical and mechatronic systems often result in an implicit second order model description of the type

$$M(\vec{q})\ddot{\vec{q}} = \vec{g}(\dot{\vec{q}}, \vec{q}, \vec{u}, t)$$

with a state-dependent mass matrix M, an acceleration vector $\ddot{\vec{q}}$ and a generalised force function $\vec{g}$.

Simulators often impose restrictions for this type of model descriptions. Only a few simulators accept the description as given above, some allow a description as an implicit first order system

$$A(\vec{z})\dot{\vec{z}} = \vec{h}(\vec{z}, \vec{u}, t)$$

and some require the explicit description given by

$$\dot{\vec{z}} = \vec{f}(\vec{z}, \vec{u}) = A(\vec{z})^{-1}\vec{h}(\vec{z}, \vec{u}, t) .$$

The symbolic derivation of the explicit form is only possible with reasonable effort for very small systems or systems with a simple-structured mass matrix. Therefore it is common practice to carry out the inversion of the mass matrix numerically.

Another interesting question is, whether a simulator that permits implicit descriptions breaks the implicit loop before integrating the states or uses an implicit integration scheme to solve the system directly. Few simulators offer so-called DAE solvers for the second method, sometimes with restrictions with respect to other features like linearisation, event handling etc. In general, advanced features like implicit description, DAE solvers, algebraic loop solvers etc. result in higher computation times and in some computational overhead. Therefore it has to be checked whether it is worth to use such a tool or to work "conventionally" by setting up an explicit system description. In order to investigate this class of problems, a model for a SCARA robot (*Selective Compliance Assembly Robot Arm*) as shown on the title page of this SNE issue was chosen.

Mechanical System (Task a)

A three-axis SCARA robot as indicated in Fig.1 is investigated. This robot type has two vertical revolute joints and one vertical prismatic joint. The axes of all three joints are vertical (parallel to the z-axis in Fig. 1).

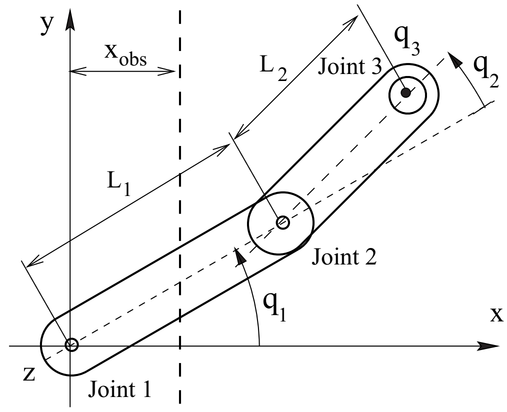

Figure 1

The joint vector $\vec{q}$ consists of the joint angles q_1 and q_2 and the joint distance q_3.

$$\vec{q} = (q_1, q_2, q_3)^T, \quad \dot{\vec{q}} = \frac{d\vec{q}}{dt}, \quad \ddot{\vec{q}} = \frac{d\dot{\vec{q}}}{dt}$$

The equations of motion of can be written in the following compact form

$$M\ddot{\vec{q}} = \vec{b} .$$

The mass matrix M is block-diagonal and can be easily inverted symbolically.

$$M = \begin{bmatrix} ma_{11} & ma_{12} & 0 \\ ma_{21} & ma_{22} & 0 \\ 0 & 0 & ma_{33} \end{bmatrix}$$

Several elements of M depend on the joint variable q_2

$$ma_{11} = \Theta_1 + 2\Theta_2\cos(q_2) + \Theta_3,$$
$$ma_{12} = \Theta_2\cos(q_2) + \Theta_3,$$
$$ma_{21} = ma_{12}, \quad ma_{22} = \Theta_3,$$
$$ma_{33} = m_{3L} + \Theta_{3mot}u_3^2 .$$

The calculation of the moments of inertia Θ_i is based on the assumption that the two physical links are rods of mass m_1, m_2 with homogeneous mass distribution along the length L_1, L_2. The stator mass of the vertical drive motor is m_{3A},the moment of inertia of the rotating parts is Θ_{3mot} and the mass of the load is m_{3L}.

$$\Theta_1 = (\frac{m_1}{3} + m_2 + m_3)L_1^2, \quad \Theta_2 = (\frac{m_2}{2} + m_3)L_1L_2,$$

$$\Theta_3 = (\frac{m_2}{3} + m_3)L_2^2, \quad m_3 = m_{3A} + m_{3L}$$

The right-hand side of the dynamic equations is

$$\vec{b} = (b_1, b_2, b_3)^T,$$

$$b_1 = T_1 + \Theta_2 (2\dot{q}_1 \dot{q}_2 + \dot{q}_2^2) \sin(q_2),$$

$$b_2 = T_2 - \Theta_2 \dot{q}_1^2 \sin(q_2), \; b_3 = T_3 - m_{3L} g$$

with the joint torques $T_1(t)$, $T_2(t)$ and the joint force $T_3(t)$. Numerical data for the geometric and mass parameters of the SCARA robot are given below:

$$m_1 = 8\text{kg}, \quad L_1 = 0.4\text{m}, \quad g = 9.81\,\text{m/s}^2,$$

$$m_2 = 6\text{kg}, \quad L_2 = 0.3\text{m}, \quad u_3 = 1047\text{m}^{-1},$$

$$m_{3A} = 2.5\text{kg}, \; m_{3L} = 0.5\text{kg}, \; \Theta_{3mot} = 9.1 \cdot 10^{-6}\,\text{kgm}^2$$

Servo Motor and PD-Control (Task b)

The electrical relationship of the armature of a robot servo motor is given by a first order differential equation

$$\dot{I}_i = \frac{(U_{ai} - k_{Ti} u_i \dot{q}_i - R_{ai} I_{ai})}{L_{ai}}, \quad i = 1,2,3$$

$$I_{ai} = [-I_i^{max} \le I_i \le I_i^{max}], \quad i = 1,2,3$$

where $U_{ai}(t)$ is the applied armature voltage. The resulting armature current I_i is limited to maximum value I_i^{max} that can be calculated from the maximum permitted torque T_i^{max}

$$I_i^{max} = T_i^{max} \left(\frac{\sqrt{3}}{2} k_{Ti}\right)^{-1}, \quad i = 1,2,3.$$

The joint torque (force) T_i of a motor is proportional to the armature current I_{ai} and given by

$$T_i = u_i \frac{\sqrt{3}}{2} k_{Ti} I_{ai}, \quad i = 1,2,3.$$

Numerical values for the motor constant k_{Ti}, the gear ratio u_i, the resistance R_i and the inductance L_i for each motor are given below. Note that u_3 includes the transformation from the rotational to the linear motion and is not dimensionless.

$$k_{T1} = 0.4\,\text{Vs}, \quad k_{T2} = 0.25\,\text{Vs}, \quad k_{T3} = 0.4\,\text{Vs},$$

$$R_{a1} = 3.9\,\text{Ohm}, \quad R_{a2} = 50\,\text{Ohm}, \quad R_{a3} = 40\,\text{Ohm},$$

$$L_{a1} = 7.3\,\text{mH}, \quad L_{a2} = 25\,\text{mH}, \quad L_{a3} = 25\,\text{mH},$$

$$u_1 = 130, \quad u_2 = 100, \quad u_3 = 1047\,\text{m}^{-1},$$

$$T_1^{max} = 2.3\,\text{Nm}, \quad T_2^{max} = 0.6\,\text{Nm}, \quad T_3^{max} = 0.6\,\text{Nm}$$

In order to control the point-to-point motion of the robot a rather primitive single-axis PD-control is employed. For a given target joint position vector $\hat{\vec{q}}$ position errors $(\hat{q}_i - q_i)$ can be calculated. From the position errors and the joint velocities $\dot{q}_i$ the control voltage U_{ai} is determined by

$$U_i = P_i(\hat{q}_i - q_i) - D_i \dot{q}_i, \quad i = 1,2,3$$

$$U_{ai} = [-U_i^{max} \le U_i \le U_i^{max}], \quad i = 1,2,3.$$

Proportional gains P_i and derivative gains D_i are given for each controller. In regular operation mode the armature voltage shall be limited by $U_{i\,reg}^{max}$. However, in an emergency situation $U_{i\,max}^{max}$ may be used (see task c).

$$P_1 = 1000\,\text{V}, \quad P_2 = 1000\,\text{V}, \quad P_3 = 5000\,\text{V},$$

$$D_1 = 10\,\text{Vs}, \quad D_2 = 25\,\text{Vs}, \quad D_3 = 10\,\text{Vs},$$

$$U_{1reg}^{max} = 100\,\text{V}, \quad U_{2reg}^{max} = 75\,\text{V}, \quad U_{3reg}^{max} = 90\,\text{V},$$

$$U_{1max}^{max} = 230\,\text{V}, \quad U_{2\,max}^{max} = 230\,\text{V}, \quad U_{3\,max}^{max} = 230\,\text{V}$$

Obstacle definition and collision avoidance manoeuvre (Task c)

An elevation profile for the *x*-*y* plane is given by

$$h = h_{obs}\ \forall\ x \le x_{obs}, \quad h = 0\ \forall\ x > x_{obs}$$

$$h_{obs} = 0.2\,\text{m}, \quad x_{obs} = 0.25\,\text{m}$$

with a straight borderline at x_{obs}, separating the elevated area h_{obs} from the area with zero elevation. The border represents an obstacle for the tool-tip of the robot arm. Contact has to be avoided when the robot tool-tip moves from the starting point to a target position in the elevated area. Possible contact must be detected during robot motion and control of the rotational drives must be changed until the tool-tip has cleared the obstacle height. Maximum voltage $U_{i\,max}^{max}$ may be used in this situation for motors 1 and 2 to obtain maximum deceleration. An obstacle sensor shall measure the distance from the robot tool-tip to the borderline and shall trigger an emergency manoeuvre if the distance d falls below the critical distance $d_{crit} = 0.1$ m.

If $(x_{tip} - x_{obs}) \le d_{crit}$ and $q_3 < h_{obs}$ then

decelerate $\dot{q}_1, \dot{q}_2$ until $q_3 > h_{obs}$.

The *x*-position of the tool-tip can be calculated from

$$x_{tip} = x_3 = L_1 \cos(q_1) + L_2 \cos(q_1 + q_2).$$

The following tasks should be performed:

Task a) Modelling method. There are several ways to formulate and implement the model, depending on the simulator's features, e.g.

i) "manual" symbolic manipulations for setting up explicit model equations, implementation of the explicit model description,

ii) derivation of explicit equations using software for symbolic calculations, implementation of the explicit model description,

iii) using special features of the simulator for deriving and simulating the equations (mechatronic modules, etc.),

iv) implementation of the implicit equations, using algebraic loop breaking features of the simulator,

iv) implementation of the implicit equations, using an implicit solver of the simulator, etc.

The simulator's features for this type of models should be sketched briefly by giving (parts of) the model description of at least one (but preferably of some) of the above given methods. In case of alternative modelling approaches the effectiveness should be compared, taking into account preparation time, necessary knowledge for certain alternatives, etc.

Task b) Simulation of a point-to-point motion, controlled by a single axis PD-control shall be performed. No obstacle is present for this task.
Initial values at $t = 0$:

$$q_1 = q_2 = q_3 = 0, \quad \dot{q}_1 = \dot{q}_2 = \dot{q}_3 = 0$$

Target (terminal) values at $\hat{t}$:

$$\hat{q}_1 = \hat{q}_2 = 2, \quad \hat{q}_3 = 0.3\,\text{m}, \quad \dot{\hat{q}}_1 = \dot{\hat{q}}_2 = \dot{\hat{q}}_3 = 0$$

As results graphs of the joint positions should be plotted. In case of alternative model descriptions simulation times are to be compared.

Task c) Collision avoidance may cause difficulties in the models descriptions. Based on the point-to-point control of task b), now an obstacle has to be avoided (see problem definition). Extend the model description by a collision avoidance feature of the proposed type, using for instance state-dependent control, state event mechanism, etc.

For documentation the program extensions are to be outlined and a plot of $x_{tip}(t)$, $(q_3(t) - h_{\text{obs}})$ and x_{obs} over t is to be given.

References: R.J.Schilling, Fundamentals of Robotics, Prentice-Hall, 1990

Acknowledgement: The authors thank Dr. G. Kronreif (TU Vienna) for providing realistic robot data.

Horst Ecker, Institute for Machine Dynamics and Measurement, email: hecker@email.tuwien.ac.at, Tel: +43-1-58801 5567, and Felix Breitenecker, SIM-TECH, Vienna University of Technology, Wiedner Hauptstr. 8-10, A-1040 Vienna/Austria.