Design of Pseudoinverse Method
for Robotic Arm

Ondrej Hock [1)] and Jozef Sedo [2)]

[1) 2)] Department of Mechatronics and Electronics, Faculty of Electrical Engineering, University of Zilina, Slovakia,
e-mail: [1)] *ondrej.hock@fel.uniza.sk,* [2)] *jozef.sedo@fel.uniza.sk*

Abstract — **Pseudoinverse method for control end-effector movement of a robotic arm is presented in this paper. For that task we used the robotic arm DOBOT. We made simulation model in Matlab environment and SimMechanics toolbox. We described the pseudoinverse method and problem-solving of pseudoinverse for non-square matrices. In the near future we will make implementation of this method to the control end-effector movement of a robotic arm and therefore we created a simulation to how many digits we will need for correct calculating of goniometric functions in DSP.**

Keywords — inverse kinematics, Matlab Simulink simulation, robotic arm, pseudoinverse method, SimMechanics.

I. INTRODUCTION

Complex control systems nowadays are designed for wide range of industrial applications. They are focused on effective management of composite processes, energy savings and optimization of production and technological processes. For these reasons industry is using robotic manipulators for which we need to know the end-effector kinematics [1].

The robot kinematics can be divided into forward kinematics and inverse kinematics. Forward kinematics problem is straightforward and there is no complexity deriving the equations [2]. Hence, there is always a forward kinematics solution of a manipulator. Inverse kinematics is a much more difficult problem than forward kinematics. The solution of the inverse kinematics problem is computationally expansive and generally takes a very long time in the real time control of manipulators. Singularities and nonlinearities that make the problem more difficult to solve. Hence, only for a very small class of cinematically simple manipulators (manipulators with Euler wrist) have complete analytical solutions [3–5].

Two main solution techniques for the inverse kinematics problem are analytical and numerical methods. In the first type, the joint variables are solved analytically according to the given configuration data. In the second type of solution, the joint variables are obtained based on the numerical techniques. In this chapter, the analytical solution of the manipulators is examined rather than numerical solution [6, 7].

The whole paper will be dedicated to the robot arm DOBOT Magician (hereafter DOBOT) shown in Fig. 1.

Fig. 1. DOBOT Magician [8].

II. THEORY OF PSEUDOINVERSE METHOD

The pseudoinverse method sets the value $\Delta\theta$ equal to

$$\Delta\theta = J^{\dagger}\vec{e} \qquad (1)$$

Where:

- J is Jacobian matrix,
- θ is vector of the individual actuator positions,
- e is vector of the end-effector positions.

The n × m matrix $J^{\dagger}$ is the pseudoinverse of J, also called the Moore-Penrose inverse of J. It is defined for all matrices J, even ones which are not square or not of full row rank. The pseudoinverse gives the best possible solution to the equation $J\Delta\theta = \vec{e}$ in the sense of least squares. In particular, the pseudoinverse has the following nice properties. Let $\Delta\theta$ be defined by equation (1) [9]. First, suppose $\vec{e}$ is in the range (i.e., the column span) of J. In this case, $J\Delta\theta = \vec{e}$; furthermore, $\Delta\theta$ is the unique vector of smallest magnitude satisfying $J\Delta\theta = \vec{e}$. Second, suppose that $\vec{e}$ is not in the range of J. In this case, $J\Delta\theta = \vec{e}$ is impossible. However, $\Delta\theta$ has the property that it minimizes the magnitude of the difference $J\Delta\theta - \vec{e}$. Furthermore, $\Delta\theta$ is the unique vector of smallest magnitude which minimizes $\|J\Delta\theta - \vec{e}\|$, or equivalently, which minimizes $\|J\Delta\theta - \vec{e}\|^2$ [9, 10].

The pseudoinverse tends to have stability problems in the neighbourhoods of singularities. At a singularity, the Jacobian matrix no longer has full row rank, corresponding to the fact that there is a direction of movement of the end effectors which is not achievable [11, 12]. If the configuration is exactly at a singularity, then the pseudoinverse method will not attempt to move in an impossible direction, and the pseudoinverse will be well-behaved. However, if the configuration is close to a singularity, then the pseudoinverse method will lead to very large changes in joint angles, even for small movements in the target position. In practice, roundoff errors mean that true singularities are rarely reached and instead singularity

have to be detected by checking values for being near-zero [13, 14].

The pseudoinverse has the further property that the matrix $(I - J^\dagger J)$ performs a projection onto the nullspace of J. Therefore, for all vectors φ, $J(I - J^\dagger J)\varphi = 0$. This means that we can set $\Delta\theta$ by

$$\Delta\theta = J^\dagger \vec{e} + (I - J^\dagger J)\varphi \qquad (2)$$

for any vector and still obtain a value for which minimizes the value $J\Delta\theta - \vec{e}$. By suitably choosing φ, one can try to achieve secondary goals in addition to having the end effectors track the target positions. For instance, φ might be chosen to try to return the joint angles back to rest positions. This can help avoid singular configurations [9, 10].

An algorithm for the pseudoinverse method can be derived as follows:

From equation (2), we get the normal equation

$$J^T J \Delta\theta = J^T \vec{e} \qquad (3)$$

Then we let $\vec{z} = J^T \vec{e}$ and solve the equation

$$(J^T J)\Delta\theta = \vec{z} \qquad (4)$$

Now it can be shown that $\vec{z}$ is always in the range of $J^T J$, hence equation (4) always has a solution. In principle, row operations can be used to find the solution to (4) with minimum magnitude; however, in the neighborhood of singularities, the algorithm is inherently numerically unstable [9, 15].

When J has full row rank, then JJ^T is guaranteed to be invertible. In this case, the minimum magnitude solution $\Delta\theta$ to equation (4) can be expressed as

$$\Delta\theta = J^T (JJ^T)^{-1} \vec{e} \qquad (5)$$

To prove this, note that if $\Delta\theta$ satisfies (5), then $\Delta\theta$ is in the row span of J and $\Delta\theta = \vec{e}$. Equation (5) cannot be used if J does not have full row rank. A general formula for the pseudoinverse for J not of full row rank can be found in [9, 12].

The pseudoinverse method is widely discussed in the literature but it often performs poorly because of instability near singularities [16].

III. Problem Solving of Pseudoinverse Method

In case, the number of independent coordinates (rotation of individual joints) is greater than the number of coordinates of the manipulator endpoint (in space 6, for translational motion 3 and 3 for rotation), the redundancy problem will arise. In this case, there can generally be infinitely many combinations of independent co-ordinates for one endpoint positioning. The Jacobi matrix has the dimensions of m lines and n columns ($m \neq n$), i.e. J is the non-square matrix. Generally, it is not possible to calculate the inverse matrix from the non-square matrix.

In order to address the inverse role of kinematics in such cases, pseudo-inversion of the Jacobi matrix (designation J^+). This method uses SVD (Singular value decomposition) of the Jacobi matrix for determination J^+.

We can decompose every matrix J, which is Jacobi matrix, by SVD decompose for 3 matrices:

$$J = U.\Sigma.V^T \qquad (6)$$

Where:

- J is Jacobi matrix with dimension $m \times n$
- U is ortogonal matrix with dimension $m \times m$, i.e. $U^{-1} = U^T$
- V is ortogonal matrix with dimension $n \times n$, i.e. $V^{-1} = U^T$
- Σ is diagonal matrix with dimension $m \times n$, which on main diagonal contained singular numbers of matrix J.

$$\begin{bmatrix} j_{11} & \cdots & j_{1n} \\ \vdots & \ddots & \vdots \\ j_{m1} & \cdots & j_{mn} \end{bmatrix} =$$
$$= \begin{bmatrix} u_{11} & \cdots & u_{1m} \\ \vdots & \ddots & \vdots \\ u_{m1} & \cdots & u_{mm} \end{bmatrix} \begin{bmatrix} \sigma_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_d \end{bmatrix} \begin{bmatrix} v_{11} & \cdots & v_{n1} \\ \vdots & \ddots & \vdots \\ v_{1n} & \cdots & v_{nn} \end{bmatrix} \qquad (7)$$

Where:
$d = m$ for $m < n$ and $d = n$ for $m > n$.

$$J.J^T = (U.\Sigma.V^T).(U.\Sigma.V^T)^T$$
$$J.J^T = U.\Sigma.V^T.V.\Sigma^T.U^T$$
$$J.J^T = U.\Sigma.I.\Sigma^T.U^T$$
$$J.J^T = U.\Sigma.\Sigma^T.U^T \qquad (8)$$

We multiply the equation (8) by U matrix for right:

$$JJ^T.U = U.\Sigma\Sigma^T.U^T.U$$
$$JJ^T.U = U.\Sigma\Sigma^T.I$$
$$JJ^T.U = U.\Sigma\Sigma^T \qquad (9)$$

We get problem of eigenvalues for the matrix JJ^T, where U is square matrix of eigenvectors for the matrix JJ^T and $\Sigma\Sigma^T$ is diagonal square matrix of eigenvalues $\lambda_1, \dots, \lambda_m$.

$$\Sigma.\Sigma^T = \begin{bmatrix} \sigma_1^2 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_d^2 \end{bmatrix} \qquad (10)$$

$$J^T.J = (U.\Sigma.V^T)^T.(U.\Sigma.V^T)$$
$$J^T.J = V.\Sigma^T.U^T.U.\Sigma.V^T$$
$$J^T.J = V.\Sigma^T.I.\Sigma.V^T$$
$$J^T.J = V.\Sigma^T.\Sigma.V^T \qquad (11)$$

We multiply the equation (11) by V matrix for right:

$$J^T J.V = V.\Sigma^T\Sigma.V^T.V$$
$$J^T J.V = V.\Sigma^T\Sigma.I$$
$$J^T J.V = V.\Sigma^T\Sigma \qquad (12)$$

We get the problem of eigenvalues for the matrix $J^T J$, where V is square matrix of eigenvectors for the matrix $J^T J$ and $\Sigma^T\Sigma$ is diagonal square matrix of eigenvalues $\lambda_1, \dots, \lambda_n$.

Matrices JJ^T and $J^T J$ are symmetric matrices and they have same non-zero eigenvalues. Eigenvalues of symmetric matrices are always the real values. Therefore we won't calculate eigenvalues of JJ^T again. It is necessary to calculate only eigenvectors of JJ^T.

IV. SIMULATION RESUTLS

We made simulation for the robotic arm in Matlab Simulink and also in m-file. Main idea was to prepare for later implementation of the pseudoinverse method to DSP. For that we made simulation for number of digits after comma and number of steps we necessarily need. Simulation results of the pseudoinverse method can be found in Figs. 2 to 4 and simulation results of necessarily number of digits and number of steps shown in Figs. 5 and 6.

Figure 2 shows the end-effector movement of the robotic arm. We design simulation for 3 reference coordinates as we can see in Fig. 2. For (50,90,80) it is blue line, (100,150,160) is red line and third coordinates were (150,180,140) which is green line.
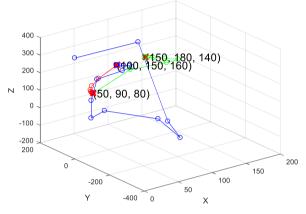

Fig. 2. Simulation result of the pseudoinverse method for DOBOT manipulator.

Figures 3 and 4 show axes X, Y and Z of the robotic arm end-effector and total error of coordinates. Figures also show number of necessarily steps for each simulation. In first result we require maximal error of coordinates equal to 1e-3 mm and number of digits equal to 6. In the second figure we require maximal error equal to 1e-2 mm and number of digits equal to 5.
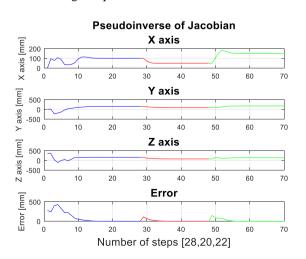

Fig. 3. Simulation result of DOBOT axes and total error of coordinates.
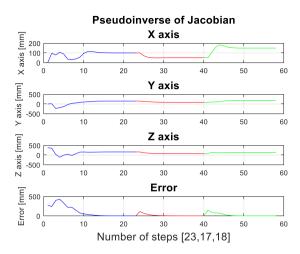

Fig. 4. Simulation result of DOBOT axes and total error of coordinates.

Since these simulations are a preparation for implementing a method into the DSP, it is necessary to know how many digits after point we needed. Number of digits and middle value of error is in TABLE I. Figure 5 shows this table for each axis.

TABLE I.
NUMBER OF DIGITS AND MEAN VALUE OF ERROR

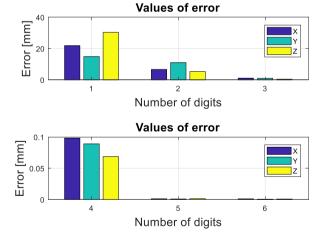| Digits	Mean value of error [mm]
1	22,31
2	7,627
3	0,786
4	0,104
5	11,3e-4
6	8,3e-4

Fig. 5. Simulation result of total error of coordinates from 1 to 6 digit.

In the TABLE II are numbers of simulation steps for reference error and for each reference coordinate from Fig. 2. Figure 6 graphically shows the previous table.

TABLE II.
NUMBER OF SIMULATION STEPS FOR DIFFERENT REFERENCE ERRORS

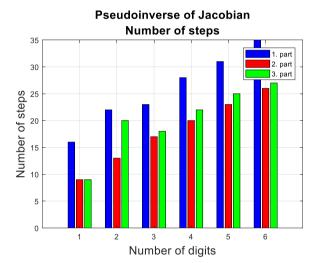| Number of simulation steps			
Reference error [mm]	1. chart	2. chart	3. chart
1	16	9	9
0.1	22	13	20
0.01	23	17	18
0.001	28	20	22
1e-4	31	23	25
1e-5	35	26	27

Fig. 6. Simulation result of total error of coordinates from 1 to 6 digit.

V. CONCLUSION

In this paper we theoretically described pseudoinverse method for the control end-effector movement of DOBOT manipulator. We needn't to research rotation of end-effector because DOBOT mechanically secures the retention of the end-effector in the plane and pad.

For reducing calculation time in DSP it is important to rightly implement the pseudoinverse method. The simulations have shown that the numbers with four decimal places at least have to be used for the achievement of the required precision of 0.1 mm.

The structural precision of the DOBOT manipulator is 0.4 mm. For the achievement of this precision, 20 iteration steps of the pseudoinverse method are needed.

REFERENCES

[1] Virgala, I.; Kelemen, M.; Prada, E.; et al.: Motion analysis of snake robot segment, In: 11th IEEE International Symposium on Applied Machine Intelligence and Informatics (SAMI), Pages: 145-148 Published: 2013

[2] Chovancova, A.; Fico, T.; Hubinsky, P.; et al.: Comparison of various quaternion-based control methods applied to quadrotor with disturbance observer and position estimator, In. ROBOTICS AND AUTONOMOUS SYSTEMS Volume: 79 Pages: 87-98 Published: May 2016

[3] Duchon, Frantisek; Hubinsky, Peter; Hanzel, Jaroslav; et al.: Intelligent vehicles as the robotic applications, In: 5th International Conference on Modelling of Mechanical and Mechatronics Systems (MMaMS) Location: SLOVAKIA Date: NOV 06-08, 2012

[4] MODELLING OF MECHANICAL AND MECHATRONICS SYSTEMS Book Series: Procedia Engineering Volume: 48 Pages: 105-114 Published: 2012

[5] Duchon, Frantisek; Hunady, Dominik; Dekan, Martin; et al.: Optimal navigation for mobile robot in known environment, In: 11th International Conference on Industrial, Service and Humanoid Robotics (ROBTEP 2012), ROBOTICS IN THEORY AND PRACTICE Book Series: Applied Mechanics and Materials Volume: 282 Pages: 33- Published: 2013

[6] Babinec, Andrej; Dekan, Martin; Duchon, Frantisek; et al.: Modifications of VFH navigation methods for mobile robots, In: 5th International Conference on Modelling of Mechanical and Mechatronics Systems (MMaMS), MODELLING OF MECHANICAL AND MECHATRONICS SYSTEMS Book Series: Procedia Engineering Volume: 48 Pages: 10-14 Published: 2012

[7] Vitko, Anton; Jurisica, Ladislav; Babinec, Andrej; et al.: Some didactic aspects of teaching robotics, In: 1st International Conference on Robotics in Education, PROCEEDINGS OF THE 1ST INTERNATIONAL CONFERENCE ON ROBOTICS IN EDUCATION Pages: 27-30 Published: 2010

[8] WebPage, 4.8.2017 : http://www.dobot.cc/

[9] BUSS, Samuel R.: Introduction to inverse kinematics with jacobian transpose, pseudoinverse and damped least squares methods. IEEE Journal of Robotics and Automation, 2004, 17.1-19: 16.

[10] R. Buss, 3-D Computer Graphics: A Mathematical Introduction with OpenGL, Cambridge University Press, 2003.
https://doi.org/10.1017/CBO9780511804991

[11] Jaydev P. Desai: D-H Convention, Robot and Automation Handbook, CRC Press, 2005 USA, ISBN 0-8493-1804-1.

[12] Adelhard Beni Rehiara: Kinematics of AdeptThree Robot Arm, InTech, 2011, ISBN 978-953-307-160-2

[13] Kucuk, S., Bingul.: The Inverse Kinematics Solutions of Industrial Robot Manipulators, IEEE Conference on Mechatronics, pp. 274-279, Turkey, June 2004, Istanbul

[14] Kucuk, S., Bingul.: Robot Kinematics: Forward and Inverse Kinematics, in Industrial Robotics: Theory, Modelling and Control, 2007, ISBN 3-86611-285-8, p 128-148

[15] Kelemen, Michal; Fabian, Michal; Kelemenova, Tatiana: Design and Development of Lift Didactic Model Within Subjects of Mechatronics, In: 5th International Conference on Modelling of Mechanical and Mechatronics Systems (MMaMS), MODELLING OF MECHANICAL AND MECHATRONICS SYSTEMS Book Series: Procedia Engineering Volume: 48 Pages: 280-286 Published: 2012

[16] Mikova, Lubica; Trebuna, Frantisek; Kelemen, Michal: Concept of Locomotion Mobile Undercarriage Structure Control for the Path Tracking, In: MECHATRONIC SYSTEMS AND MATERIALS IV Book Series: Solid State Phenomena Volume: 198 Pages: 79-83 Published: 2013