

# IMPEDANCE CONTROLLERS FOR ELECTRIC-DRIVEN ROBOTS

Submitted: 15<sup>th</sup> January 2013; accepted: 3<sup>rd</sup> June 2013

*Edward Jeziarski, Artur Gmerek*

## Abstract:

*This article presents a proposal of impedance controller, which is able to influence the compliance of a robot, based on information obtained from a low-level controller about robot interaction with the environment. The proposed system consists of a low-level controller with proximity sensor, based on which mechanical impedance is adjusted. The intention of the creators was to develop a universal impedance regulator in the sense of hardware and software layers. Because of TCP/IP architecture, the designed regulator can be easily adapted to different robot controllers. Experiments were conducted on a real 1-DOF manipulator driven by BLDC motor, as well as using simulation on a 2DOF planar robot.*

**Keywords:** *impedance controller, admittance controller, 2DOF manipulator, BLDC drive*

## 1. Introduction

Nowadays, in an industrial environment, we often deal with positional robot control. This type of control takes place when the manipulator is highly rigid, thus the kinematic chain is characterized by high impedance. However sometimes is needed to soften a kinematic chain. In this situation impedance control can be used. A robot's mechanical impedance is usually equated with the impedance of its mechanical part. The concept of impedance control was first introduced to the field of robotics by Hogan [3]. If we assume that we are utilizing the analogy of velocity-current [5, 6], then mechanical impedance can be defined with a greater precision. Namely, after the adaption of an incremental linear dynamical model in the vicinity of a chosen operating point, the mechanical impedance of the kinematic chain is defined as the Laplace transform of the force exerted by the end-effector of the manipulator to the Laplace transform of effectors' velocity.

Proper control of a robot's actuators can change the impedance of its kinematic chain. It is relatively easily manageable in the case of pneumatic or hydraulic actuators [2]. Their rigidity is naturally related to energy-transferring media. In the case of electric drives, it should be utilized to control kinematic chain stiffening or softening with the help of feedback mechanisms. In this process, it is necessary to anticipate how the manipulator will interact with the environment, as changes in impedance cannot be infinitely fast. This is due to the fact that electric actuators are stiff according to the configuration of drives (usually high velocity and high gear ratio).

In most cases interaction with the environment must be detected early enough to enable the controller to respond effectively. This can be achieved by gathering information about the environment, obtained from the robot's external sensors.

The article proposes a system that makes it easy and efficient – to connect the manipulator controller with a proximity sensor. It consists of a low-level driver which works with the designed computer system. The data collected by the sensor is sent to the impedance controller software which processes the data and controls an electric drive associated with controller in order to obtain desired flexibility. The manipulator can thus be prepared to interact with an external object to avoid harmful collisions. Experiments were done on robot with one degree of freedom (DOF), as well as using simulation on a 2DOF planar robot.

Impedance control is currently under development in a growing number of research centers. Both stiffness and damping adjusters can be viewed as primitive impedance drivers. There are only a few studies describe how to interact with the control system with the use of external sensors in order to optimize manipulator impedance. Interaction with the environment is mostly affected by the use of force sensors [1, 7]. However, force sensors are effective only when the manipulator's movements are sufficiently slow. If manipulator velocity is high, potential collisions with a rigid environment should be detected in advance. This can be accomplished with the help of proximity sensors, radars or vision systems. An example of this type of solution is described in [9]. The authors used information from the vision system to impact manipulator impedance. Impedance was switched abruptly when the robot approached an object.

## 2. Impedance Controller

The manipulator's desirable properties are determined by the environment and the features of the robot controller. Usually it is assumed that if the robot is close to a person or detects an undesirable object on its planned trajectory, the susceptibility of the kinematic chain should increase. When the sensor detects an approaching object, the controller changes the manipulator's impedance, depending on the distances between the robot's arm and an object.

The system presented in this article consists of a brushless DC motor (BLDC) which drives the robot link, an inductive proximity sensor with a low-level controller and a computer system that processes the data and controls the motor. The computer system

consists of two programs. The first captures and processes data from the low-level driver, while the second controls the motor (Fig. 1).

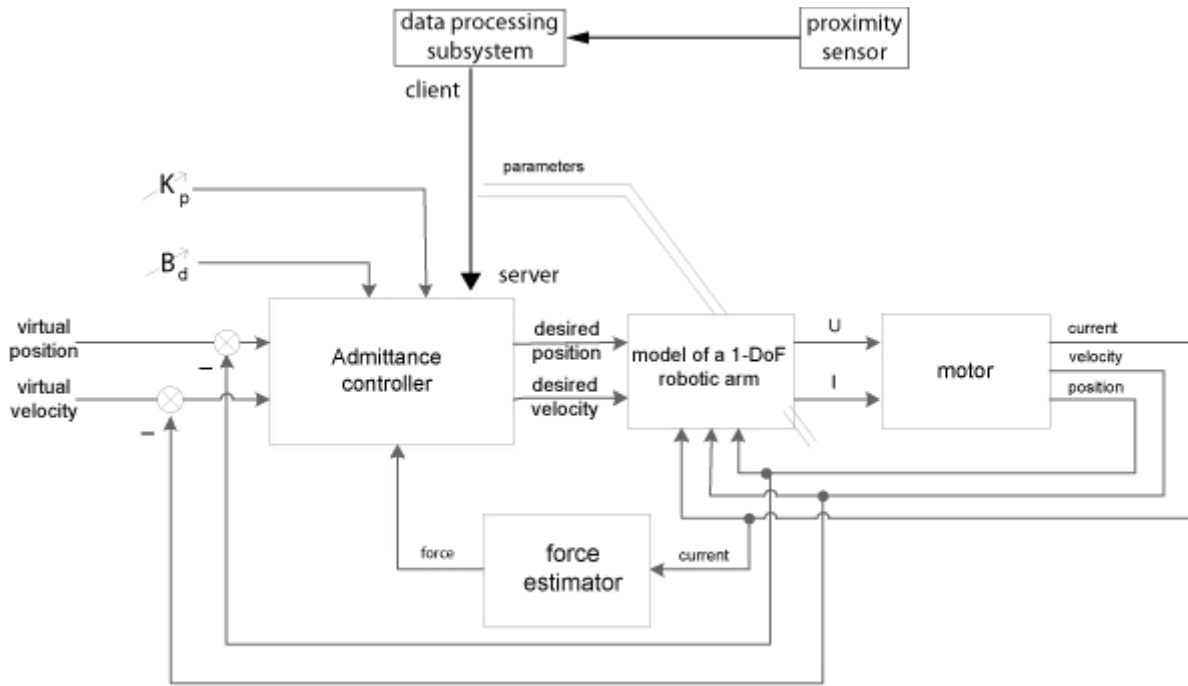


Fig. 1. The overall block diagram of the control system for BLDC motor

## 2.1 The Impedance Driver

Description of the dynamics of the manipulator in contact with the environment can be described as follows:

$$B(q)\ddot{q} + C(q, \dot{q})\dot{q} + h(q) + f(q, \dot{q}) = \tau - F_{ext} \quad (1)$$

where  $B$  is an inertia matrix,  $C$  – a matrix of velocity couplings,  $h$  – a vector of gravitational forces,  $f$  – a friction vector,  $\tau$  – a vector of control torques and forces,  $F_{ext}$  – a vector of external forces. The vector of external forces acting on the manipulator can be approximated by means of the mass-spring-damper system, it gives:

$$-F_{ext} = M(\ddot{q} - \ddot{q}_d) + B_d(\dot{q} - \dot{q}_d) + K_p(q - q_d) \quad (2)$$

where  $M$  – is the environment's inertia,  $B_d$  – is damping and  $K_p$  is stiffness. Transforming equation (2) to determine joint acceleration we get:

$$\ddot{q} = \ddot{q}_d - M^{-1}(F_{ext} + B_d(\dot{q} - \dot{q}_d) + K_p(q - q_d)) \quad (3)$$

substituting (3) to (1) we get:

$$B(q)(\ddot{q}_d - M^{-1}(F_{ext} + B_d(\dot{q} - \dot{q}_d) + K_p(q - q_d))) + C(q, \dot{q})\dot{q} + h(q) + f(q, \dot{q}) + F_{ext} = \tau \quad (4)$$

In the case of a 1DOF manipulator, the last equation can be written as:

$$\left(\frac{1}{2}m_2l^2 + \frac{1}{2}m_1R^2\right)(\ddot{q}_d - M^{-1}(F_{ext} + B_d(\dot{q} - \dot{q}_d) + K_p(q - q_d))) + \frac{1}{2}m_2gl\cos(q) + f(q, \dot{q}) + F_{ext} = \tau \quad (5)$$

The equation can be used to control the robot by the means of “the computed torque feedforward control”. Equation (5) shows that the response of a conventional driver's impedance depends on the value of acceleration, velocity and position, as well as the value of the force controller, set controller parameters ( $M$ ,  $K_p$  and  $B_d$ ) and the object's dynamic parameters ( $B$ ,  $C$ ,  $h$ ,  $f$ ).

However, the equation fulfills its function only when the manipulator actively interacts with the environment. Due to the fact that the force can be recorded only when the manipulator establishes contact with the environment it was decided to use virtual force instead of actual force. The equation for the modified force can be expressed as follows:

$$F_{ext} = \mu a \quad (6)$$

where  $a$  is an estimated position based on information obtained from the position sensor, and  $\mu$  [N/m] is the proportionality parameter (obtained by heuristic methods).

The equation shows that the reaction of the manipulator does not depend on torque value but is associated with the distance from the object to the manipulator. The proportionality factor has been selected to provide an appropriate response when approaching the outer arm to the object.

### 3. Implementation of Hardware and Software

The aim of the project was to design a system that will enable efficient implementation of impedance control with the help of different types of physical hardware. Problems with the integration of various systems occur frequently. In the case being described, a BLDC electric motor was controlled in real time using the *Python* interpreter, which cooperates with a data acquisition card, as well as drivers which are dedicated to work with *C* compiled programs or other compilers. It was therefore necessary to develop software which would ensure proper operation of the controller.

The problem of software integration occurs very often when creating systems of greater complexity. There are several methods to combine various programs to form a common system on platforms such

the popular ATmega8 Atmel microcontrollers. Drivers communicated with the computer via a USB-UART protocol.

### 4. Experiments and conclusions

#### 4.1 Experiments on Object with One Degree of Freedom

One of the final experiments was carried out on the model with one degree of freedom (Fig. 3).

In order to determine the optimal control parameters several experiments were performed to illustrate how changes in stiffness and damping coefficients affect the system's properties. When the manipulator stiffness decreases, it no longer tracks the desired trajectory, as opposed to situations when rigidity is high. This principle was used to determine the values of stiffness and damping parameters that cause a desired change in the robot's operation.

```

C:\Documents and Settings\ARR\Moje dokumenty\Wisia...
wys lano 5537.
wys lano 21538.
wys lano 16118.
wys lano 2082.
wys lano 22929.
wys lano 16541.
wys lano 4833.
wys lano 31115.
wys lano 4639.
wys lano 29658.
wys lano 22704.
wys lano 9930.
wys lano 13977.
wys lano 2306.
wys lano 31673.
wys lano 22386.
wys lano 5021.
wys lano 28745.
wys lano 26924.
wys lano 19072.
wys lano 6270.
wys lano 5829.
wys lano 26777.

Python Shell
Otrzymano dane: 5537.
Otrzymano dane: 4639.
Otrzymano dane: 29658.
Otrzymano dane: 22704.
Otrzymano dane: 9930.
Otrzymano dane: 13977.
Otrzymano dane: 2306.
Otrzymano dane: 31673.
Otrzymano dane: 22386.
Otrzymano dane: 5021.
Otrzymano dane: 28745.
Otrzymano dane: 26924.
Otrzymano dane: 19072.
Otrzymano dane: 6270.
Otrzymano dane: 5829.
Otrzymano dane: 26777.
Ln: 151 Col: 0

```

Fig. 2. The example of communication between server, written in Python and compiled C-client

as Windows or Linux. One of the common ways is to implement multithreading functionality, what means running several routines at the same time and exchanging data via, for example, a swap file. A common resource in this case may be in a computer's internal memory or on a hard disk drive. However, this type of cooperation between two programs makes it difficult to implement it correctly on the various operating systems. In addition, one of the project's assumptions was that computer programs are easily integratable.

Due to the above-mentioned issues, a technique of combining two programs using TCP/IP was used. One of the programs acts as a server and the second as client. Both server and client can swap data. It was decided that the server is a program written in *Python*. It controls the motor and enables connection with client. The client is written in *C++* and is responsible for processing data from the proximity sensor.

Experiments were performed using several types of acquisition cards in order to demonstrate the versatility of the solution. We have successfully used cards manufactured, inter alia, by Advantech (USB-4704, PCI-1710), as well as drivers based on

Experiments consisted of altering the value of stiffness and damping parameters when motor was at a steady-state ( $v = const$ ).

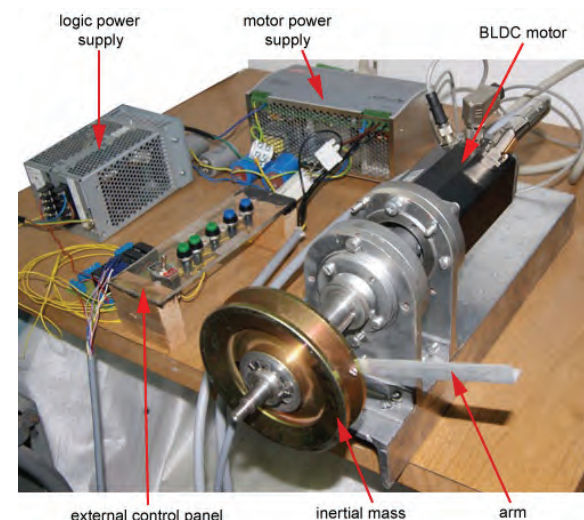
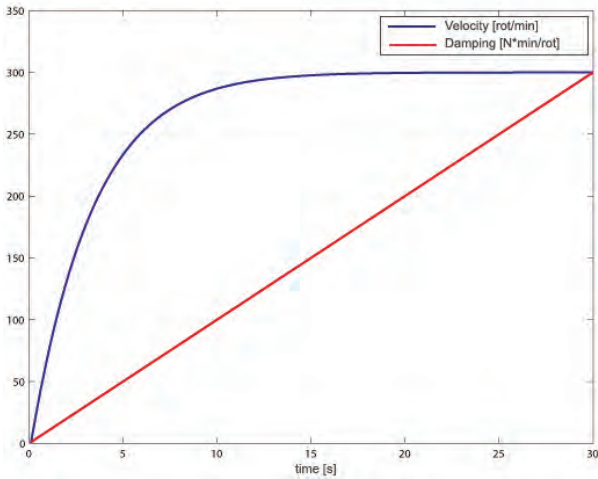
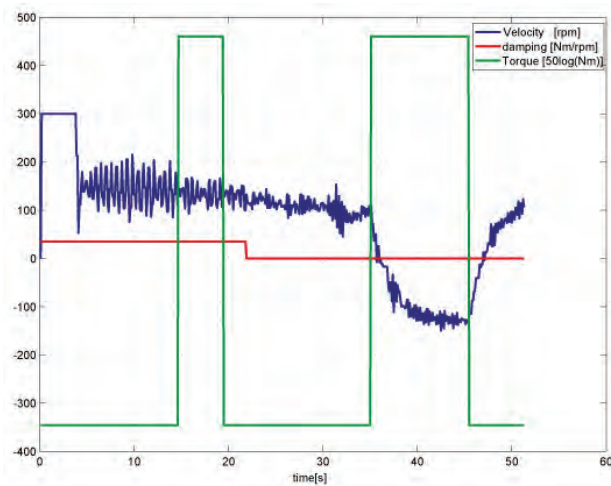


Fig. 3. The laboratory position with the BLDC motor





**Fig. 4. Graph of desired virtual speed of the manipulator when changing the damping coefficient. The virtual speed was 300 RPM. During the experiments the motor was rotating**

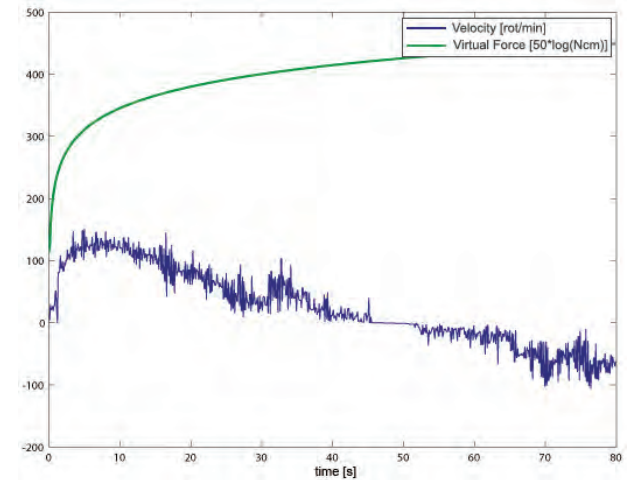


**Fig. 5. Velocity of the manipulator with discrete changes of virtual forces (the distance of object from the proximity sensor) and changes in controller's damping parameter. The virtual velocity was 300 RPM**

One of conducted experiments was to suddenly change the force which acts on the manipulator; meaning the distance between the object and the proximity sensor, while also suddenly changing attenuation values (Fig. 5). From the graph below it

can be concluded that a high damping value makes the system so rigid that it does not respond to the operating force. If the damping is small, the driver reacts causing even a change in the direction of rotation. In some situation it is the manipulator's most desirable response.

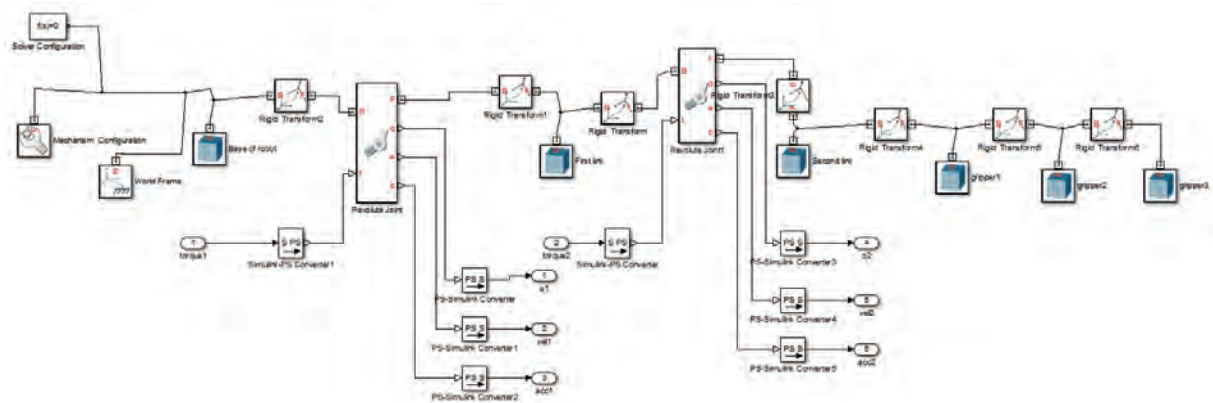
The second experiment was to increase the manipulator's impedance when the object is closer to the robot. The graph shows that when the value of the virtual force is in the range of 0.18 to 0.19 Nm (the time interval from 45 to 47.5 s), the motor stops. In some cases this is the most optimal response. After crossing this threshold, the motor begins to slowly rotate in the opposite direction (Fig. 6). If the object moved away from the motor, velocity would return to its original value.



**Fig. 6. Changes in velocity when approaching the object to the sensor when distance from the object to the sensor is less than 0.5 cm, the motor direction of rotation changes**

#### 4.2 Simulation Study of Cartesian Impedance Controller

Second experiment was conducted in Cartesian space in order to test what the properties of control system are, when impedance controller processes task variables. These parts of experiments were done on 2 DOF planar manipulator in Matlab-Simulink environment. Manipulator was modeled in second edition of Sim-Mechanics toolbox (Fig. 7).



**Fig. 7. Manipulator with 2 degrees of freedom modeled in Sim-Mechanics environment**

The dynamic model of the manipulator was simplified to mass points placed in the ends of rigid links. This can be done without limitation to the generality of the results achieved. The dynamics of a manipulator can be written as:

$$\begin{bmatrix} m_1 l_1^2 + m_2 (l_1^2 + 2l_1 l_2 \cos \theta_2 + l_2^2) & m_2 l_2 (l_1 \cos \theta_2 + l_2) \\ m_2 l_2 (l_1 \cos \theta_2 + l_2) & m_2 l_2^2 \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} + \begin{bmatrix} -m_2 l_1 l_2 (2\dot{\theta}_1 \dot{\theta}_2 + \dot{\theta}_2^2) \sin \theta_2 \\ m_2 l_1 l_2 \dot{\theta}_1^2 \sin \theta_2 \end{bmatrix} + \begin{bmatrix} (m_1 + m_2) g l_1 \cos \theta_1 + m_2 g l_2 \cos (\theta_1 + \theta_2) \\ m_2 g l_2 \cos (\theta_1 + \theta_2) \end{bmatrix} + \begin{bmatrix} f_{friction1} \\ f_{friction1} \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} \quad (7)$$

Forward kinematics of the manipulator is at the form:

$$\begin{aligned} x_c &= l_1 \cos \theta_1 + l_2 \cos (\theta_1 + \theta_2) \\ y_c &= l_1 \sin \theta_1 + l_2 \sin (\theta_1 + \theta_2) \end{aligned} \quad (8)$$

while inverse kinematics is given by:

$$\begin{aligned} \theta_2 &= \pm \arccos \left( \frac{x_c^2 + y_c^2 - l_1^2 - l_2^2}{2l_1 l_2} \right) \\ \theta_1 &= \pm \arccos \left( \frac{x_c^2 + y_c^2 + l_1^2 - l_2^2}{2l_1 \sqrt{x_c^2 + y_c^2}} \right) + \arctg \left( \frac{y_c}{x_c} \right) \end{aligned} \quad (9)$$

From (8) Jacobian matrix of the manipulator can be obtained:

$$J(\theta) = \begin{bmatrix} \frac{\partial x_c}{\partial \theta_1} & \frac{\partial x_c}{\partial \theta_2} \\ \frac{\partial y_c}{\partial \theta_1} & \frac{\partial y_c}{\partial \theta_2} \end{bmatrix} =$$

$$\begin{bmatrix} -l_1 \sin \theta_1 - l_2 \sin (\theta_1 + \theta_2) & -l_2 \sin (\theta_1 + \theta_2) \\ l_1 \cos \theta_1 + l_2 \cos (\theta_1 + \theta_2) & l_2 \cos (\theta_1 + \theta_2) \end{bmatrix} \quad (10)$$

and finally an inverse Jacobian is of the form:

$$J^{-1}(\theta) = \frac{1}{l_1 l_2 \sin \theta_2} \begin{bmatrix} l_2 \cos (\theta_1 + \theta_2) & l_2 \sin (\theta_1 + \theta_2) \\ -l_1 \cos \theta_1 - l_2 \cos (\theta_1 + \theta_2) & -l_1 \sin \theta_1 - l_2 \sin (\theta_1 + \theta_2) \end{bmatrix} \quad (11)$$

The other subsystems of control were modeled in standard Simulink environment. The view of the whole system is presented in Figs. 8 and 9.

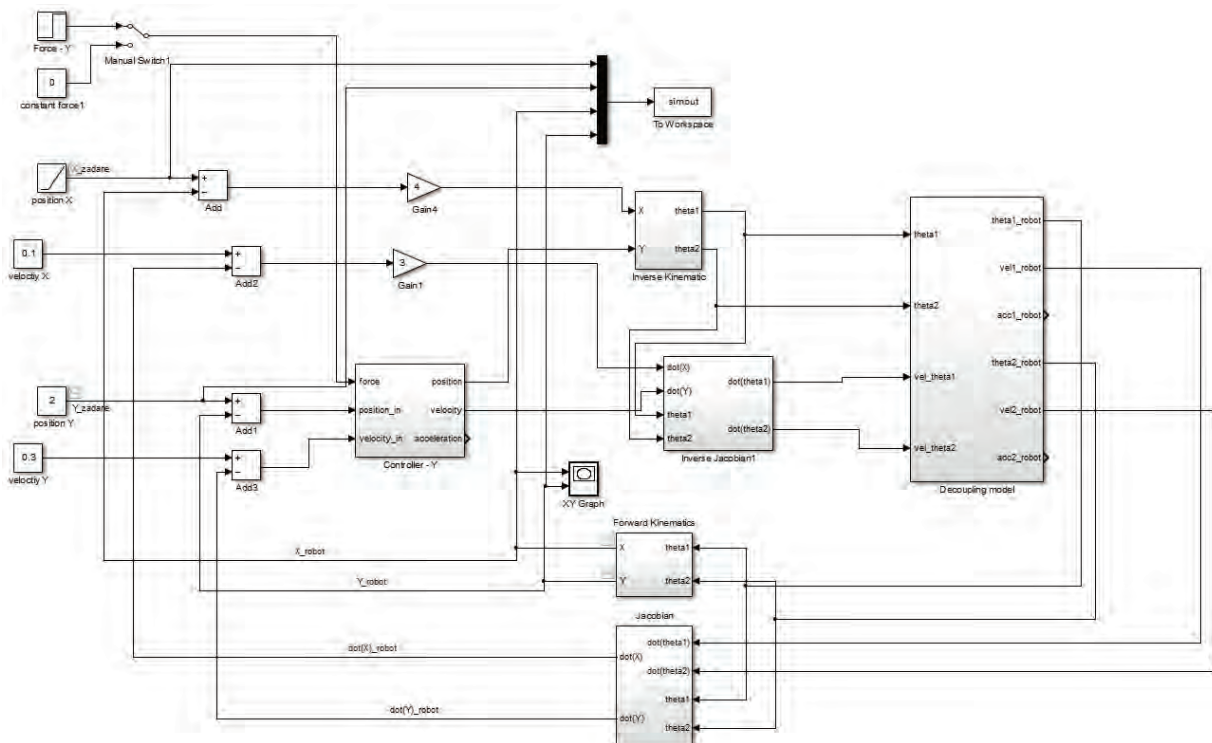
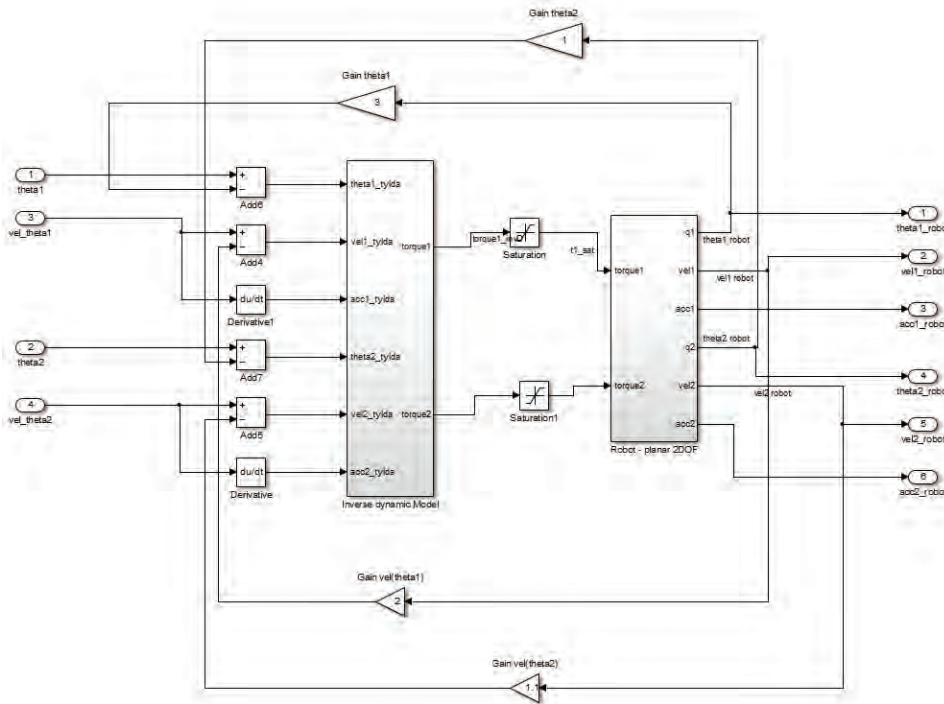
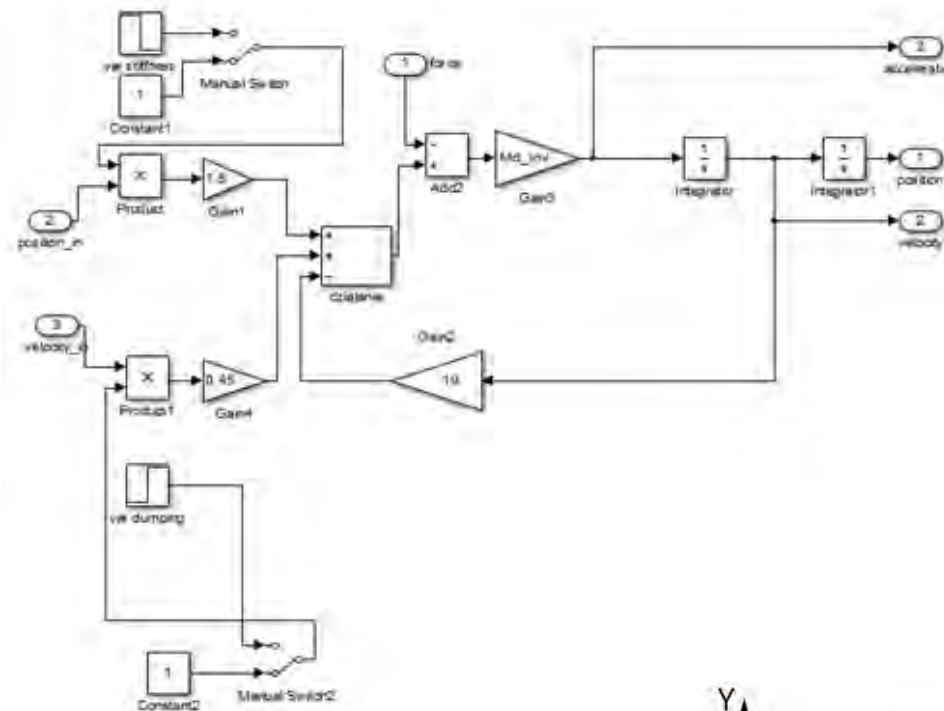


Fig. 8. The overall scheme of Simulink simulation



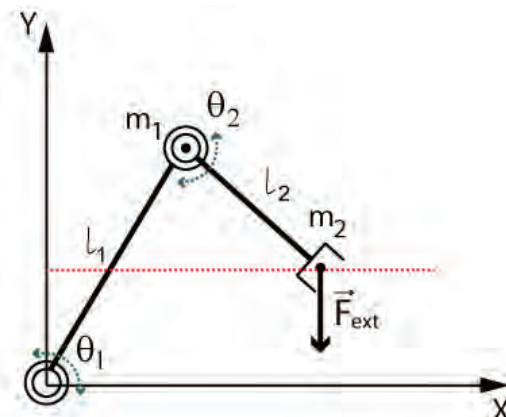
**Fig. 9. The view of decoupling model subsystem (model of manipulator control by the designated torque method). Model was decoupled with PD regulator**



**Fig. 10. Simulink scheme of admittance controller**

It can be seen from Fig. 8, that the impedance controller was applied in Y-coordinate in manipulator control system, while in the X-axis a standard PD regulator was used.

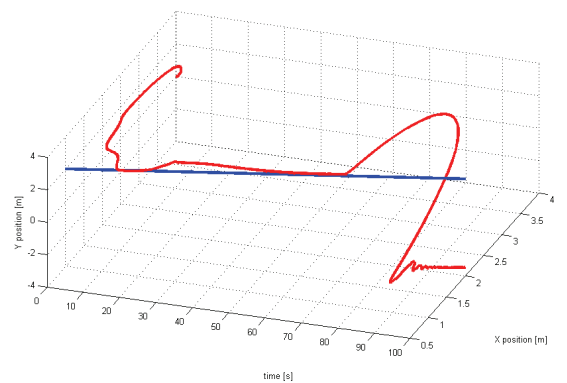
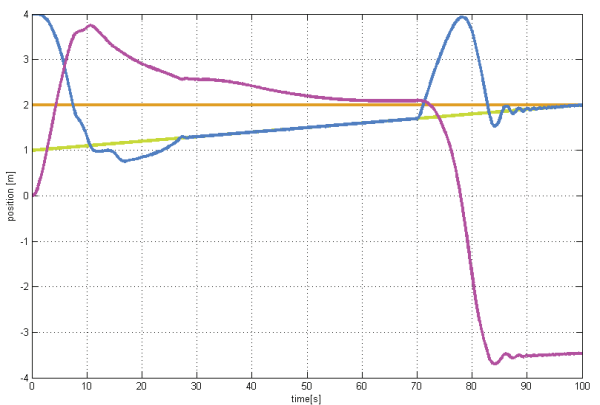
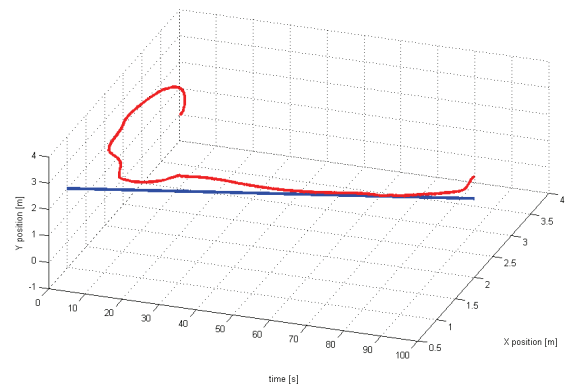
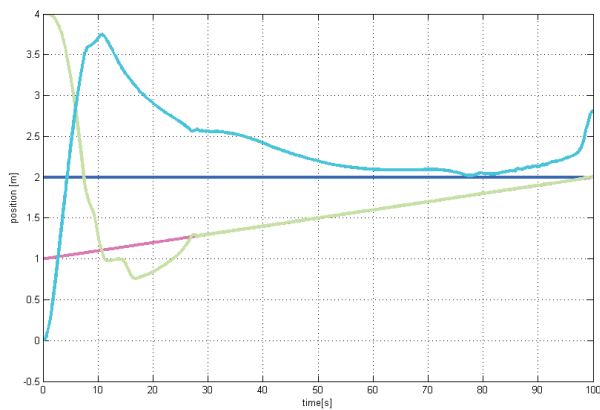
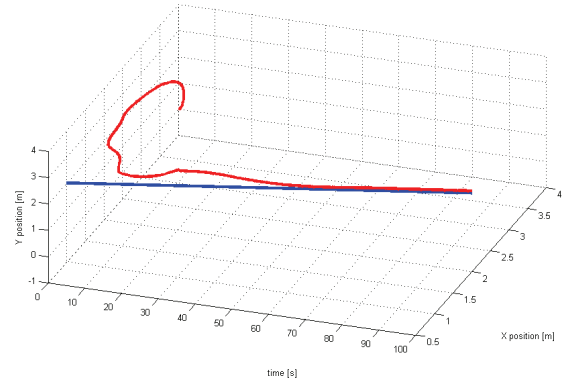
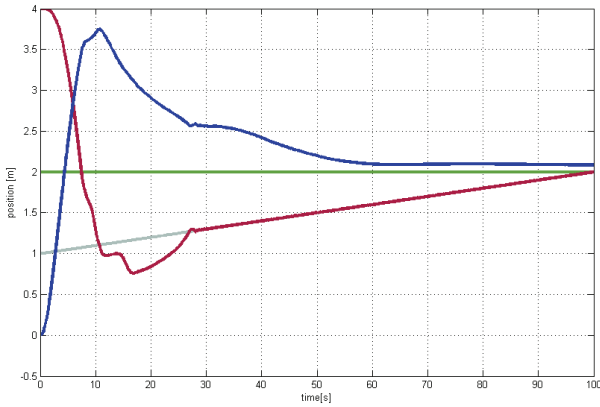
Experiments were designed to see how the manipulator behaves with different impedance controller settings. Simulation consisted of controlling the 2DOF robot on the desired trajectory. During 70 seconds of simulation high external force was applied in Y-axis. This situation can be likened to a collision with an obstacle (Fig. 11).



**Fig. 11. Schematic representation of 2DOF manipulator and force acting in Y axis**

The results of research were presented on below graphs. Three runs show the trajectory tracking in case of rigid manipulator, when impedance was lower and very low impedance of the manipulator.

In the last case, as it was presented in previous experiments, the manipulator reacts by subjecting the force. Disturbances that occurred at the end of the workspace are associated with non-stationary of boundary conditions.



**Fig. 12.** The illustration of changing the impedance of the manipulator. The figure from the left presents the decreased value of impedance (left figure – very rigid manipulator, middle figure – impedance of controller slightly reduced, right figure – very low value of impedance)

## 5. Future Research

During experimentation it was discovered that choosing the most adequate parameters for the impedance controller is a complicated process. A set of parameters is closely related to the numerical methods used to solve differential equations, with a limited speed of calculation, as well as the selection of the scaling factors for information obtained from the position sensors. This is probably one of the main reasons why impedance/admittance controllers are rarely applied in practice.

Upcoming research will therefore be associated with the creation of an adaptive model, probably based on the methods of artificial intelligence, by means of which it will be possible to effectively select appropriate controller settings, depending on the desired behavior of the robot. The next scheduled project will also involve the integration of an impedance controller to control robots with higher degrees of freedom.

### Acknowledgements

This work is financially supported by the Ministry of Science and Higher Education of Poland (Grant No. N N514 469339).

**Note:** This is an expanded version of the paper presented during The Polish National Conference on Robotics, 2012.

### AUTHORS

**Edward Jezierski\*, Artur Gmerek**

Institute of Automatic Control,  
Lodz University of Technology,  
email: edward.jezierski@p.lodz.pl  
tel. 0-42 631-25-40

\*Corresponding author

### References

- [1] Caccavale F., Siciliano B., "Six-DOF impedance control based on angle/axis representations", *IEEE Transactions on Robotics and Automation*, vol. 15, no. 2, 1999, 289–300.
- [2] Granosik G., Jezierski E., Kaczmarek M., "Modeling, simulation and control of pneumatic jumping robot". In: *Proc. European Robotics Symposium 2008. Springer Tracts in Advanced Robotics*, Berlin: Springer 2008, 155–164.
- [3] Hogan N., "Impedance control: An approach to manipulation", Parts I, II, III, *ASME Journal of Dynamic Systems, Measurements, and Control*, 1985, vol. 107, 1–23.
- [4] Jezierski E., "Od sterowania pozycyjnego robotów do sterowania impedancyjnego". In: Tchoń K. (ed.) *Postępy robotyki – sterowanie, percepcja, komunikacja*, Warszawa: WKiŁ, 2006, 13–36. (In Polish)
- [5] Jezierski E., "On electrical analogues of mechanical systems and their using in analysis of

robot dynamics". In: Kozłowski K.R. (ed.), *Robot Motion and Control – Recent Developments*, Berlin: Springer, Berlin 2006, 391–404.

- [6] Jezierski E., *Dynamika robotów*, WNT, Warszawa, 2006. (In Polish)
- [7] Ott C., Albu-Schäffer A., Kugi A., Hirzinger G., "On the passivity-based impedance control of flexible joint robots", *IEEE Transactions on Robotics*, vol. 24, no. 2, 2008, 416–429.
- [8] Siciliano B., Sciavicco L., Villani L., Oriolo G., *Robotics – modelling, planning and control*, Springer, 2009.
- [9] Twuji T., Akamatsu H., Hatagi M. and Kaneko M., "Vision-Based Impedance Control for Robot Manipulators". In: *IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, 1997.
- [10] Winiarski T., Zieliński C., "Podstawy sterowania siłowego w robotach", *PAR*, no. 6, 2008, 5–10. (in Polish)
- [11] Jezierski E., Gmerek A., "Admittance control of a 1-DoF robotic arm actuated by BLDC motor". In: *17<sup>th</sup> International Conference on Methods and Models in Automation and Robotics (MMAR)*, 2012, 633–638.