2018 26th Mediterranean Conference on Control and Automation (MED)
Zadar, Croatia, June 19-22, 2018

On-line Estimation of F/T Sensor Offset for Arbitrary Orientation of Robot Tool by Evaluating Two Machine Learning Algorithms

Matko Kulaš, Zdenko Kovačić, *Senior Member, IEEE* and Matko Orsag, *Member*, *IEEE*

Abstract— **The offset in the output of a force/torque (F/T) sensor embedded in the robot's wrist is usually handled by deducting the measured force value when the robot tool encounters the surface and achieves the desired orientation. The problem becomes more complicated when the tool's orientation changes continuously or the tool is changed during the robot's work. This paper explored the possibility of using two well-affirmed machine learning algorithms (k-nearest neighbors and polynomial regression) for on-line estimation of F/T sensor offset. Both machine learning algorithms were implemented and evaluated on the state-of-the-art 6-DOF robot arm Schunk LWA4P equipped with the OptoForce HEX-70-CE-2000N F/T sensor. Evaluation of the two on-line offset estimation methods was made during a force-controlled writing on the office board with different robot tool orientations. It was found that the frequency of generating the offset estimate is much higher with the polynomial regression algorithm.**

Keywords: Force control, sensor offset, robot, machine learning, on-line offset estimation, variable orientation

I. Introduction

Soft robotic solutions penetrate all branches of modern robotics and regardless of the robotic application being studied, the problem of force sensor offsets becomes a universal problem. In most cases, this problem is only solved by deducting the sensor output value measured when the robot tool encountered the environment and reached the desired orientation. This simple solution works well only if the robot tool's orientation remains constant throughout the entire task execution. But even if that is the case, there are other applications (e.g. CNC machines with automatic tool changers), where the attachment of different tools results in different sensor offsets.

Machine learning algorithms are applied in many robotic systems embracing areas like computer vision (object classification), imitation learning (humanoid robots, collaborative industrial robots), assistive technologies (rehabilitation, therapy), collective learning (multi-agent systems, robotic societies) and many others [1-4]. For example, an Associative Search Network (ASN) is used to cope with the contact task problem having the specific trajectory and force direction constraints [5]. Machine learning is present in force-guarded and force-guided robotic applications, with accent on neural networks [6, 7].

As noted in [8], if a load (e.g. tool) is attached to the force sensor, the measured forces/torques are not only caused by contact of the load with the environment but also by the inertial forces, centrifugal forces, Coriolis forces, and associated torques the moving load exerts onto the sensor. The recursive total least-squares approach taken in [9] aimed to estimate on-line inertial parameters of the load. That problem was treated in [10] in a way that the proposed impedance controller accurately shaped the impedance of a manipulator carrying a relatively heavy load. Estimation of dynamic contact force/torque was proposed in [11] for improved interaction control. Walk-through programming for admittance controlled robots is one of use cases which also requires tool compensation [12, 13]. The way of tackling the *in-situ* calibration of six-axes force torque sensors by using accelerometer measurements is presented in [14]. Further improvement is presented in [15] where a model based approach is used and the calibration matrix estimation problem is formulated as a regularized least square problem.

The work described in this paper is aimed at solving the problem of estimating and compensating for the force sensor offset in the case of slower operations such as cutting, grinding or writing, where the robot tool's orientation changes in 3D space, creating continuous changes of sensor offset due to the dominant influence of gravity force. If in such cases a fine control of force is required, only the deduction of the offset value detected during the initial contact of the tool jeopardizes the fineness of the force control and creates possible negative effects on the quality of the task. During conducting this research aimed to solve the practical aspect of the problem of on-line offset estimation, it has been found that the estimation of the force sensor offset can be formulated as a regression problem by modeling the dependence between the mounted tool, its current orientation and the force output, thus obtaining two well-affirmed machine learning algorithms, the *k-nearest neighbors* and *polynomial regression*, suitable for deployment [16, 17]. The aim is to find optimal hyperparameters that give the best offset estimates with these two algorithms and to evaluate their maximum frequency of generating estimates for on-line compensation.

The paper is organized in the following way. First, we give the description of a robot system and then we describe the two machine learning algorithms selected for implementation of an on-line estimation of the force sensor offset. Further on, we analyze the experimental results obtained during the tasks of writing in two different orientations, on the vertically and the horizontally positioned office board. We conclude the paper with final comments and ideas for future work.

M. Kulaš, Z. Kovačić and M. Orsag are with the Faculty of Electrical Engineering and Computing, University of Zagreb, Zagreb, Croatia (e-mail: {matko.kulas; zdenko.kovacic; matko.orsag}@fer.hr).

II. ROBOT SYSTEM

The work described here is motivated by previous work on a force-controlled Cartesian industrial robot [18], where a tool's approach vector was constantly normal to a flat horizontal work surface. This enabled simple compensation of the force sensor offset, because the offset was constantly the same. By using an articulated 6-DOF manipulator operating in the 3D space, it has become necessary to estimate and compensate a deviating force sensor offset value. The robot used in experiments is a force-controlled articulated 6-DOF Schunk LWA4P manipulator (see Fig. 1).

The task of writing on the office board with a marker changing its orientation with respect to the work environment caused the appearance of variable offset in the output of the force sensor. This has influenced the quality of writing a lot, especially provided the reference forces should not be larger than 1 *N*. In order to allow controlled writing on the office board arbitrarily located in the 3D space, the reference coordinate systems on the board and on the robot are set up and their spatial connection is described using homogeneous transformation matrices and Euler angles.

Figure 1: 6DOF articulated robotic arm for writing on the board.

The sensor used in the force control loop is the 6 axis F/T sensor OptoForce HEX-70-CE-2000N, mounted at the end of a robotic arm, underneath a tool that allows us to measure force and torque in all directions. Sensor characteristics are shown in Tab. 1. An analog signal obtained at the output of the force sensor is filtered due to the presence of noise. A standard PID controller with anti-windup property is used to control the force.

Table 1. Characteristics of a force sensor OptoForce HEX-70-CE-2000N

	Fz compress	*Fz* tension	*Fxy*	*Txy*	*Tz*
Nominal capacity	2000 N	800 N	300 N	15 Nm	10 Nm
Resolution (N.C.)	6000	-	± 8000	± 8000	± 8000
Overload	200 %	200 %	200 %	200 %	200 %
Deformation (N.C.)	1.5 mm	1 mm	± 1.5 mm	±2°	±5°
Sampling frequency	1000 Hz	1000 Hz	1000 Hz	1000 Hz	1000 Hz

The robot system is controlled through the ROS (Robot Operating System) environment. Communication is achieved via the CANopen communication protocol (data transfer rate is 500 kB). The ROS package that takes the trajectory points and manages the communication is *schunk_canopen_driver*.

III. MACHINE LEARNING ALGORITHMS

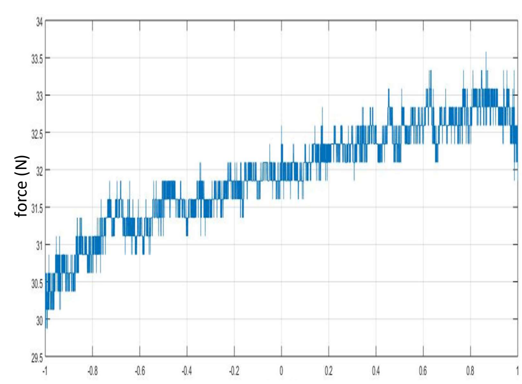
As already mentioned in the introductory section, the estimation of the force sensor offset can be formulated as a regression problem by modeling the dependence between the mounted tool, its current orientation and the force output. Two machine learning algorithms, *k*-nearest neighbors and polynomial regression are considered suitable for possible implementation. Since the goal is to evaluate the suitability of algorithms for on-line estimation, we need to find optimal hyperparameters that give the best offset estimates with these two algorithms.

A. Collecting a set of training data

The selected machine learning algorithms are trained on the set of data *D*. Offset on the sensor depends on the current tool mounted on the robot, and its orientation relative to the *z*-axis (approach vector). This state is read from the homogeneous transformation matrix as the third element of the third column (S stands short for sin, C for cos):

$$\begin{bmatrix} C(\varphi)S(\theta)C(\psi)+S(\varphi)S(\psi) \\ S(\varphi)S(\theta)C(\psi)-C(\varphi)S(\psi) \\ C(\theta)C(\psi) \end{bmatrix} \tag{1}$$

For collecting a set of training data, the robot is positioned in a configuration where its tool orientation has value z = 1. During slow movement, using only the fifth robot joint, towards orientation where z = -1, data are collected in pairs, including orientation opposite of *z*-axes and current raw (no filtered) force reading from the sensor. The data acquisition routine stores one pair of data in the memory every 20 *ms* (50 *Hz*). To eliminate other influences on the sensor like Coriolis, centrifugal and inertial forces, the data acquisition process lasts about 400 *s* collecting almost *N* = 21.000 data pairs. The acquired raw data are used for training, while the filtered data are used for control (Fig. 2).



a) Raw force measurement data

force (N)

tool orientation

b) Filtered force measurements data

Figure 2: Data set acquired for training of machine learning algorithms

B. *k-nearest-neighbors algorithm (k-NN)*

The *k*-nearest-neighbors (from now referred to as *k*-NN) is a non-parametric classification algorithm used in the regression for force offset estimation. The idea behind the *k*-NN algorithm is to take an input x and to locate k nearest points in the data set *D*. After locating the points, the mean of output pairs is calculated. There are more criteria for finding neighbors (i.e. number k as a hyperparameter), the most popular being Euclidean distance and the use of kernels.

Then, the output estimation can be calculated as the mean of the sum of the k nearest points in the whole data set *D*:

$$h(x) = \frac{1}{k} \sum_{(x^{(i)},y^{(i)}) \in NN_k(x)} y^{(i)} \tag{2}$$

The error function is computed to validate the algorithm:

$$E(h|D) = \frac{1}{2}\sum_{i=1}^{N}(y^{(i)} - h(x^{(i)}))^2 \tag{3}$$

The search for hyperparameters k in *k*-NN is done by computing the error function (3) for different values *k*. If a search is performed on the test data set, it is referred to as a cross-validation-search, but there is no training done in *k*-NN, so an error is calculated across the entire data set *D*. The result is shown in Fig. 3, and it is clear that the best value k for this data set is 11, achieving an average error of 0.0073 *N*. The maximum error on this set of data is 0.2118 *N*.

The results of the *k*-NN algorithm are shown in Fig. 4. The *k*-NN algorithm usually has a problem called a curse of dimensionality, which occurs when the number of input features n is raised. The points then become too distant, and the distance is usually non-discriminatory, but in this particular problem, this is not the case because of offset's dependence on only one variable, the orientation relative to the *z*-axis, making $n = 1$.

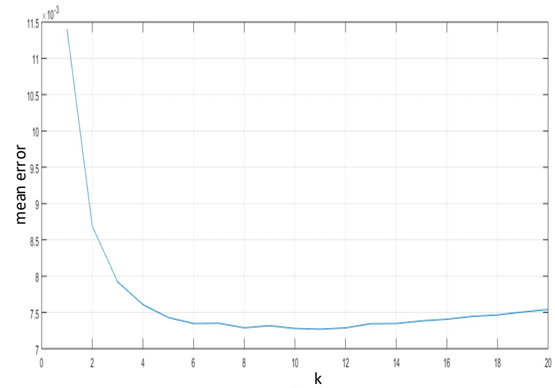


Figure 3: Cross validation graph for the best *k* of *k*-NN

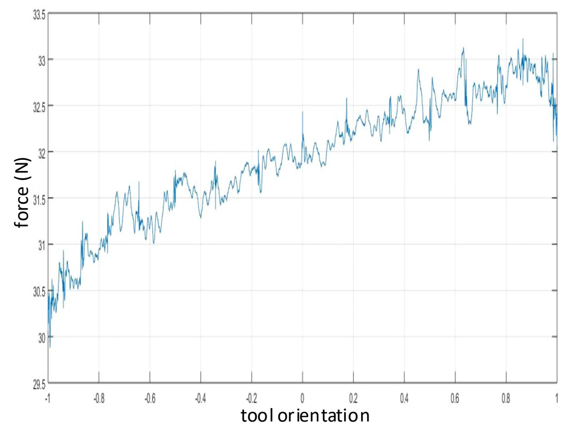


Figure 4: *k*-NN algorithm force offset estimation with *k*=11

C. *Polynomial regression algorithm*

The algorithm used for polynomial regression is a parametric nonlinear univariate regression form, meaning that the output space is f(x) = y, and the feature space is greater than 1 ($n > 1$). In order to achieve nonlinearity, the input space is mapped to a higher dimension of space by means of the following basis function:

$$\begin{gathered} \boldsymbol{\phi} : \mathbb{R}^n \rightarrow \mathbb{R}^{m+1} \\ \boldsymbol{\phi}(x) = [\phi_0(x), \ldots, \phi_m(x)] = [1, x, x^2, \ldots, x^m] \end{gathered} \tag{4}$$

The polynomial regression model for the output estimation is given with:

$$h(\boldsymbol{\phi}; \mathbf{w}) = w_0 + w_1\phi_1 + \cdots + w_m\phi_m = \boldsymbol{w}^T\boldsymbol{\phi} \tag{5}$$

where $\mathbf{w}$ is a vector of polynomial coefficients that by multiplying the input variable maps the space of input data set into the output space. The error function (3) is used to compute the coefficients, where x is replaced by $\boldsymbol{\phi}$. By using the ordinary least square (OLS) optimization method in the matrix form,

$$\begin{gathered} E(\boldsymbol{w}|D) = \frac{1}{2}(\boldsymbol{\phi w} - \boldsymbol{y})^T(\boldsymbol{\phi w} - \boldsymbol{y}) \\ \boldsymbol{\nabla}_w E = \boldsymbol{\phi}^T\boldsymbol{\phi w} - \boldsymbol{\phi}^T\boldsymbol{y} = 0 \end{gathered} \tag{6}$$

a closed form solution is computed

$$\boldsymbol{w} = (\boldsymbol{\phi}^T\boldsymbol{\phi})^{-\mathbf{1}}\boldsymbol{\phi}^T\boldsymbol{y} = \boldsymbol{\phi}^{+}\boldsymbol{y} \tag{7}$$

where $\boldsymbol{\phi}^{+}$ is a *Moore-Penrose* pseudoinverse. To prevent the matrix singularity $\boldsymbol{\phi}^T\boldsymbol{\phi}$, L2-regularization is introduced, which adds a small λ constant multiplied by the eye matrix without the first element before the inversion. Some of the resulting polynomials are shown in Fig. 5.

Cross-validation is used to find the hyperparameter *m*, which indicates the degree of polynomial used in mapping to a higher space dimension. Data set *D* is permutated and separated into two sets of points, a set for training and a set for testing. Coefficients of polynomials with different degrees were calculated from the closed form solution (7) and tested with the testing data set by computing the error (3). The results of cross-validation are shown in Fig. 6. One can see that even for polynomials up to the 34th degree, the error is still falling, but the computational complexity becomes very expensive from the on-line estimation point of view. Therefore, the highest degree of polynomial *m*=34 is taken as optimal, with an average error of 0.0164 *N* and the biggest error 0.2466 *N*. This algorithm gave a bit worse result than the *k*-NN algorithm in regard to estimation accuracy, but it outperformed the *k*-NN algorithm in regard to the estimation frequency (50 *Hz* compared to *k*-NN's 30 *Hz*). The resulting optimal degree polynomial of the 34th degree is shown in Fig. 7 together with the filtered data set of acquired force measurements.

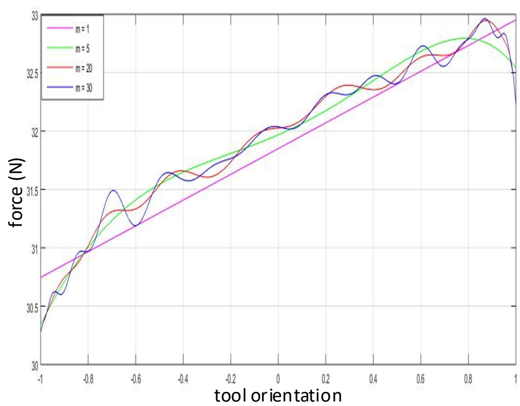


Figure 5: Different degree functions for polynomial regression

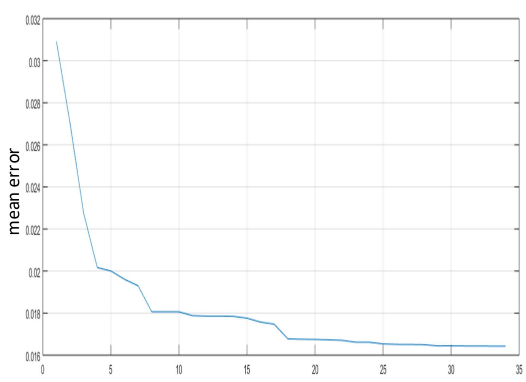


Figure 6: Cross validation graph for polynomial regression

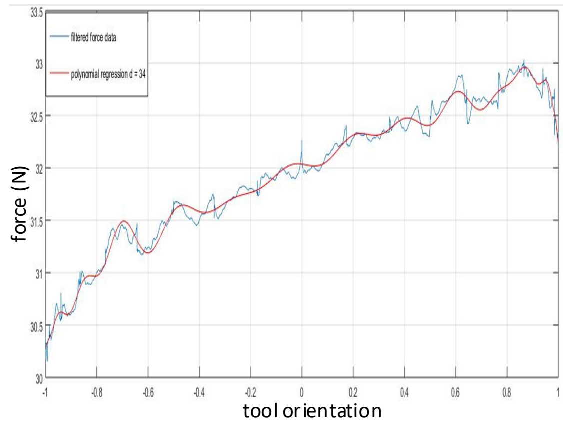


Figure 7: Filtered force and polynomial regression estimation for *m*=34

IV. Experimental Results

The aim of experiments is to validate the performance of the *k*-NN and polynomial regression algorithms in two tasks of writing on the vertically and the horizontally positioned board (see Fig. 8), where the impact of gravity on the force sensor offset varies the most. Once the contact of the tool was established, the sequence of stepwise force reference values of 10 *N*, 20 *N* and 30 *N* and back (*cake test*) was applied to test the performance of two estimation algorithms.

Figure 8: Two robot tasks: left) writing on a vertical work surface, right) writing on a horizontal work surface

The first experiment was conducted during writing on the vertical work surface with offset estimation made by the polynomial regression algorithm (see Fig. 9). One can see that the estimated value of the force sensor offset is around 32 *N*, which matches the estimated value shown previously in Fig. 7, confirming that the algorithm is working well also in the case of closed-loop force control.

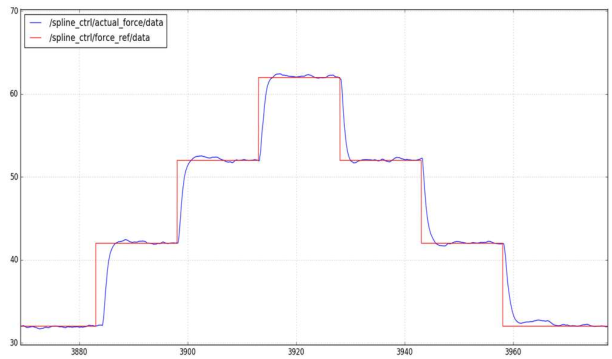


Figure 9: Test for a polynomial regression force offset estimation algorithm in case of writing on the vertically positioned work surface. The initial value of 32 *N* corresponds to the previously estimated force sensor offset value.

The following test was conducted in the same vertical position of the work surface but with the *k*-NN algorithm running (see Fig. 10). One can see that the initially estimated value of force sensor offset is again around 32 *N*, proving that the *k*-NN estimation algorithm also works well in the closed-loop force control conditions. It should be noted that during the force control with the *k*-NN algorithm, the estimation frequency in the corresponding ROS node was set to 30 *Hz* – the highest possible frequency at which the algorithm was able to work on-line.

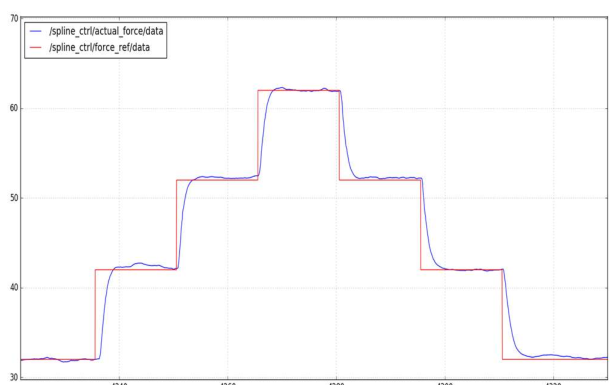


Figure 10: Test for a *k*-NN force offset estimation algorithm in case of writing on the vertically positioned work surface. The initial value of 32 *N* corresponds to the previously estimated force sensor offset value.

This finding was the reason to continue the next experiment of force-controlled writing on the horizontal surface with the polynomial regression algorithm only (Fig.11). One can see that the estimated value of the force sensor offset is around 30 *N*, which differs about 2 *N* from the sensor offset value estimated in the previous experiments. If the writing tool would be made of metal rather than of plastic, this difference between the estimated values would be larger. The difference in offsets of 2 *N* can have a negative impact on the closed-loop force control if a reference force value does not have an appropriate (adequately greater) value or if one wants to achieve very precise contact force control.

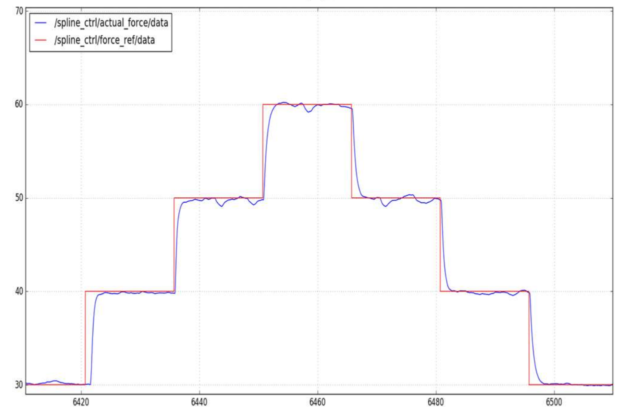


Figure 11: Test for a polynomial regression force offset estimation algorithm in case of writing on the horizontally positioned work surface. The initial value of 30 *N* differs 2 *N* from the offset value in the vertical plane.

In the final experiment (Fig. 12), the task of writing on the vertically positioned office board with the plastic marker was executed with the desired contact force reference of 0.75 *N*. The whole trajectory for writing the word “Schunk” lasted about 200 *s*. The trajectory planning for the text being displayed was performed with maximum speed of 1 m/s and maximum acceleration of 1.2 m/s^2. The controlled force response is shown in Fig. 13. One can notice that the control is fine enough to maintain the desired contact force value during the entire process of writing a given word.

Figure 12: The task of writing on the vertical board with the writing force reference value of 0.75 *N*.

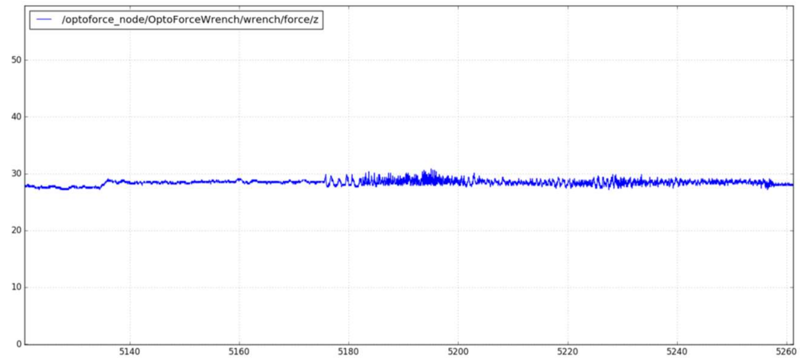


Figure 13. The response of the contact force during writing the letter S of the word shown in Fig. 12. The force is maintained at the desired value 0.75 N, with the force offset value clearly estimated at the beginning of writing.

V. Conclusion

The purpose of this work was to explore the possibility of developing and implementing machine learning algorithms for estimating and removing the force sensor offset that is present during execution of a robot's task in the presence of continuous contact with the environment. Technically, the main goal was to estimate offset for different tool orientations in 3D space, making the estimation algorithms applicable for arbitrary tool orientations in various force control-based robotic applications.

Having in mind the requirement for on-line estimation of the force sensor offset, two already well-affirmed machine learning algorithms were selected as possibly suitable for deployment - the *k-nearest neighbors* and *polynomial regression* algorithms. Though both algorithms proved the ability of accurate estimation of offset, the *k*-NN algorithm was a bit more accurate, but slower, and from the on-line estimation point of view, it did not fulfill the real-time execution requirements (maximum frequency reached only 30 *Hz*). The polynomial regression algorithm met the requirements and based on the analysis of experimental results made, the algorithm is recommended for machine learning-based estimations provided the dimension of input variables remains low. Cross-validation has shown that even better results can be achieved by higher-order polynomials only if this is not limited by the computing performance of state-of-the-art computer devices.

Future efforts will be directed towards the organization of control tests on uneven work surfaces that require a continuous change in the orientation of the robot tool parallel to continuous force control.

References

[1] S. G. Khan, G. Hermann, F.L. Lewis, T. Pipe, C. Mellhuish, "Reinforcement learning and optimal adaptive control: An overview and implementation examples", *Annual Reviews in Control*, Volume 36, Issue 1, pp. 42-59, April 2012.

[2] E. Gribovskaya, A. Kheddar, A. Billard, "Motion learning and adaptive impedance for robot control during physical interaction with humans", IEEE International Conference on Robotics and Automation ICRA 2011, pp. 4326-4332, 2011.

[3] J. Buchli, E. Evangelos Theodorou, F. Stulp, and S. Schaal, "Variable impedance control - a reinforcement learning approach," in Robotics Science and Systems, 2010.

[4] M Kalakrishnan, "Learning Force Control Policies for Compliant Manipulation", 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 4639-4644, 2011.

[5] B. Kim, S. Park, "Impedance Control of Robot Manipulator in Contact Task Using Machine Learning", International Joint Conference SICE-ICASE, pp. 2590-2594, 2006.

[6] Z. Yang, M. Li, "A Method on Impedance Control of Robots Based on the Neural Network", Intl. Conf. on Intelligent Information Hiding and Multimedia Signal Processing IIHMSP '08, pp. 1433-1436, 2008.

[7] S. Jung, T.C. Hsia, "Neural Network Impedance Force Control of Robot Manipulator", IEEE Transactions on Industrial Electronics, Vol. 45, No. 3, pp. 451-461, June 1998.

[8] D. Kubus, T. Kroger, F.M. Wahl, "Improving force control performance by computational elimination of non-contact forces/torques", IEEE International Conference on Robotics and Automation ICRA 2008, pp. 2617-2622, 2008.

[9] D. Kubus, T. Kroger, F. M. Wahl, "On-line estimation of inertial parameters using a recursive total least-squares approach", IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pp. 3845-3852, 2008.

[10] F. Aghili, "Impedance control of manipulators carrying a heavy payload", IEEE/RSJ International Conference on Intelligent Robots and Systems IROS 2009, pp. 3410-3415, 2009.

[11] G. Bätz, B. Weber, M. Scheint, D. Wollherr, M. Buss, "Dynamic contact force/torque observer: Sensor fusion for improved interaction control", *The International Journal of Robotics Research*, vol. 32, pp. 446, 2013.

[12] C. Talignani Landi, F. Ferraguti, C. Secchi, C. Fantuzzi, "Tool compensation in walk-through programming for admittance-controlled robots", 42nd Annual Conference of the IEEE Industrial Electronics Society IECON 2016, pp. 5335-5340, 2016.

[13] F. Ferraguti, C. Talignani Landi, C. Secchi, C. Fantuzzi, M. Nolli, M. Pesamosca, "Walk-through Programming for Industrial Applications", *Procedia Manufacturing*, vol. 11, pp. 31, 2017.

[14] S. Traversaro, D. Pucci, F. Nori, "In Situ Calibration of Six-Axis Force-Torque Sensors using Accelerometer Measurements", IEEE Int. Conf. on Robotics and Automation (ICRA), pp. 2111 – 2116, 2015.

[15] F.J.A. Chavez, S. Traversaro, D. Pucci, F. Nori, "Model Based In Situ Calibration of Six Axis Force Torque Sensors", 16th IEEE Int. Conf. on Humanoid Robots (Humanoids), pp. 422-427, Nov 15-17, 2016.

[16] T.M. Cover, P.E. Hart, "Nearest neighbor pattern classification". *IEEE Transactions on Information Theory*, 13 (1): pp. 21–27, 1967.

[17] A. Pečkov, "A Machine Learning Approach to Polynomial Regression", PhD Thesis, Jožef Stefan International Postgraduate School, Ljubljana, Slovenia, 2012.

[18] V. Brkić, Z. Kovačić, "Cartesian Robot Operation with Fuzzy Adaptive Force Control", The 25th Mediterranean Conference on Control and Automation, Valletta, Malta, 352-357, 2017.