Accelerometer based evaluation of industrial robot kinematics derived in Maple

Johanna Wallén, Mikael Norrlöf, Svante Gunnarsson

Division of Automatic Control

E-mail: johanna@isy.liu.se, mino@isy.liu.se,
svante@isy.liu.se

20th September 2007

Report no.: LiTH-ISY-R-2825
Accepted for publication in Mekatronikmöte 2007, Lund, Sweden

Address:
Department of Electrical Engineering
Linköpings universitet
SE-581 83 Linköping, Sweden

WWW: http://www.control.isy.liu.se

AUTOMATIC CONTROL
REGLERTEKNIK
LINKÖPINGS UNIVERSITET

Abstract

In this paper a step toward making a toolbox for industrial robot modelling based on the computer algebra tool MAPLE is taken. It can be seen as a modelling platform for efficient derivation of the necessary equations for doing, *e.g.*, sensor fusion or state estimation by an Extended Kalman Filter (EKF) algorithm. Forward kinematics is studied and the position and orientation of the robot tool are determined in terms of the Denavit-Hartenberg joint variables. Linear and angular velocities and accelerations are derived using the Jacobian. The toolbox is exemplified using an IRB1400 from ABB Robotics with a so called parallelogram linkage structure. The kinematic relations received are verified using the robot IRB1400 with an accelerometer placed on the robot tool and it is shown that measured acceleration and theoretical acceleration derived from the kinematics agree well. However no flexibilities in the modelling are taken into account, which results in differences between measured and derived acceleration when several motors of the robot cooperate in the motion.

Keywords: Mechanical systems/robotics, Evaluation, Kinematics, MAPLE

Accelerometer based evaluation of industrial robot kinematics derived in MAPLE

Johanna Wallén, Mikael Norrlöf, and Svante Gunnarsson

Abstract— **In this paper a step toward making a toolbox for industrial robot modelling based on the computer algebra tool MAPLE is taken. It can be seen as a modelling platform for efficient derivation of the necessary equations for doing, *e.g.*, sensor fusion or state estimation by an Extended Kalman Filter (EKF) algorithm. Forward kinematics is studied and the position and orientation of the robot tool are determined in terms of the Denavit-Hartenberg joint variables. Linear and angular velocities and accelerations are derived using the Jacobian. The toolbox is exemplified using an IRB1400 from ABB Robotics with a so called parallelogram linkage structure. The kinematic relations received are verified using the robot IRB1400 with an accelerometer placed on the robot tool and it is shown that measured acceleration and theoretical acceleration derived from the kinematics agree well. However no flexibilities in the modelling are taken into account, which results in differences between measured and derived acceleration when several motors of the robot cooperate in the motion.**

I. INTRODUCTION

Nowadays industrial robots are not only seen in traditional robotics applications like for example welding and industrial production, but also in various fields as space and underwater applications, food industry, medical care and service among others [2]. These new areas increase the demands on performance and productivity. Low-cost alternatives of motors, gears and other components are also used to a greater extent when manufacturing the robots, which brings a higher degree of nonlinearity and mechanical flexibility and makes it more difficult to achieve the demands on performance. One way to maintain, or even improve the performance, is to use low-cost sensors such as accelerometers, gyros, cameras, *etc.* Using information from the sensors requires efficient methods for integrating information from several sensors, and implicitly also good models and good modelling tools. The size of the kinematic equations are large for robots with only a few links as well, which motivates using a computer algebra tool.

As an illustration, consider a rigid body model of a robot,

$$M(q)\ddot{q}+C(q,\dot{q})\dot{q}+g(q)=u, \tag{1}$$

where M is the inertia matrix, C the Coriolis and centripetal term, g represents the gravity, u is the input torque, and q is a vector of the motor angles. The model (1) can rewritten in state-space form

$$\dot{x}=f(x,u) \tag{2}$$

This work was supported by the Swedish Research Council (VR).

J. Wallén, M. Norrlöf, and S. Gunnarsson are with Division of Automatic Control, Department of Electrical Engineering, Linköping University, SE-58183 Linköping, Sweden, {johanna, mino, svante}@isy.liu.se

with the states $x=(q,\ \dot{q})^T$. Sensor integration can now be interpreted as modelling the measurements produced by the sensor as a function of the states,

$$y=h(x). \tag{3}$$

A gyro measures the angular velocity $\omega_n^n = R_n^0\omega_0^n$ in the gyro coordinate system, which is described by the Jacobian. Using an accelerometer instead gives the linear acceleration $\ddot{x}_n^n = R_n^0\ddot{x}_0^n$ in the accelerometer frame. In order to model the measurement as a functin of the states, the Jacobian has to be differentiated

$$\begin{aligned}\begin{pmatrix}\ddot{x}_0^n\\ \dot{\omega}_0^n\end{pmatrix} &= \frac{d}{dt}\big(J_0^n(q)\dot{q}\big) = J_0^n(q)\ddot{q}+\frac{d}{dt}\big(J_0^n(q)\big)\dot{q}\\ &= J_0^n(q)\ddot{q}+\left(\sum_{i=1}^{n}\frac{\partial}{\partial q_i}\big(J_0^n(q)\big)\dot{q}_i\right)\dot{q}.\end{aligned} \tag{4}$$

Note that $\ddot{q}$ appears explicitly in (4), but it is not part of the state vector. To compute $\ddot{q}$, the model (1) must be utilized and it will make the function h in (3) very complex. Again this is a motivation to use a symbolic modelling tool. [11]

Previous work of robot modelling using symbolic tools is for example Bienkowski and Kozlowski [1], where MATHEMATICA is used for robot modelling, however not using the Denavit-Hartenberg representation [7]. A fairly well-known contribution is *Robotics Toolbox* [4], where numerical calculations of kinematics, dynamics and trajectory planning are possible. It provides simulation and also analysis of results from real robot experiments. Other contributions by Corke are [5], [6], but the focus there is on how dynamic models of robots can be simplified using MAPLE. The MATHEMATICA package *Robotica* [9] is based upon [12], and gives support for both kinematic as well as dynamic modelling. It is however no longer supported or updated.

II. ROBOT KINEMATICS

The forward kinematics describes how the position and orientation of the robot tool are determined in terms of the joint variables. In this work the Denavit-Hartenberg (DH) representation [7] is used, which is a systematic way to parameterise the forward kinematics for rigid robots. It is here described shortly, a more thorough description can be found in [12]. In [14] and [13] an overview is given and applied to the robot IRB 1400 from ABB, shown in Figure 1, which is also used in the experiments in this paper.

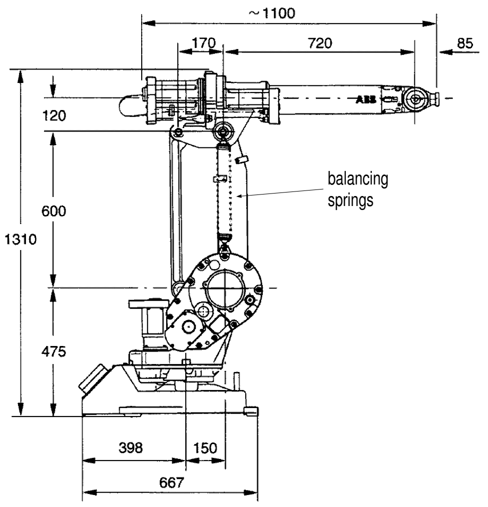

Fig. 1. Blueprint of IRB1400 from ABB, used in the experiments.

A. Short general overview of the forward kinematics

A general transformation between coordinate frame i and frame $i-1$ can be expressed by the rigid motion

$$p_{i-1} = R^i_{i-1} p_i + d^i_{i-1}, \tag{5}$$

which also can be represented by a homogeneous transformation. This gives the transformation as

$$\begin{aligned} P_i &= \begin{pmatrix} p_i \\ 1 \end{pmatrix} \\ A_i &= \begin{pmatrix} R^i_{i-1} & d^i_{i-1} \\ 0 & 1 \end{pmatrix} \\ P_{i-1} &= A_i P_i. \end{aligned} \tag{6}$$

In the DH representation it is regarded as a product of two rotations and two translations, as in

$$A_i = \mathrm{Rot}_{z,\theta_i} \mathrm{Trans}_{z,d_i} \mathrm{Trans}_{x,a_i} \mathrm{Rot}_{x,\alpha_i}. \tag{7}$$

The total homogeneous transformation T^j_i between frame j and frame i is

$$T^j_i = A_{i+1} A_{i+2} \cdots A_{j-1} A_j = \begin{pmatrix} R^j_i & d^j_i \\ 0 & 1 \end{pmatrix}. \tag{8}$$

In the transformation $A_i(q_i)$, three of the four DH link parameters angle θ_i, offset d_i, length a_i, twist α_i are constant, and the fourth is the actual DH joint variable, named q_i. For a revolute joint θ_i is the joint variable, whereas it is d_i for a prismatic joint. Here we only consider prismatic or revolute joints, because all other types of joints can be described by means of them [12].

The Jacobian of the forward kinematic equations determines how the linear velocity $v^n_0 = \dot{x}^n_0$ and angular velocity ω^n_0 of the joint n of the robot, expressed in the base frame 0, is related to the joint velocities according to

$$\begin{pmatrix} v^n_0 \\ \omega^n_0 \end{pmatrix} = \begin{pmatrix} J_v(q) \\ J_\omega(q) \end{pmatrix} \dot{q} = J^n_0(q)\dot{q}. \tag{9}$$

Differentiating the Jacobian gives the linear acceleration $a^n_0 = \ddot{x}^n_0$ and angular acceleration $\dot{\omega}^n_0$ of the joint n, expressed in the base frame 0, resulting in

$$\begin{pmatrix} a^n_0 \\ \dot{\omega}^n_0 \end{pmatrix} = \frac{d}{dt}\left(J^n_0(q)\dot{q}\right). \tag{10}$$

B. Forward kinematics of IRB1400 with accelerometer

The robot IRB1400 only has revolute joints [13]. In the experiments performed in this paper the accelerometer is placed on the robot tool, displaced $x = 32\,\mathrm{mm}$, $y = -26\,\mathrm{mm}$ and $z = 90\,\mathrm{mm}$ relative to the local robot tool frame, oriented as the base coordinate frame. Assume now for simplicity that the robot is only moving its first three motors. The robot can then be seen as consisting of the ordinary moving robot links 1, 2, 3, and the fictitious "links" 4 and 5, which are still. The robot has a closed kinematic chain (motor 2 and 3 are mechanically coupled), which, due to the bilinear form, makes it possible to rewrite the motor angles to DH joint variables for the corresponding, open structure. The closed kinematic chain and the zero pose of the robot now result in the links and fictitious links $1, \ldots, 5$ with the parameters and DH joint variables given in Table I, where the variables q_i are the motor angles for the actual motor. [10], [13]

TABLE I

DH LINK PARAMETERS AND JOINT VARIABLES. [10], [13]

Link i	θ_i $[rad]$	d_i $[m]$	a_i $[rad]$	α_i $[m]$
1	q_1	0.475	0.150	$-\pi/2$
2	$q_2 - \pi/2$	0	0.600	0
3	$-q_2 + q_3$	0	0.120	$-\pi/2$
4	0	0.837	0.090	$\pi/2$
5	$\pi/2$	-0.026	0	$\pi/2$

In this work the transformation between different coordinate systems is used in order to transform the measured acceleration, expressed in the local coordinate frame for the accelerometer ("link" 5), to the robot base frame 0, as in

$$a^5_{acc,0} = R^5_0 a^5_{acc,5}. \tag{11}$$

The measured acceleration is then compared to the theoretical acceleration of the accelerometer expressed in the robot base frame 0. In our case with motors 4 and 5 still, it gives

$$\begin{aligned} \begin{pmatrix} a^5_0 \\ \dot{\omega}^n_0 \end{pmatrix} &= \frac{d}{dt}\left(J^n_0(q)\dot{q}\right) \\ \dot{q} &= \begin{pmatrix} \dot{q}_1 & \dot{q}_2 & \dot{q}_3 & 0 & 0 \end{pmatrix}^T, \end{aligned} \tag{12}$$

which is a function of both the DH joint variables q and its derivatives $\dot{q}$ and $\ddot{q}$.

C. MAPLE implementation

The implementation of the robot kinematics in the toolbox, called `DH-toolbox`, using the computer algebra tool MAPLE is here only shortly described, see [13] for the details. After determining the DH parameters for the robot, each `link` can easily be created in MAPLE as a list by the command DHlink. Then all links are combined to a robot, DHrobot(link1,link2, ...,link5), which is a list of lists, and the following procedures operates on either the robot or the Jacobian. The transformation R_0^5 is given in two steps by first creating T_0^5 by DHmatrixT(robot), followed by DHrotation(T[5]). The Jacobian J can now be calculated by means of DHjacobian(robot). When deriving the accelerations (12), the variables in the differentiation have to be specified with variables:= (Vector[column])([q[1],q[2],q[3],0,0]). Then DHacceleration(J,variables) differentiates the Jacobian with respect to the variables defined, giving the accelerations. The code is thereafter converted and used in MATLAB, for example the transformation R_0^5 by (CodeGeneration['Matlab'])(R05).

III. EVALUATION USING AN ACCELEROMETER

A. Experimental setup

The experiments are performed on the medium size, six degrees of freedom industrial robot IRB1400 from ABB with a payload of 5 kg. Figure 1 shows a blueprint of the robot, and as described in Section II-B, the accelerometer is placed on the robot tool, slightly displaced from the origin of the tool frame. The accelerometer used in the experiments is an Inertial Measurement Unit (IMU) of type MTx from Xsens Technologies [3], which is a miniature inertial three degrees of freedom orientation tracker. The measurement consists of both three dimensional acceleration, rate of turn (rate gyro), and earth-magnetic field, expressed in the local coordinate frame of the accelerometer, but in these experiments only the acceleration measurement is used. The MTx can measure accelerations up to ± 1.7 g, and the default update rate, also used here, is 100 Hz, *i.e.*, a sample time of 10 ms. The measurements are however noisy, mostly depending on that the robot is vibrating when the motors are switched on, even if they just stand in their null position. Therefore the measurements are filtered with a second order Butterworth filter with cutoff frequency 5 Hz, using filtfilt in MATLAB in order to achieve zero phase behaviour.

The experiments are performed by driving the robot in a "circular" motion by moving motors 1, 2, 3, and keep motors 4, 5, 6 still. The motion on the motor side for the moving motors 1, 2, 3 is shown in Figure 2. First, only motor 3 is moved with no restrictions on the motor acceleration and its derivative applied, in order to get a fast movement with distinct measured accelerations and thereby to be able to synchronise the robot motor angular measurements and the accelerometer measurements. Then, motor 1, 2, 3 are moved simultaneously, now an motor acceleration and derivative limited to 10% of their maximum values, giving a smoother trajectory with hopefully accleration with more low-frequency characteristics. This results in the "circular" motion seen in Figure 3, expressed in the robot base frame.

In the experiments an interface between the robot control system S4C and MATLAB is used. The motor angles are logged on the motor side with sample time 4.032 ms with an experimental controller, then converted by the gear change

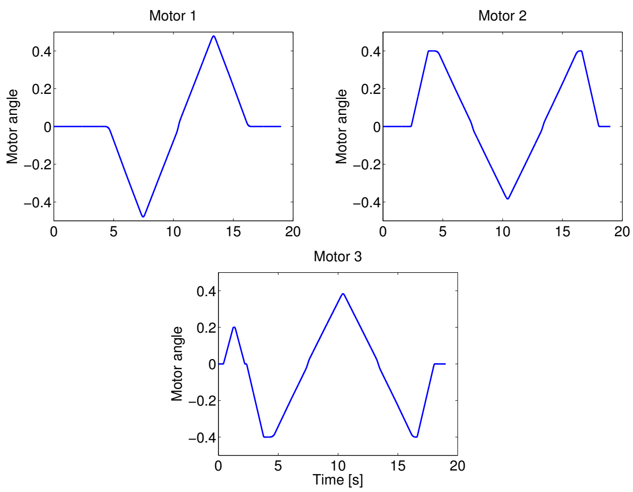

Fig. 2. Experimental motion in motors 1, 2, 3. Motors 4, 5, 6 are still, and therefore not shown here. Motor 3 first performs a synchronisation motion and the circular motion starts just before $t = 5$ s.

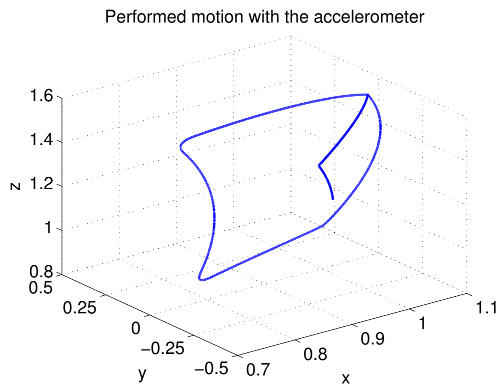

Fig. 3. The motion that the accelerometer, placed on the robot tool, performed in the experiment, expressed in the robot base frame.

to radians on the arm side and down-sampled to sample time 10 ms. Numerical derivatives of the motor angles are needed to derive the theoretical acceleration a_0^5 in (12). They are computed from the down-sampled motor angle measurements using straightforward differentiation

$$\begin{aligned} \dot{q}(t) &\approx \frac{q(t+h)-q(t)}{h}, \qquad h = 0.01\,s \\ \ddot{q}(t) &\approx \frac{q(t+h)-2q(t)+q(t-h)}{h^2}, \end{aligned} \tag{13}$$

and thereafter low-pass filtered.

B. Results

In Figure 4 the acceleration measured by the accelerometer placed on the robot tool, filtered and transformed to the robot base frame, can be seen in x-, y-, and z-direction. These signals are compared to the acceleration (12) derived from the kinematic relations with the DH-toolbox, described in Section II-C. Viewed on the whole, the accelerations derived by the kinematic relations agree well with the measured acceleration. The small offset in x- and y-direction can be explained by an accelerometer position slightly different

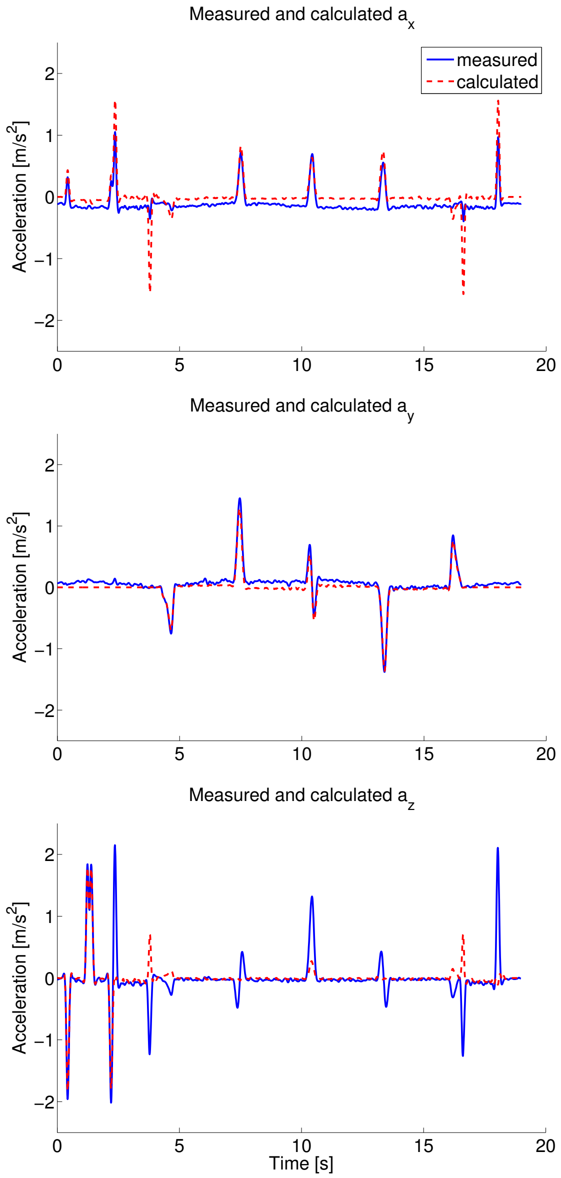

Fig. 4. Acceleration measured by the accelerometer and compared to the acceleration from the kinematic relations derived by the toolbox.

from its measured position. Another reasonable explanation could be that the accelerometer is slightly twisted compared to the robot tool, thereby giving small offsets.

It is interesting to note that the measured and derived acceleration follow each other very well in x- and y-direction, but not in z-direction. In the beginning of the motion, $t \approx [0\ 4.4]$ s, a synchronisation movement with only motor 3 is performed. The accelerations follow each other well, despite that the robot is moved more rapidly with no limits for the motor acceleration and its derivative. With the "circular" motion, involving motors 1, 2, 3, it seems much harder to achieve agreement between measured and calculated acceleration, although the motion is smoother with restrictions on the motor acceleration and derivative applied. This can be explained by unmodelled effects, because the acceleration derived by means of the kinematics is an ideal acceleration, taking no flexibilities into account. These effects especially arise when several motors cooperate when performing the motion. This behaviour can explain the two large differing components of the acceleration in the z-direction at $t = 2.4$, 18.0 s, because the rapid motion when motor 3 stops also involutarily and unexpectedly affects motor 5 and 6 with small, but rapid changes in the magnitude of 10^{-4} rad. This behaviour is neglected in the kinematics, but of course gives rise to a sudden and rapid change of the measured acceleration, because motor 5 and 6 are placed near the location of the accelerometer. Future work naturally includes to incorporate some of these flexibilities in the model.

IV. CONCLUSIONS AND FUTURE WORK

In this paper the forward kinematic relations for a rigid robot are derived and measurements from an accelerometer placed on the robot tool are used to verify the results. The accelerations agree well, but with some of the flexibilities incorporated in the model, the compliance can be improved.

A future application is for example to use the accelerometer measurements in an Iterative Learning Control (ILC) algorithm, which compensates for repetitive errors when the robot is performing a trajectory repeatedly. The acceleration measurements combined with signal processing algorithms then give accurate estimates of the actual path of the tool and can thereby improve the learning. Some works in this direction are presented in for example [8] and [11].

REFERENCES

[1] A. Bienkowski and K. Kozlowski. A toolbox for teaching and researching robotics. In *Proceedings of Worldwide Mathematica Conference*, Chicago, USA, June 1998.

[2] G. S. Bolmsjö. *Industriell robotteknik*. Studentlitteratur, Lund, 2 edition, 1992. In Swedish.

[3] Xsens Technologies B.V. `www.xsens.com`, September 2007.

[4] P. I. Corke. A robotics toolbox for MATLAB. *IEEE Robotics and Automation Magazine*, 3(1):24–32, March 1996.

[5] P. I. Corke. Symbolic algebra for manipulator dynamics. Technical report, CSIRO Division of Manufacturing Technology, Australia, August 1996.

[6] P. I. Corke. A symbolic and numeric procedure for manipulator rigid-body dynamic significance analysis and simplification. *Robotica*, 16:589–594, 1998.

[7] J. Denavit and R. S. Hartenberg. A kinematic notation for lower-pair mechanisms based on matrices. *ASME Journal of Applied Mechanics*, pages 215–221, June 1955.

[8] S. Gunnarsson, M. Norrlöf, E. Rahic, and M. Özbek. On the use of acceleormeters in iterative learning control of a flexible robot arm. *International Journal of Control*, 80(3):363–373, March 2007.

[9] J. F. Nethery and M. W. Spong. Robotica: A Mathematica package for robot analysis. *IEEE Robotics and Automation Magazine*, 1(1):13–20, 1994.

[10] M. Norrlöf. Modeling of industrial robots. Technical Report LiTH-ISY-R-2208, Department of Electrical Engineering, Linköpings universitet, Sweden, December 1999.

[11] M. Norrlöf and R. Karlsson. Position estimation and modeling of a flexible industrial robot. In *Proceedings of IFAC World Congress 2005*, Prague, Czech Republic, July 2005.

[12] M. W. Spong, S. Hutchinson, and M. Vidyasagar. *Robot Modeling and Control*. John Wiley & Sons, Ltd, New York, 2006.

[13] J. Wallén. On robot modelling using Maple. Technical Report LiTH-ISY-R-2723, Department of Electrical Engineering, Linköpings universitet, Sweden, August 2007.

[14] J. Wallén, S. Gunnarsson, and M. Norrlöf. Derivation of kinematic relations for a robot using Maple. In *Proceedings of Reglermöte 2006*, Royal Institute of Technology, Stockholm, Sweden, May 2006.

Avdelning, Institution
Division, Department

Division of Automatic Control
Department of Electrical Engineering

Datum
Date

2007-09-20

Språk
Language

☐ Svenska/Swedish
☒ Engelska/English
☐ ______________

Rapporttyp
Report category

☐ Licentiatavhandling
☐ Examensarbete
☐ C-uppsats
☐ D-uppsats
☒ Övrig rapport
☐ ______________

ISBN
—

ISRN
—

Serietitel och serienummer
Title of series, numbering

ISSN
1400-3902

LiTH-ISY-R-2825

URL för elektronisk version

http://www.control.isy.liu.se

Titel
Title

Accelerometer based evaluation of industrial robot kinematics derived in Maple

Författare
Author

Johanna Wallén, Mikael Norrlöf, Svante Gunnarsson

Sammanfattning
Abstract

In this paper a step toward making a toolbox for industrial robot modelling based on the computer algebra tool Maple is taken. It can be seen as a modelling platform for efficient derivation of the necessary equations for doing, *e.g.*, sensor fusion or state estimation by an Extended Kalman Filter (EKF) algorithm. Forward kinematics is studied and the position and orientation of the robot tool are determined in terms of the Denavit-Hartenberg joint variables. Linear and angular velocities and accelerations are derived using the Jacobian. The toolbox is exemplified using an IRB1400 from ABB Robotics with a so called parallelogram linkage structure. The kinematic relations received are verified using the robot IRB1400 with an accelerometer placed on the robot tool and it is shown that measured acceleration and theoretical acceleration derived from the kinematics agree well. However no flexibilities in the modelling are taken into account, which results in differences between measured and derived acceleration when several motors of the robot cooperate in the motion.

Nyckelord
Keywords

Mechanical systems/robotics, Evaluation, Kinematics, Maple