Combining holistic programming with kinematic parameter optimisation for robot machining

Ulrich Schneider, Julian Ricardo Diaz Posada, Manuel Drust, Alexander Verl
Fraunhofer Institute for Manufacturing Engineering and Automation IPA, Stuttgart, Germany

Johnny van der Zwaag
Delcam plc, Birmingham, United Kingdom

Abstract

Industrial robot technology provides an excellent base for machining as robots are very flexible and cost effective compared to CNC machines. In this article, two methods are presented to reduce set-up time when machining with industrial robots. Combining an advanced programming and simulation environment together with a kinematic compensation approach results in a seamless interaction of all steps in order to avoid manual interference in a later stage of the process. Experimental results will validate the presented approach and outline the potential.

1 Introduction

Machining is traditionally performed on machine tools which are optimised for the milling process: stiffness and accurate movements are essential for precision. Yet handling the workpieces for loading and unloading the machine was originally not considered and robots were used for the tasks of handling and equipping. Robots also provide an excellent base for machining as robots are very flexible and cost effective. Robots are now used for simple machining tasks like milling, drilling, grinding and deburring in soft materials like wood and plastic. Their limited positional accuracy, combined with the build-up of positional errors due to the low stiffness of the robot arm and higher process forces for tougher materials result in poor accuracy in aluminium and steel parts. The robot's structure, mechanics, drives and gears are not designed to withstand the requirements of such tasks.
The main error sources responsible for the limited machining accuracy of industrial robots are described in [1, 2, 3]. Static and dynamical errors can arise because of robot properties, positioning errors and machining forces. In [4] chatter in machining and the development of compensation methods to allow stable cutting have been analysed. Sources for static positioning errors can be found in the tool, mechanical components, gears and bearings [5, 6]. In addition, external disturbances such as temperature effects have an influence on the accuracy [7]. Though if a robot could compensate those errors and provide a satisfying accuracy one could benefit from its smaller costs and higher flexibility than a conventional machine tool. Finally the need for expertise in robotics still prevents end users from integrating robots into their production processes. Suitable support could enable them to access new processes and a higher degree of automation. Most usual tasks for an industrial robot are pick-and-place processes. The repeatability is representing the robot's accuracy for those tasks. The repeatability is much higher than the absolute accuracy. Iterative programming is required to get a satisfying accuracy. Normal machining processes in serial production require a higher accuracy less than 50 µm. The accuracy of conventional tooling machines is in a range of 1 to 5 µm [9]. The EU/FP7 project COMET [8] adresses machining with industrial robots. Sources of errors are analysed and coped with and the user is supported by appropriate software providing particularly robot knowledge. **Figure 1** illustrates four steps towards accurate machining with industrial robots. The upper two pieces of the puzzle, kinematic model and adaptive robot path generation, will be presented in this paper. The two lower parts describe an external real-time compensation mechanism [10] similar to [2, 3] and a fast real-time measurement system.

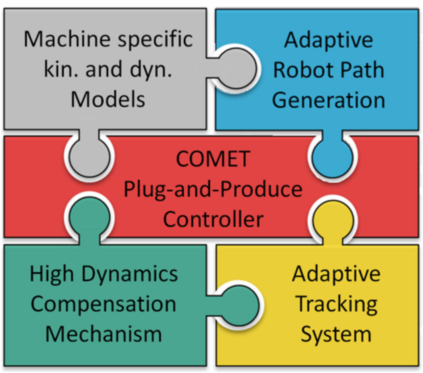

Figure 1: Concept of accurate machining with industrial robots [8]

ISBN 978-3-8007-3601-0

After introducing the topic in Section 1 an advanced programming and simulation approach towards accurate machining is described in Section 2. Section 3 exposes a kinematic approach increasing positioning accuracy of industrial robots. Section 4 presenting experimental results will be followed by conclusions and future work in Section 5.

2 Holistic programming and simulation

2.1 Research aspects

The two most common robot programming methods are using a teaching device or offline programming. Both are not suitable for machining processes. Within COMET [8] state-of-the-art programming solutions from CNC machining will be combined with kinematic and dynamic knowledge of the robot to achieve a single, integrated holistic programming and simulation solution for robotics. In COMET, Delcam is project coordinator and lead developer for the PSIR module: Programming and Simulation environment for Industrial Robots. PSIR is founded on the CAD/CAM software PowerMILL [11], which is designed for generating toolpaths for CNC machines. PowerMILL is the world's leading specialist for NC CAM software for the manufacture of the complex shapes typically found in the aerospace, automotive, medical device and toolmaking industries. Robots have a different kinematic design and as such exhibit certain characteristics not found in CNC machines such as singular points, configurations and different movement behaviour. These characteristics need to be included in the holistic programming solution for robot machining. The PSIR module consists of four main topics:

- Toolpath calculation
- Robot pose calculation
- Optimisation and adaption
- Simulation, verification and post processing

This section of the paper gives an introduction to the four tasks and explains the holistic programming and simulation environment. The remainder of the paper describes an example of optimisation and adaptation of the robot program using kinematic compensation.

2.2 Adaption to robot kinematic

The basic required functionalities for a robot programming environment are similar to one for CNC machining centres: ability to calculate tool paths across a 3D part and ability to generate programs in the correct machine code through a post processor. However, one of the major differences is the kinematic machine design of a robot compared to a CNC machining centre. With a robot, a point in space can be reached by multiple robot poses. It is of importance that the programming and simulation of the robot poses match the one done by the real robot to come to a 100% correct robot program. Robot poses for a given tool path are calculated using a kinematic solver: by using an Inverse Kinematics transformation, the robot joint angle values are calculated for a given toolpath in Cartesian space. COMET focuses on serial robots with six rotational joints (6R). In the general case, the transformation of position from Cartesian to joint coordinates does not have a closed-form (i.e. single) solution, which makes the Inverse Kinematics a difficult problem to solve:

- Redundancy: a unique solution to the problem does not exist.
- The number of possible solutions increases with the number of DOF.
- The number of solutions depends of the number of Denavit-Hartenberg non-null parameters. For a 6R manipulator there are at most 16 solutions.

For the general 6R case, the only possibility is to use an iterative numerical solution. However, for a certain group of 6R robots, an analytical solution does exist. The group of 6R robots that have three consecutive axes intersect at a point can use the intersection point to separately solve the equations for the first three and the last three joints. This analytical method is called Pieper's method [12]. The analytical method has been implemented, resulting in a fast solver: 16.000 points processed in 3 s compared to 278 s using a general iterative solver. Another benefit of the analytical solution is the fact that all 16 solutions are found within a fraction of the time it would take the iterative solver to find one solution, opening the door for future optimisations.

2.3 Toolchain for CAD/CAM based offline-programming

The process chain for offline programming of robot machining is in fact comparable to normal CNC machines. The starting point is the customer's geometry for which the individual toolpaths are generated using the available strategies in PowerMILL. With the Inverse Kinematics (see section 2.2), each toolpath is processed to obtain the joint values for all robot joints. The robot movement can then be analysed by simulating the robot movements and detailed analysis using the graphs, reach limit analysis, singularity analysis and collision checking. Multiple tools exist to further fine-tune the robot movements, control the number of points on a toolpath, control the head movement, change axis priorities in order to optimise the movements and avoid singularities in each toolpath. The final stage in the programming chain is the post processing of the toolpaths into the specific robot language. Dedicated post processors are available in PowerMILL for all major robot brands.

ISBN 978-3-8007-3601-0

2.4 Adaptation of toolpath

As discussed in the introduction, robots have tremendous opportunities due to their relative costs, reach and scalability, for example when a robot is mounted on a linear rail. But robots lack stiffness, which can be a factor of 60 to 100 less compared to a CNC milling machine [13, 14]. The COMET approach aims at applying kinematic and dynamic models to compensate for certain errors that occur in robot machining. In the programming stage, an optimisation and adaptation step is added. Within the programming and simulation environment a custom interface is added, allowing researchers within the COMET consortium access to the toolpath and joint angle values and provides an interface to modify them based on a number of mathematical models. One example is the Cartesian compensation, where an error map of the robot, obtained with a laser tracker, is used to re-map the toolpath. Future models include the estimation of process forces and various stiffness models to counter the robot error sources.
The next sections present a specific example of modifying a robot program using a kinematic compensation.

3 Improving absolute accuracy by kinematic parameter optimisation for robot machining

One major source of errors when machining with industrial robots is their limited positioning accuracy. Whereas the repeatability is usually acc_{rep} < 100µm, the absolute accuracy acc_{abs} strongly depends on the precise description of the kinematic chain and can easily be in the range of millimeters. However, an optimisation of the kinematic parameters can lead to noticeable improvements [15]. Industrial robots are available in several shapes. Whereas robots with parallel kinematics are only visible in very specific applications, robots with serial kinematics are used in various applications because of their flexibility. The most common type is an industrial robot with a serial kinematic chain with six rotational joints. Therefor this type of robot will be examined in this paper.

3.1 Kinematic descriptions

In order to move robots cartesianly in space a transformation from the base of the robot to its effector is set up containing the geometry of the robot and the movements of the six rotational joints. The change from the frame of one joint to the following is described with one transformation. Popular descriptions of this transformation are the relative description [12], the parameter set defined by Denavit and Hartenberg [16] and a description introduced by Hayati [17]. Whereas Denavit, Hartenberg and Hayati use four parameters to describe this transformation, the relative description uses six parameters. Vaishnav and Magrab [18] even propose a model using nine parameters in order to take into consideration non-orthogonal systems. Whereas an increasing number of parameters permits to describe a system more detailed and more specific, for model identification there are other requirements. Schröer [19] describes how the number and the choice of parameters influence their identification. A set of four parameters is able to describe the transformation between rotational joints and at the same time allows good identification conditions. Depending on the configuration of the system different combinations of rotations and translations are needed. Denavit and Hartenberg [16] propose the transformation P_{DH}:

$$P_{DH} = R_z * T_z * T_x * R_x \tag{1}$$

with R_z a rotation around z-axis and T_z a translation along z-axis, T_x and R_x respectively. This representation has good identification properties for rotational joints with orthogonal rotation axes [19]. However for consecutive parallel rotation axes parameters numerical properties become worse. Hence Hayati proposed an alternative model providing good numerical properties for this case [17]:

$$P_{Hay} = R_z * T_x * R_x * R_y \tag{2}$$

However, this representation shows bad numerical properties for consecutive orthogonal rotation axes. Hence a sensible model of a 6-axes industrial robot must contain the representations of Denavit and Hartenberg P_{DH} and of Hayati P_{Hay} in order to enable good identification conditions. According to [19] good identification properties can be achieved with a combination of the descriptions of [16] and [17].

3.2 Identification of kinematic parameters

After setting up the kinematic transformation which describes the behavior of the robot precisely and at the same time enables good identification conditions, parameters g (containing e.g. joint angles or arm lengths) of the transformation need to be identified in order to set up an error compensation. In general two different approaches can be distinguished: Kinematic parameters can be identified in a closed kinematic chain (e.g. parallel kinematics) or in an open kinematic chain. Whereas the identification with a closed kinematic loop requires internal sensors measuring e.g. deformation, forces, torques or movements, the open kinematic chain requires an external measurement device capturing the end effector position. Note that each open kinematic chain can be transformed into a closed kinematics chain by clamping the open end to the base. Assuming a six axes serial robot closing the kinematic chain would require an additional clamping tool. Therefor the parameter identification with an open kinematic is considered in order to preserve the flexibility of the robot. The used measurement equipment is described in Section 4.1.

ISBN 978-3-8007-3601-0

By means of one taken measurement Pos_{meas} and using

$$m = Pos_{meas} - Pos_{bot} \tag{3}$$

with the robots internal position Pos_{bot}, constraints can be set up describing the unknown kinematic parameters g:

$$m = F(g) \tag{4}$$

where $F(g)$ represents a function of the kinematic parameters g. As in general one measurement supplies less conditions than unknown parameters, several measurements need to be taken at different robot poses. This overconstrained system can then be solved by minimizing a norm. Most commonly least squares are used. Alternatively a Kalmanfilter can also solve the problem [20].
In this case the pseudoinverse is used to solve the overconstrained system. Detailing equation 4 and separating $F(g)$ one receives for one measurement m_l

$$m_l = A_l(g_l)dg \tag{5}$$

where dg is the difference between the assumed and the true kinematic parameters and $A_l(g_l)$ is the Jacobian containing the derivatives of the measured coordinates with respect to the kinematic parameters [12].

$$A_l = \begin{pmatrix} \frac{\partial Pos_1(g)}{\partial g_1} & \cdots & \frac{\partial Pos_1(g)}{\partial g_i} \\ \vdots & & \vdots \\ \frac{\partial Pos_j(g)}{\partial g_1} & \cdots & \frac{\partial Pos_j(g)}{\partial g_i} \end{pmatrix} \tag{6}$$

with i the number of kinematic parameters and j the number of measurement signals. Using equation 5 for one measurement we receive for k measurements

$$M = Sdg \tag{7}$$

with

$$M = \begin{pmatrix} m_1 \\ \vdots \\ m_k \end{pmatrix} \tag{8}$$

and with the sensitivity matrix S

$$S = \begin{pmatrix} A_1(g_1) \\ \vdots \\ A_k(g_k) \end{pmatrix} \tag{9}$$

Using the pseudoinverse [21] S^+ of S leads to the optimised parameter offset

$$dg = S^+ M \tag{10}$$

The actual kinematic parameters can then be calculated as

$$g = g_{ini} + dg \tag{11}$$

with the initial kinematic parameters g_{ini}.

3.3 Evaluation of robot poses for identification

Nevertheless, an appropriate model and a good identification algorithm as described in section 3.1 and 3.2 do not guarantee good identification results. The identifiability of the kinematic parameters strongly depends on the chosen robot poses [15, 20, 22, 23]. Different observability indices have been proposed evaluating the setup conditions [20, 24]. Most of these indeces refer to the singular values of the sensitivity matrix S in equation 9.

4 Implementation and experimental results

The strategies presented in section 2 together with a calibration approach will be verified in experiments. Their single use as well as their combined application will be examined. The results will be compared to the uncompensated robot in order to show their performance. The following subsection will describe the components of the experiments and the experimental setup.

Figure 2: Experimental setup with KR125 and Leica Lasertracker

ISBN 978-3-8007-3601-0

4.1 Components and parameters

A robot cell is build up and a robot is equipped with the necessary hardware for milling. The model identification will be done in the target environment in order to cover all possible sources of errors. A KR125 robot from KUKA is used together with a TwinCAT CNC-control of BECKHOFF where updated kinematic parameters can be implemented easily and compensation is performed on the controller. As the robot is 13 years old, wear contributes to a limited precision in the case without compensation mechanisms. In **Figure 2** the full setup is shown.

An essential component of model identification is the measurement equipment. The quality of the identified models depends strongly on the precision of the measurement device. Especially due to the large work space of the robot and the required precision only a few devices are worth to be considered. A Leica Absolute Tracker AT901 is chosen as measurement device because of its three dimensional measurements, its tracking functionality and its error of $err_{LT} < 20\,\mu\text{m}$ for the chosen area. Two different configurations of milling robots can be identified. In one configuration the robot holds the workpiece and the spindle is attached to the ground. This configuration allows the robot to perform the handling and the machining operation at the same time. No additional device is needed to load and unload the robot. However, this configuration can only be realized with workpieces which do not exceed the work space and the maximum payload of the robot. In a second configuration the workpiece is attached to the ground and the spindle is attached to the end effector of the robot. In this configuration the robot can benefit from its great work space yet the handling cannot be performed by the robot itself. For the presented experiments this second configuration is chosen. The used spindle is a ZS80-H445.06 S19W2/2 of Alfred Jäger which can run 45.000 rpm and hence reduce forces on the robot which could lead to deviations. A 6 mm tool is chosen to mill a pocket in Ureol with 20.000 rpm.

4.2 Identification of the kinematic model

Robot positions used for identification of kinematic parameters are chosen so that they represent the work space which is relevant for machining. Due to geometric dependencies the identified parameters are limited to 6 joint angles and 4 arm lengths. A cube around the targeted area filled with a grid of points is chosen (see **Figure 3**). This strategy does not optimise the kinematic parameters for the full work space but for the area where milling will be performed. As only 2.5 D milling is considered, the end effector stays in the same orientation.

Figure 4 shows the subset of kinematic parameters which is subject to optimisation. As the robot's TCP is kept in the same orientation, dependencies between several parameters occur and therefor the following parameters are excluded from optimisation: L_4, L_6, L_7, L_8. A previous calibration procedure ensures the correct values of these parameters. The result of the optimisation procedure is presented in **Table 1**.

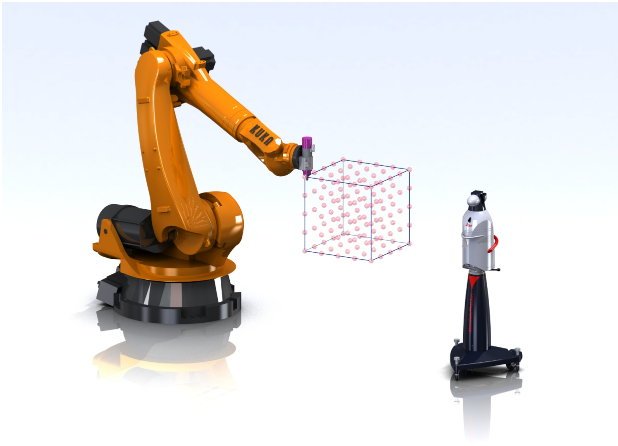

Figure 3: Identified grid describing the positioning error of the robot: Used for the identification of kinematic parameters

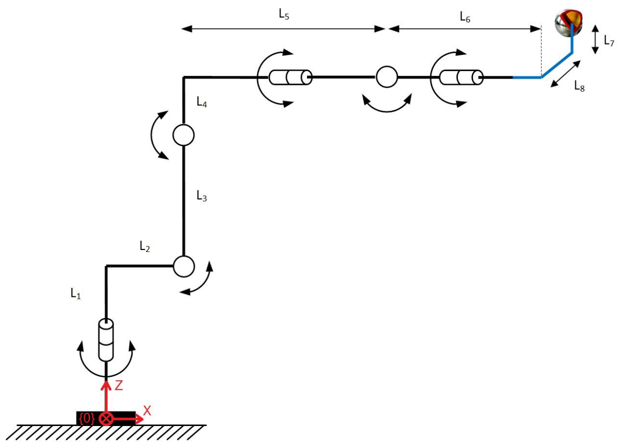

Figure 4: Chosen subset of kinematic parameters: Joint angles θ_i and arm lengths L_i

Kinematic parameters	Initial value	Offset
θ_1	-	+0.00917 °
θ_2	-	-0.10019 °
θ_3	-	+0.04798 °
θ_4	-	+0.05242 °
θ_5	-	-0.21941 °
θ_6	-	-0.21941 °
L_1	865 mm	+1.01343 mm
L_2	410 mm	+1.48882 mm
L_3	1000 mm	+0.39679 mm
L_5	1000 mm	-0.50421 mm

Table 1: Offsets of kinematic parameters resulting from optimisation

ISBN 978-3-8007-3601-0

4.3 Positioning improvements

The positioning accuracy for industrial robots, also named absolute accuracy, is defined in ISO 9283 [25]. The authors distinguish between repeatability and absolute accuracy. Repeatability is a limit which can only be overcome by using an external online measurement device. However, the absolute accuracy can be improved by kinematic optimisation. [25] defines the acurracy AP_P of point P as

$$AP_P = \sqrt{d_x^2 + d_y^2 + d_z^2} \tag{12}$$

with d_x, d_y, d_z the deviations of the measured point P. Together with the definition of accuracy, a representative set of measurement points is indicated (compare **Figure 5**). According to the procedure described in [25] measurements were performed for the uncompensated and the compensated robot (see **Table 2**).

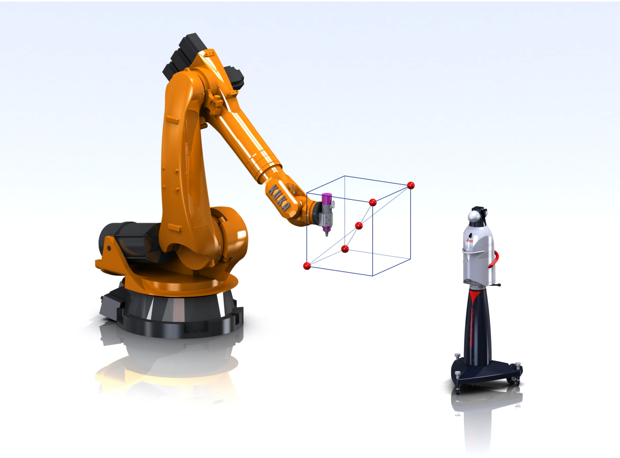

Figure 5: Measurement points representing the positioning accuracy of an industrial robot

	uncompensated robot	compensated robot
Point 1	5.70 mm	0.54 mm
Point 2	5.70 mm	0.15 mm
Point 3	5.64 mm	0.41 mm
Point 4	5.63 mm	0.43 mm
Point 5	5.53 mm	0.38 mm

Table 2: Positioning errors of the uncompensated and the compensated robot according to [25].

5 Conclusions

This article has investigated the impact of the combination of two different methods. An advanced programming and simulation environment is used to create an ideal path for robot machining while pointing out possible critical aspects of robot behavior. Discontinuities such as singularities as well as collisions are detected and provided to the user. Secondly a kinematic compensation approach is used to increase accuracy of industrial robots. This compensation allows to match the information generated by simulation precisely on the real robot cell without any manual adjustments. Improvements in the domain of accuracy are verified in experiments according to ISO 9283 [25]. It could be shown that the positioning error could be reduced by over 90% from > 5 mm to < 0.6 mm. Therefor a higher workpiece precision is enabled in machining which increases the range of applications.
The combination of the two methods allows an advanced planning containing detailed information about the robot behavior. Consequently this data is transferable to the real robot cell without any manual adaptations due to improved robot positioning accuracy.

Acknowledgments

This work is supported by the European Commission under the Seventh Framework Programme within the Project COMET under contract FoF.NMP.2010-1-258769. Further information about the project is available at: http://www.cometproject.eu/.

References

[1] U. Schneider, M. Ansaloni, M. Drust, F. Leali and A. Verl: *Experimental Investigation of Error Sources in Robot Machining.* Int. Conf. on Flexible Automation and Intelligent Manufacturing (FAIM), Porto, Portugal, pp. 14–26, 2013

[2] M. Weigold: *Kompensation der Werkzeugabdrängung bei der spanenden Bearbeitung mit Industrierobotern.* Dissertation, Schriftenreihe des PTW, Aachen, Shaker Verlag, 2008

[3] H. Zhan et al.: *Machining with Flexible Manipulator: Toward Improving Robotic Machining Performance.* International Conference on Advanced Intelligent Mechatronics. Monterey, California, USA, pp. 1127-1132, 24-28 July 2005

[4] G. Quintana and J. Ciurana: *Chatter in machining processes: A review.* International Journal of Machine Tools and Manufacture, Vol. 51, pp. 363-376, 2011

[5] E. Abele et al.: *Prediction of the Tool Displacement by Coupled Models of the Compliant Industrial Robot and the Milling Process.* Proceedings of the Int. Conf. on Process Machine Interactions, pp. 223-230, 2008

[6] C. Lehmann, B. Olofsson, K. Nilsson, M. Halbauer, M. Haage, A. Robertsson, O. Sörnmo, U. Berger: *Robot Joint Modeling and Parameter Identification Using the Clamping Method.* Proceedings of IFAC Conference on Manufacturing Modelling, Management and Control (MIM), Saint Petersburg, Russia, pp. 813-818, 2013

ISBN 978-3-8007-3601-0

[7] U. Heisel, F. Richter and K.H. Wurst: *Thermal Behavior of Industrial Robots and Possibilities for Error Compensation.* CIRP - Annals Manufacturing Technology, Vol. 46, No. 1, pp. 283-286, 1997

[8] C. Lehmann, M. Pellicciari, D. Manuel and J.W. Gunnink: *Machining with industrial robots: the COMET project approac.h* Int. Conf. on Flexible Automation and Intelligent Manufacturing (FAIM), Porto, Portugal, pp. 27-36, 2013

[9] K.-J. Conrad: *Taschenbuch der Werkzeugmaschinen* Fachbuchverlag Leipzig 2002

[10] U. Schneider, M. Drust, A. Puzik, A. Verl: *Compensation of errors in robot machining with a parallel 3D-piezo compensation mechanism.* 46th CIRP International Conference on Manufacturing Systems, pp. 305-310, 2013

[11] Delcam: *PowerMILL 2012, What's New guide.* 2011

[12] J.J. Craig: *Introduction to robotics: mechanics and control.* New York: Addison-Wesley, 1986

[13] Zengxi Pan, Hui Zhang: *Improving Robotic Machining Accuracy by Real-time Compensation.* ICROS-SICE International Joint Conference, 2009

[14] G. Vosniakos and E. Matsas: *Improving feasibility of robotic milling through robot placement optimisation.* Robotics and Computer-Integrated Manufacturing, Vol. 26, No. 5, pp. 517-525, 2010

[15] J.M. Hollerbach and C.W. Wampler: *The Calibration Index and Taxonomy for Robot Kinematic Calibration Methods.* The International Journal of Robotics Research, Vol. 15, pp. 573-591, 1996

[16] J. Denavit, R.S. Hartenberg: *A kinematic notation for lower-pair mechanisms based on matrices.* J. Appl. Mech., Vol.22, pp. 215-221, 1955

[17] S.A. Hayati, M. Mirmirani: *Improving the absolute positioning accuracy of robot manipulators.* J. Robot. Syst. 2, pp. 397-413, 1985

[18] R.N. Vaishnav and E.B. Magrab: *A General Procedure to Evaluate Robot Positioning Errors.* Int. J. of Robotics Research 6(1), pp. 59-74, 1987

[19] K. Schröer: *Identifikation von Kalibrationsparametern kinematischer Ketten.* Hanser, München, 1993

[20] B. Siciliano and O. Khatib: *Handbook of Robotics.* Springer-Verlag, New York, 2008

[21] R. Penrose: *A generalized inverse for matrices.* Proceedings of the Cambridge Philosophical Society 51(3), pp. 406-413, 1954

[22] D. Daney, B. Madeline, Y. Papegay: *Choosing Measurement Poses for Robot Calibration with the Local Convergence Method and Tabu Search.* Int. J. Robot. Res. 24(6), pp. 501-518, 2005

[23] Li Tian et al.: *Optimal measurement configurations for kinematic calibration of six-DOF serial robot.* Journal of Central South University of Technology, Vol. 18(3), 2011

[24] Yu Sun and J.M. Hollerbach: *Observability Index Selection for Robot Calibration.* International Conference on Robotics and Automation, 2008

[25] DIN EN ISO 9283: *Manipulating industrial robots Performance criteria and related test methods.* International Organization for Standardization, 1998

ISBN 978-3-8007-3601-0