

Real-Time Implementation of Data-Driven Predictive Controller for an Artificial Muscle

Osman ULKIR¹*, Gazi AKGUN², Erkan KAPLANOGLU²

¹ Department of Mechatronics Engineering, Mus Alparslan University, Mus, 49200, Turkey
o.ulkir@alparslan.edu.tr (*Corresponding author)

² Department of Mechatronics Engineering, Marmara University, Istanbul, 34730, Turkey
gaziakgun@marmara.edu.tr, ekaplanoglu@marmara.edu.tr

Abstract: This study presents a position tracking control method, with reference to Data-Driven Predictive Controller (DDPC), for a Pneumatic Artificial Muscle (PAM) system. The design of predictive controller is created from the subspace identification matrices acquired by input/output data. The control scheme is entirely data-based without explicit use of a model in the control application that can rectify the nonlinearity and uncertainties of the PAM. Firstly, subspace matrices are developed employing the identification method as a predictor by using open-loop experiments. Secondly, the estimated subspace matrices are used to design the so-called DDPC. In this instance, the quadratic programming (QR) decomposition method is used to obtain the prediction matrices. Consequently, experiments are carried out by the PAM actuator with different testing and loading conditions. The real-time experimental results demonstrate the feasibility and efficiency of the suggested control approach for nonlinear systems.

Keywords: Pneumatic artificial muscle, Data-driven predictive controller, Subspace identification, Nonlinear system.

1. Introduction

The pneumatic artificial muscle known as the ‘pneumatic muscle actuator’ or ‘fluidic muscle’ is a tube-like actuator which operates as a contractile or extensional device for actuating length when the pressurized air fills a pneumatic bladder (Andrikopoulos et al., 2014). PAM has several advantages in comparison with the traditional electric motors and pneumatic or hydraulic actuators, such as its lightness, large force, high power/weight ratio, and high power/volume ratio. Structurally, PAM resembles a human muscle in power, size, and weight. This is an advantage for humanoid robots, soft robotics, prosthetic and orthotic devices, as well as rehabilitation or training exoskeleton systems (Liu et al., 2015; Ai et al., 2018) in which heavy actuators can significantly increase the payload.

PAM consists of a cylindrical, flexible rubber tube wrapped inside a membrane netting and suitable metal (aluminum or steel), with fittings attached to each extremity. The membrane material is made of the grid and mesh-shaped fibres with non-elastic guiding (Lu et al., 2016). When air pressure is applied to the pneumatic muscle, it shortens in the axial direction and expands in the angular direction. With this structure, PAM generates a large contraction force in these directions and both the contraction and generated forces rely on the strength of the applied pressure.

The maximum contraction is approximately 25% of the muscle length.

PAM’s nonlinear nature, hysteresis and unknown terms make it difficult to model its behavior or to design a controller for a high-performance position tracking system. These terms are produced by the air complexity, the friction inside the braided mesh and the nonlinearity of its geometric structure, leading to the modeling and control challenges outlined above. In addition, a survey of PAM-related applications has described the importance of researching an optimal solution for position control.

In recent year, many approaches have proposed various mathematical models to describe the dynamic and static structures of PAM (Saga & Saikawa, 2008). In these models, PAM’s characteristics are identified as stemming from a physical analysis of pressure force relationships, its geometric structure and flexible nature (Ba et al., 2016). However, these studies have shown that PAM doesn’t easily lend itself to a precise description by way of these methods. Hence, various experimental approaches have been conducted to obtain more accurate PAM dynamics (Pujana-Arrese et al., 2010). Although the dynamics predicted from empirical data have been successfully applied to the position control, the common applicability of these methods is

limited. Nevertheless, both mathematical methods and experimental approaches include time-varying parameters and uncertainties that need to be rectified by the controller.

Due to the high nonlinearity inside PAM, it is not easy to accurately implement the position control. As such, a number of control algorithms have been proposed to deal with its uncertainties and these can be divided into two categories: Model-Based Control (MBC) theory and the Data-Driven Control (DDC) method. When applying the model-based control theory, the first step is to model the plant and then to do the designing of the controller based on the plant model. Some of the MBC algorithms include adaptive position control (Zhu et al., 2008), sliding mode control (Cao, Xie & Das, 2018), adaptive backstepping control (Chang, 2010), switching model control (Jiang et al., 2015), nonlinear optimal predictive control (Todorov et al., 2010) and active model-based control (Bleicher et al., 2011). For the DDC method, the controller is designed directly using online or offline input/output data of the controlled system without employing the mathematical model of the controlled plant. Examples of DDC algorithms are PID control (Fan et al., 2015), neural network nonlinear control (Chiang & Chen, 2017), fuzzy control (Jiang et al., 2015), model-free adaptive control (Ahmed, Wang & Yang, 2018) and data-driven predictive control. Although MBC ensures a higher positioning precision for PAM applications compared to DDC, its design process requires that the system be modelled and that there be an adequate knowledge of control theory, making it difficult for engineers who are unacquainted with the modeling and controller design to implement and make the controller impracticable for real-time systems.

The novelty of the study is to explore a DDPC method for position tracking control of a PAM system by applying a DDPC algorithm for the first time. This algorithm is the result of a combination of subspace identification and model predictive control. Unlike the model-based method, a novel approach was applied using the results from the subspace system identification, being well-suited to multivariable systems for

developing state-space models directly from input-output data (Favoreel, De Moor & Van Overschee, 2000). Studies of DDPC are generally simulation-based (Xia et al., 2013; Sun et al., 2017); however, in this study, real-time control is performed using DDPC. Recent studies of DDPC for industrial and academic applications can be found for instance in (Smarra et al., 2018; Hou, Liu & Tian, 2017).

The aim of the present study is to develop a controller in order to achieve high-sensitive position tracking control with PAM using a DDPC algorithm. When estimated subspace matrices are directly employed in a predictive control design, this is referred to as a 'DDPC'. In this case, the QR decomposition is required for obtaining the prediction matrix constructed from the input-output measurements. Neither a model order assumption nor a prediction needed to be carried out owing to the fact that the state-space matrices are not retrieved here. Real-time experimental studies are applied to observe and analyze the characteristics of the control system using the DDPC algorithm under several testing and loading conditions. The designed method is able to confirm good position tracking performance with a fast response and a high robustness for the PAM system.

The rest of the paper is organized as follows: Section 2 describes the experimental setup; Section 3 presents the designed DDPC algorithm; Section 4 shows the experimental results; Section 5 concludes the study.

2. Experimental Setup

A schematic diagram of 1 axis pneumatic artificial muscle manipulator is shown in Figure 1. The hardware includes a personal computer to send the control signal for controlling a 5/3 way proportional valve (FESTO, MPYE-5-1/8-HF-010 B) through a microcontroller (ATmega 2560) and to drive the circuit (L298N), converting the digital signal to an analog voltage, and air is injected from the pressure supply (6 bar). A linear position sensor (Novotechnik,

conductive plastic potentiometer) is used to measure the position of the PAM (FESTO, MAS-20-N-500-AA-MCGK).

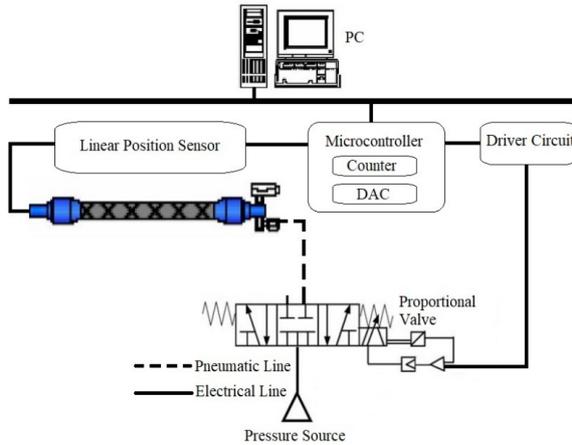


Figure 1. Schematic diagram of 1 axis PAM

In this study, the first joint of the PAM is fixed, and the control algorithm is applied to control the muscle position. PAM generates pulling forces to pull and push the linear position sensor along the moving direction within a working range of 0 to 50 mm. The outside inertia load is tested with three different loading conditions (2, 3 and 4 kg) using a double acting pneumatic cylinder. The experiments are conducted under a pressure of 6 bar and the proposed controller is implemented on a PC using Matlab/Simulink. The technical details of the device used in the experimental studies are listed in Table 1 and a photograph of the experimental setup is in Figure 2.

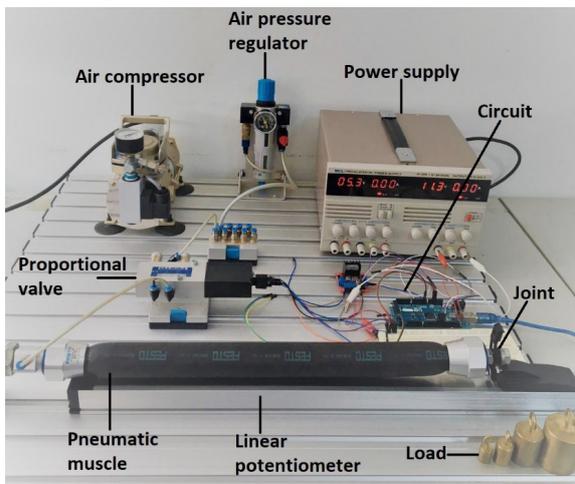


Figure 2. PAM experimental setup

Table 1. Technical data of experimental equipment.

Equipment	Description
PAM	Type: (FESTO, MAS-20-N-500-AA-MCGK) Nominal length: 500 mm Operation pressure: 0 ... 6 bar Force compensation: 1500 N
Proportional Valve	Type: FESTO, MPYE-5-1/8-HF-010 B Maximum pressure: 10 bar Set point value: 0 ... 10 V DC
Microcontroller	Type: ATmega 2560 AI and AO: 10 bit resolution
Linear Potentiometer	Type: Novotechnik TLH 0900 Nominal resistance: 10 K Ω

3. Control System

Due to the compressibility of the air source and the nonlinear nature of the PAM force, a DDPC is developed accordingly and applied into the position tracking system based on the subspace identification method. The key features of the developed controller are as follows:

1. The controller theory is designed using input/output data with on-line and off-line mechanisms and the cost function of model predictive control (MPC).
2. The data is taken as knowledge without mathematical models or implicit and explicit information.

The parameters of the controlled system varied depending on the patient or the healing process, making the system difficult to model, especially when using pneumatic muscle actuators, i.e., rehabilitation robots and orthotic or prosthetic applications. Consequently, data-driven algorithms are preferred over model-based algorithms and the control model undergoes a difficult and expensive process. Furthermore, it is not efficient for a system that will be applied to multi-user processes with different mechanical properties. The DDPC algorithm weight parameters can be calculated with the obtained subspace matrices and the system is able to track the reference with a control rule based on previous input/output data and tracking errors within the defined horizon. The rule based on past

data is able to respond quickly by following all the actions along the horizon.

An MPC scheme is adopted whereby at each time step an optimal control problem is solved through quadratic programming aiming at minimizing the deviations of the system variables from their equilibrium points. Because the cost function of the DDPC solves this problem, there is no need to linearized it around the equilibrium points.

3.1 Subspace Identification Method

This subsection provides the background on the subspace identification matrices from the open loop data. These matrices will be used in the following section to design a data-driven predictive controller. Firstly, a state-space illustration of the system will be defined; hence, a Linear Discrete Time-Invariant (LTI) system can be written in a state-space form as following equations:

$$x_{k+1} = Ax_k + Bu_k + Ke_k \quad (1)$$

$$y_k = Cx_k + Du_k + e_k \quad (2)$$

Here $u_k \in \mathbb{R}^m$, $y_k \in \mathbb{R}^l$ and $x_k \in \mathbb{R}^n$ are the input variables, the output variables and the state vector variables of the system, respectively, and $e_k \in \mathbb{R}^l$ is the white noise disturbance. This case is confined to a fictitious scenario in which the noise does not affect the measurements or the system. The system matrices $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{l \times n}$, $D \in \mathbb{R}^{l \times m}$ and $K \in \mathbb{R}^{l \times l}$ are the state, input, output, feed-through and Kalman filter gain matrices, respectively.

The measurements of the inputs, u_k , and the outputs, y_k ($k \in \{1, 2, \dots, N\}$), are assumed to be accessible for system identification and the input Hankel matrices for u_k , represented as U_p and U_f , are defined as:

$$U_p \triangleq \begin{bmatrix} u_1 & u_2 & \cdots & u_{N-2M+1} \\ u_2 & u_3 & \cdots & u_{N-2M+1} \\ \cdots & \cdots & \cdots & \cdots \\ u_M & u_{M+1} & \cdots & u_{N-2M+1} \end{bmatrix} \quad (3)$$

$$U_f \triangleq \begin{bmatrix} u_{M+1} & u_{M+2} & \cdots & u_{N-M+1} \\ u_{M+2} & u_{M+3} & \cdots & u_{N-M+2} \\ \cdots & \cdots & \cdots & \cdots \\ u_{2M} & u_{2M+1} & \cdots & u_N \end{bmatrix} \quad (4)$$

where the indexes ‘ f ’ and ‘ p ’ represent the ‘future’ and ‘past’ matrices of the variables. Similarly, Hankel matrices for y_k , represented as Y_f and Y_p . The dimensions of the matrices are $\{Y_p, Y_f\} \in \mathbb{R}^{M \times N-2M+1}$, $\{U_p, U_f\} \in \mathbb{R}^{M \times N-2M+1}$.

$$Y_p \triangleq \begin{bmatrix} y_1 & y_2 & \cdots & y_{N-2M+1} \\ y_2 & y_3 & \cdots & y_{N-2M+2} \\ \cdots & \cdots & \cdots & \cdots \\ y_1 & y_1 & \cdots & y_{N-M+1} \end{bmatrix} \quad (5)$$

$$Y_f \triangleq \begin{bmatrix} y_{M+1} & y_{M+2} & \cdots & y_{N-M+1} \\ y_{M+2} & y_{M+3} & \cdots & y_{N-M+2} \\ \cdots & \cdots & \cdots & \cdots \\ y_{2M} & y_{2M+1} & \cdots & y_N \end{bmatrix} \quad (6)$$

In the subspace identification method, these Hankel matrices are made to be rectangular. Thus, the undesired effects of noise on the identification system are minimized. This condition can be achieved by having a large data set, indicated by the variable N . Moreover, M in equations (3)-(6) represents the order of the predictor equation. For successful identification, the order M must be bigger than or at least equal to the real system order n as demonstrated in the size of the state matrix A (Van Overche & De Moor, 2012). The system state matrices are written as:

$$X_p \triangleq [x_1 \quad x_2 \quad \cdots \quad x_{N-2M+1}] \quad (7)$$

$$X_f \triangleq [x_{M+1} \quad x_{M+2} \quad \cdots \quad x_{N-M+1}] \quad (8)$$

Consequently, the derivation of equations (1) and (2) can be written as below. These equations are known as the subspace matrix input-output equations (Kadali, Huang & Rossiter, 2003) used in subspace identification.

$$Y_f = \Gamma_M X_f + H_M^d U_f + H_N^d E_f \quad (9)$$

$$Y_p = \Gamma_M X_p + H_M^d U_p + H_N^s E_p \quad (10)$$

$$X_f = A^M X_p + \Delta_M^d U_p + \Delta_M^s E_p \quad (11)$$

$\Gamma_M \in \mathbb{R}^{Ml \times n}$ can be described as the extended observability matrix, $\Delta_M^d \in \mathbb{R}^{n \times Mm}$ as reversed prolonged deterministic controllability matrix, and $\Delta_M^s \in \mathbb{R}^{n \times Ml}$ as the reversed prolonged stochastic controllability matrix (Mardi, 2010). Y_f can be written as below:

$$Y_f = \begin{bmatrix} \Gamma_M A^M \Gamma_M^T - A^M (H_M + \Delta_M) \\ U_p \end{bmatrix} \begin{bmatrix} Y_p \\ U_p \end{bmatrix} \quad (12)$$

$$+ H_M U_f + (\Delta_M^s - A^M \Gamma_M^T H_M^s) E_p + H_M^s E_f$$

Since the effect of E_f is the white noise constant and is the reason for the constancy of a Kalman filter, equation (12) can be written to obtain an optimal prediction expression of the system output \hat{Y}_f as follows:

$$\hat{Y}_f = L_w W_p + L_u U_f \quad (13)$$

where $W_p = \begin{bmatrix} Y_p & U_p \end{bmatrix}^T$, U_f consists of past inputs, outputs, and future inputs, respectively. $L_w \in [0; 10] \mathbb{R}^{Ml \times M(l+m)}$ is the subspace matrix corresponding to past input-output states; and $L_u \in \mathbb{R}^{Ml \times Mm}$ is the subspace matrix corresponding to future inputs. In equation (13), future outputs can be approached as a linear combination of past input-output states and the future input of the system. Equation (13) is employed to define the characterization of the system, by restoring the identification methods giving the state-space definition of the system.

The following least squares problem is solved by calculating N and L_u using Hankel matrices.

$$\min \left\| Y_f - \begin{bmatrix} L_w & L_u \end{bmatrix} \begin{bmatrix} W_p \\ U_f \end{bmatrix} \right\|_F^2 \quad (14)$$

This problem can be solved from shifting the orthogonal projection of the row space Y_f into the row space of the matrices $W_p = \begin{bmatrix} Y_p & U_p \end{bmatrix}^T$. This can be defined by equation (15) as follows:

$$\hat{Y}_f = Y_f / \begin{bmatrix} W_p \\ U_f \end{bmatrix} \quad (15)$$

3.2 Data-Driven Predictive Controller

The DDPC consists of the subspace-based predictor defined in subsection 3.1 and the cost function of the model predictive control algorithm. The structure of the proposed control method is shown in Figure 3.

In Figure 3, M and N are the length of data and $y_d(k+1)$, $y(k+1)$, $u(k-M)$ are the desired output, output, and input, respectively. All I/O data is stored in a database for reuse in the control method. This data is generated offline using Matlab and applied to PAM through serial communication between the computer and microcontroller. The collected I/O data are used for subspace identification in Matlab. The values and calculated in Matlab are sent to the microcontroller via serial communication. The control algorithm is then executed on microcontroller with those values and actual I/O data through control horizon.

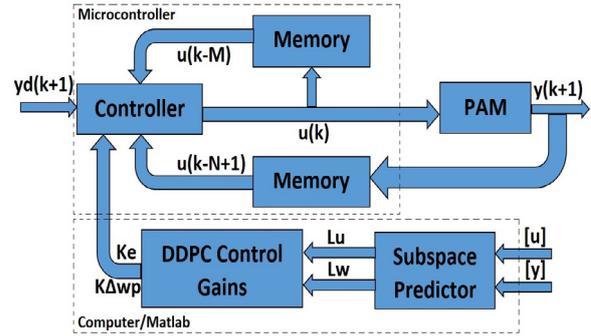


Figure 3. Structure of the DDPC in a block diagram

The MPC algorithm cost function form (Venkat et al., 2008) can be written with the prediction and control horizon N_p and N_c equal to f as follows:

$$J = \sum_{k=1}^N (\hat{Y}_{t+k} - r_t)^T W_Q (\hat{Y}_{t+k} - r_{t+k}) \quad (16)$$

$$+ \sum_{k=1}^N \Delta U_{t+k}^T W_R \Delta U_{t+k}$$

where W_Q and W_R are the weight matrices, r_t is the reference signal at present t , N_p and N_c are the prediction and control horizon, respectively, with N_c being lower than or equal to the prediction horizon N_p ($N_c \leq N_p$) or $N_c \leq f$.

The improvement of the basics of the DDPC via rewriting the cost function of the MPC equation (16) in quadratic form is maintained.

Using equation (13) and the reference signal r_t , the cost function can be updated as follows:

$$J = \left(L_w \Delta W_p + L_u^{N_c} \Delta U_{N_c} + Y_t - r_{t+1} \right)^T W_Q \times \left(L_w \Delta W_p + L_u^{N_c} \Delta U_{N_c} + Y_t - r_{t+1} \right) + \Delta U_{N_c}^T W_R \Delta U_{N_c} \quad (17)$$

If the cost function is solved, the control rule can be written as follows:

$$\Delta U_f = - \left(\left(L_u^{N_c} \right)^T W_Q \left(L_u^{N_c} \right) + W_R \right)^{-1} \times \left(L_u^{N_c} \right)^T W_Q \left(L_w \Delta W_p + Y_t - r_{t+1} \right) = -K_{\Delta W_p, N_c} \Delta W_p - K_{e, N_c} (Y_t - r_{t+1}) \quad (18)$$

where $-K_{\Delta W_p, N_c}$ and K_{e, N_c} are the weight of the past data and the tracking error, respectively.

At each time condition, the first element ΔU_{N_c} is used to compute the control input U_{t+1} , which complies with ΔU_{t+1} . Hence, abbreviating the first m rows in equation (18) leads to:

$$\Delta U_{t+1} = -K_{\Delta W_p} \Delta W_p - K_e (Y_t - r_{t+1}) \quad (19)$$

with,

$$K_{\Delta W_p} = \begin{bmatrix} I_m & 0_{m \times (M-1)m} \end{bmatrix} K_{\Delta W_p, N_c} \quad (20)$$

$$K_e = \begin{bmatrix} I_m & 0_{m \times (M-1)m} \end{bmatrix} K_{e, N_c} \quad (21)$$

Consequently, the control input U_t can be written as follows:

$$U_t = U_{t-1} + \Delta U_t \quad (22)$$

4. Experimental Results

The data-driven predictive control method was validated on a real-time system by different experimental studies. The experimental setup was explained in Section 2. The system position was directly measured using a linear potentiometer and the control algorithm was applied using Matlab/Simulink on a PC with a sampling time of 10 ms. This sampling time was chosen to

achieve an optimal control performance and to be convenient for the bandwidth of the used proportional valve.

The working ranges of the PAM system were detected as follows:

$$y \in [0; 50] \text{ mm } P \in [0; 6] \text{ bar, } u \in [0; 10] \text{ V}$$

where y , P and u are the position and pressure of PAM and the driving voltage range of the proportional valve, respectively.

To design the DDPC, the input/output data collected from the open-loop experiment is necessary to define the subspace matrices. This experiment is implemented with a variable frequency sinusoidal signal of magnitude 0.7 for the inputs.

The subspace predictor matrices, $L_w (100 \times 100)$, $L_u (100 \times 100)$ are determined by using subspace identification, $i = 100$ (row blocks) and $j = 100$ (column blocks) in the Hankel matrices. The sequence of the Hankel matrices was set as $N_c = 15$, $f = 100$, $p = 50$, while the weighting matrices are selected as $W_Q = I_{fl}$, $W_R = I_{fm}$.

Depending on the prediction horizon, the subspace prediction scheme was implemented at approximately 0.91 seconds. If the prediction horizon was selected any higher, it would be over 1 second.

The designed input signal, output signal, control signal to drive the proportional valve, and the control gains $K_{\Delta W_p}$ and $K_e = 0.0518$ are shown in Figure 4. With the identification data, the control gains can be calculated using equation (19). The significant coefficients from calculating the gains are N_c, f, p, W_Q, W_R . The optimum parameters are calculated as above and real-time control of the PAM is realized by applying the control gains to the DDPC control rule determined in equation (22).

The DDPC was applied to the PAM for position control, based on the theory given in subsection 3.2. For position tracking, multistep and sine signals at various frequencies (0.5, 1, 1.5 and 0.2 Hz) and with/without loading (2, 3, and 4 kg) were separately utilized.

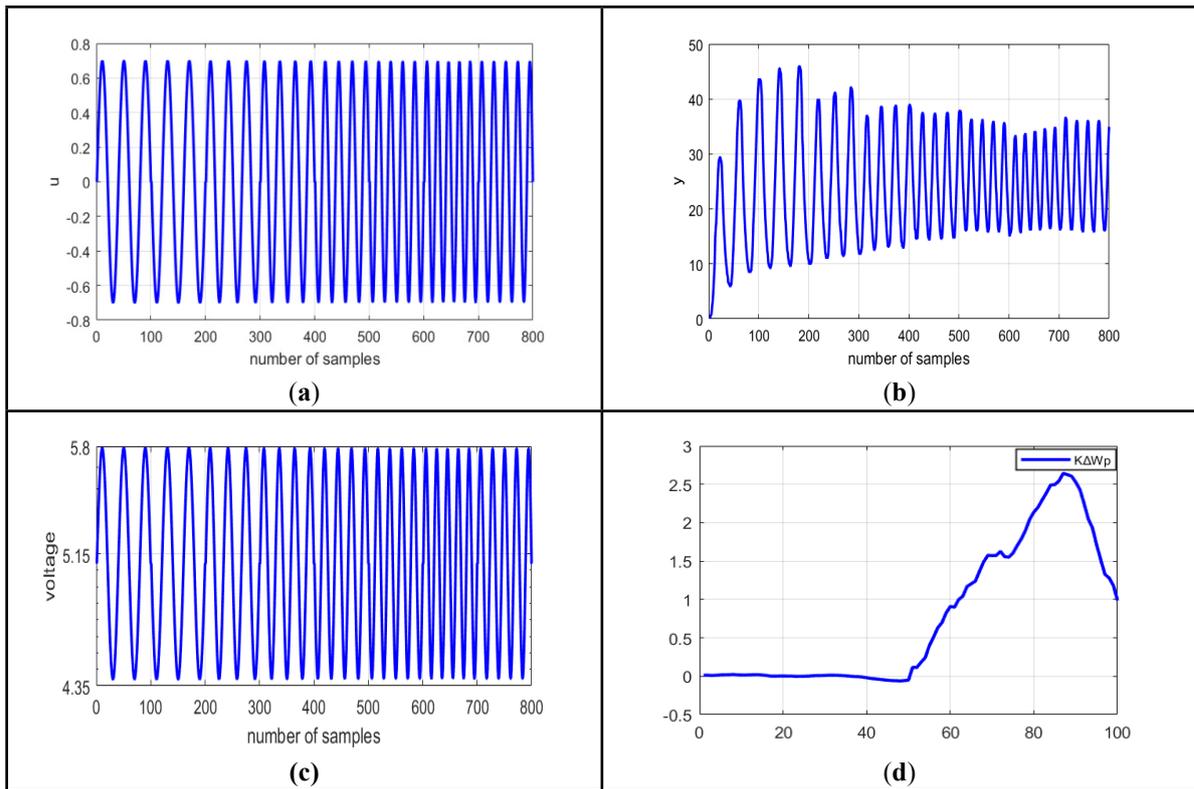


Figure 4. Identification data for PAM and control gains. (a) Sine input signal for identification. (b) Measured output signal (c) Control input voltage (d) Control gains: $K_{\Delta W_p}$ and $K_e = 0.0518$

Experimental study 1: The experiments were implemented on the PAM to track the multistep trajectory within the range of [5; 15] mm in a free-load situation. The results are shown in Figure 5. As shown in the figure, the designed controller drove the system accurately at different set points. As the settling time is not good enough, the error between the change of reference orbit is high. The average steady-state error in the multistep trajectory is 0.142 mm and the maximum error is 1.012 mm. As a result, the suggested control algorithm generated a good transient response and steady-state performance.

Experimental study 2: The experiments were implemented on the PAM to track a sine signal trajectory with an amplitude of 30 mm and at various frequencies of 0.5, 1 and 1.5 Hz under free-load conditions. The experimental results are given in Figure 6. The maximum errors of sine signal tracking are 1.85 mm, 2.04 mm, and 2.31 mm at 0.5 Hz, 1 Hz, and 1.5 Hz, respectively. Compared to 0.5 Hz, the DDPC control error increased from 6.2% at 1 Hz to 12.8% at 1.5 Hz. Because of its robustness, the behavior of the

controller was sustained within improved ranges of 0.5 Hz \rightarrow ± 0.48 mm (4.8%), 1 Hz \rightarrow ± 0.88 mm (8.8%), 1.5 Hz \rightarrow ± 1.38 mm (13%) of the control errors. Consequently, the controller is deemed to have good robustness and exhibit favorable hysteresis performance.

Experimental study 3: The experiments were implemented to track a sine signal trajectory with a frequency of 0.2 Hz and an amplitude of 20 mm under four conditions: free-load and a further three loads of 2, 3 and 4 kg. The controller performance and error results acquired under each of the load conditions are shown in Figure 7. The maximum control errors of the sine signal tracking are 0.95 mm, 1.14 mm, 1.19 mm, and 1.21 mm under the free-load, 2, 3 and 4 kg load conditions, respectively. The best position tracking performance was achieved using a DDPC algorithm characterized by robustness and adaptability. As a result, the proposed DDPC algorithm is able to consistently restrict the control errors under the free- and 4-kg load conditions to within an acceptable range of 0.32 mm (3.2%) and 0.72 mm (7.2%), respectively.

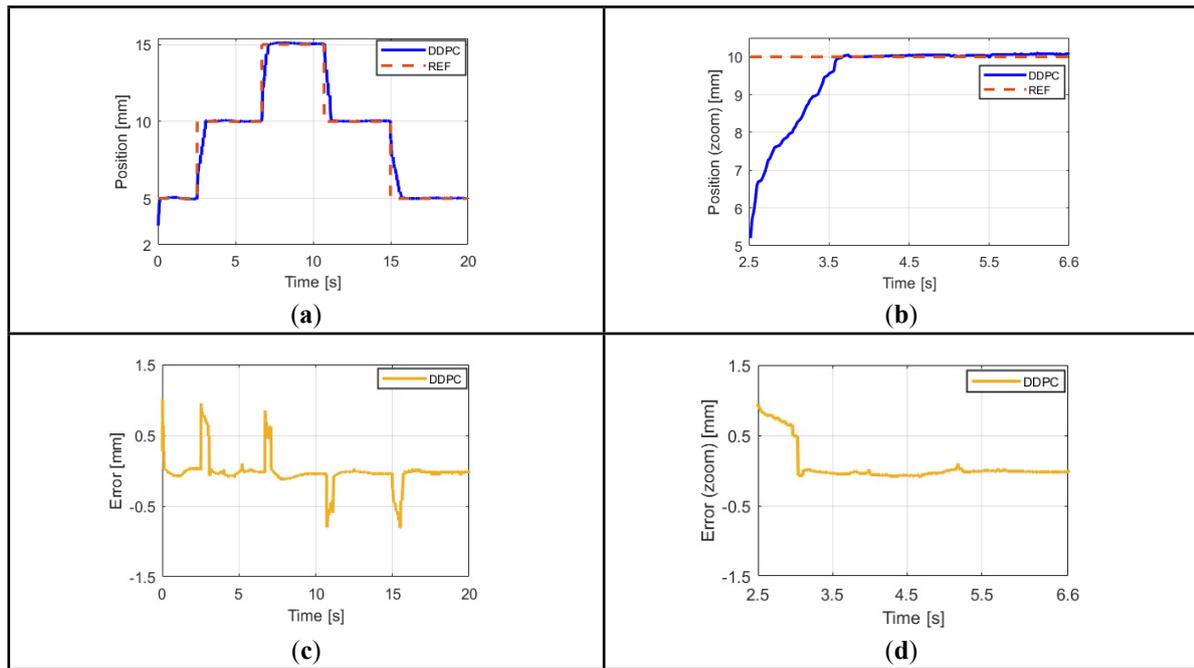


Figure 5. Data from experimental study 1 (a) Multistep tracking results (b) Zoomed-in tracking results from 2.5 s to 6.6 s (c) Tracking errors in the multistep test (d) Zoomed-in tracking errors from 2.5 s to 6.6 s

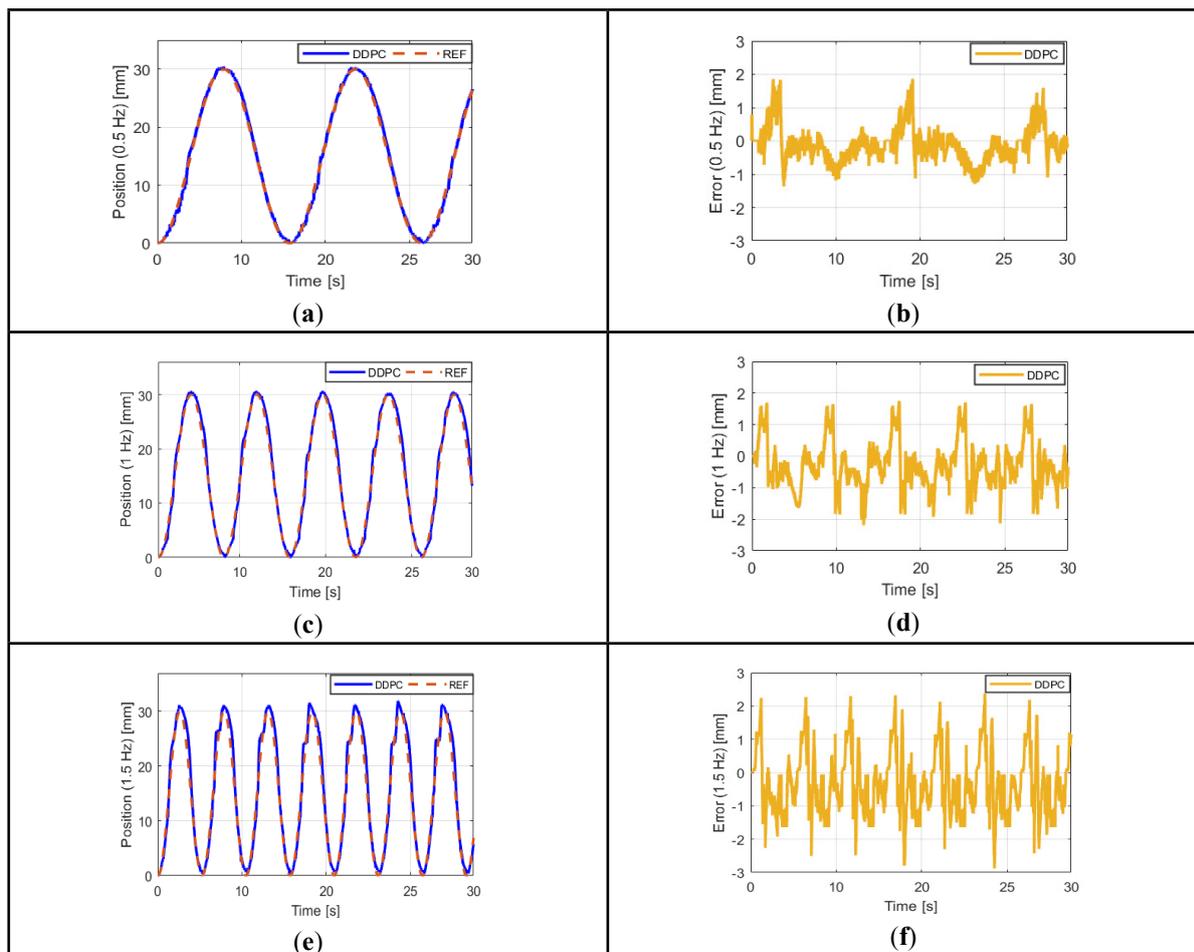


Figure 6. Data from experimental study 2 (a) Sine tracking results at 0.5 Hz (b) Sine tracking errors at 0.5 Hz (c) Sine tracking results at 1 Hz (d) Sine tracking errors at 1 Hz (e) Sine tracking results at 1.5 Hz (f) Sine tracking errors at 1.5 Hz

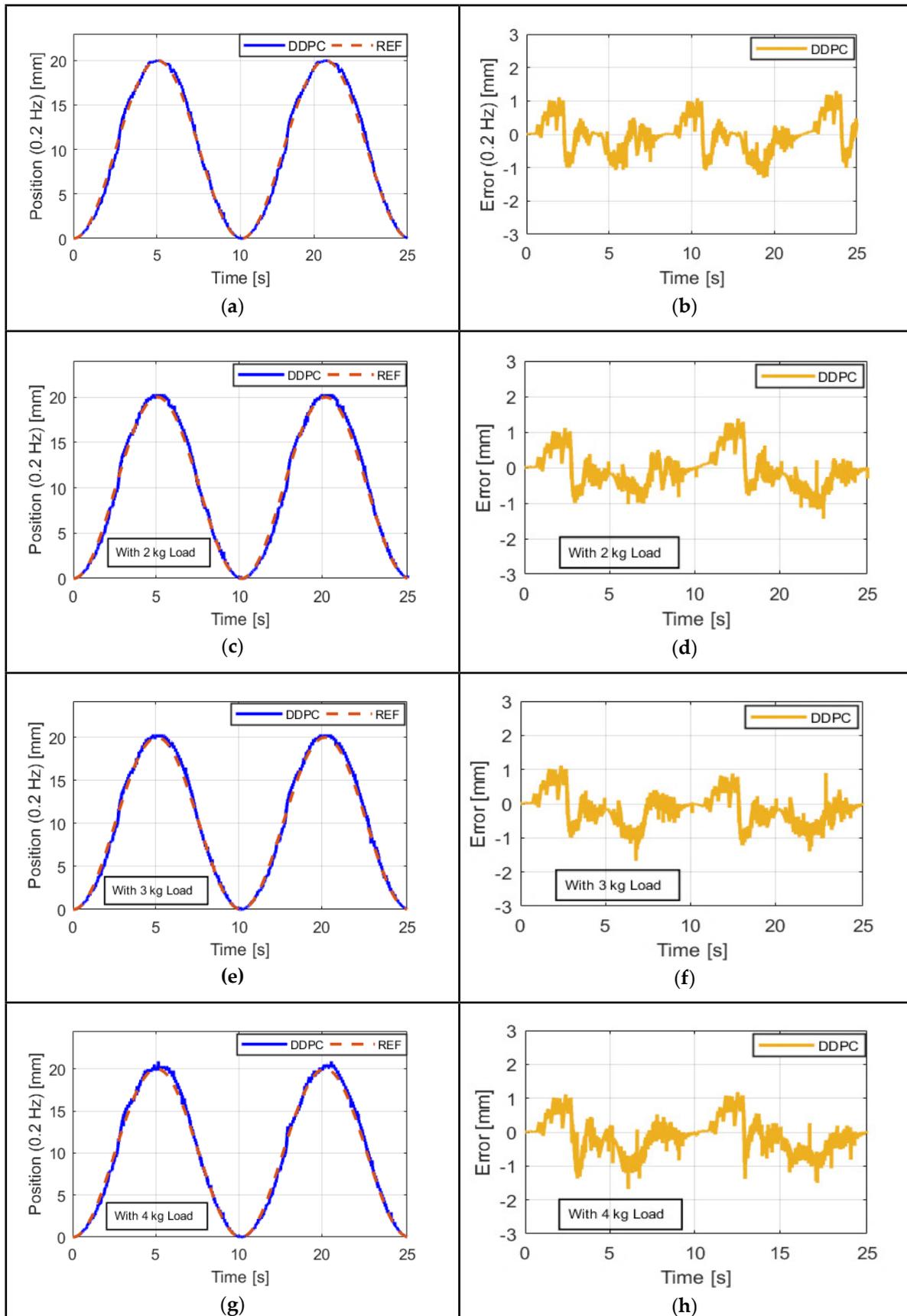


Figure 7. Data from experimental study 3 (a) Sine tracking results at 0.2 Hz (b) Tracking errors at 0.2 Hz (c) Sine tracking results at 0.2 Hz with 2 kg (d) Tracking errors at 0.2 Hz with 2 kg (e) Sine tracking results at 0.2 Hz with 3 kg (f) Tracking errors at 0.2 Hz with 3 kg (g) Sine tracking results at 0.2 Hz with 4 kg (h) Tracking errors at 0.2 Hz with 4 kg.

Table 2. Performances of all the experimental studies

		RMS Error
Experiment	Trajectory	DDPC
Experiment 1 Free-Load	Multistep signal	0.278
Experiment 2 Free-Load	Sin 30 mm 0.5 Hz 1 Hz 1.5 Hz	0.485 0.848 1.34
Experiment 3 Free-Load	Sin 20 mm 0.2 Hz	0.357
Experiment 3 Load 2 kg	Sin 20 mm 0.2 Hz	0.408
Experiment 3 Load 3 kg	Sin 20 mm 0.2 Hz	0.425
Experiment 3 Load 4 kg	Sin 20 mm 0.2 Hz	0.463

Compared to the model-based controller described in the literature using a similar PAM (Chang, 2010), the proposed controller demonstrated significant positioning accuracy and fast response with a low overshoot and high level of robustness.

The root mean square (RMS) errors of the data-driven predictive controller in all the case studies are given in Table 2. These results show the robustness and effectiveness of the DDPC.

5. Conclusion

In order to solve the problem of modelling and position tracking control of a nonlinear PAM system, a data-driven predictive control strategy was proposed using subspace identification and model predictive control theory. Subspace matrices were developed using the identification method as a predictor and adopting open-loop experiments. The estimated subspace matrices were employed for a predictive control design referred to as 'DDPC' and applied to a PAM for

the first time within this field of research. Due to its data-driven nature, DDPC was unstable and could be easily adapted to other types of systems without knowledge of the mathematical models of the plant.

An experimental setup was developed to observe the position tracking performance of the PAM system. Subsequently, the real-time experimental results relating to the different reference signals and load conditions were approved having good dynamic properties, robustness and effectiveness, indicating feasibility of the proposed control method for real-time applications, including for linear and nonlinear systems.

In future studies, optimization of the DDPC algorithm will be realized to solve the rising and setting time problems in the PAM system. Furthermore, the DDPC will be compared to PID and sliding mode controllers in order to evaluate its performance. A forthcoming work aims to achieve an accurate position control of a pneumatic ankle-foot orthosis device based on the proposed DDPC algorithm.

REFERENCES

1. Ahmed, S., Wang, H. & Tian, Y. (2018). Model-free control using time delay estimation and fractional-order nonsingular fast terminal sliding mode for the uncertain lower-limb exoskeleton, *Journal of Vibration and Control*, 24(22), 5273-5290.
2. Ai, Q., Zhu, C., Zuo, J., Liu, Q., Xie, S. & Yang, M. (2018). Disturbance-estimated adaptive backstepping sliding mode control of a pneumatic muscles-driven ankle rehabilitation robot, *Sensors*, 18(1), 66.

3. Andrikopoulos, G., Nikolakopoulos, G., Arvanitakis, I. & Manesis, S. (2014). Piecewise affine modelling and constrained optimal control for a pneumatic artificial muscle, *IEEE Transactions on Industrial Electronics*, 61(2), 904-916.
4. Ba, D. X., Dinh, T. Q. & Ahn, K. K. (2016). An integrated intelligent nonlinear control method for a pneumatic artificial muscle, *IEEE/ASME Transactions on Mechatronics*, 21(4), 1835-1845.
5. Bleicher, A., Schlaich, M., Fujino, Y. & Schauer, T. (2011). Model-based design and experimental validation of active vibration control for a stress ribbon bridge using pneumatic muscle actuators, *Engineering Structures*, 33(8), 2237-2247.
6. Cao, J., Xie, S. Q. & Das, R. (2018). MIMO sliding mode controller for gait exoskeleton driven by pneumatic muscles, *IEEE Transactions on Control Systems Technology*, 26(1), 274-281.
7. Chang, M. K. (2010). An adaptive self-organizing fuzzy sliding mode controller for a 2-DOF rehabilitation robot actuated by pneumatic muscle actuators, *Control Engineering Practice*, 18(1), 13-22.
8. Chiang, C. J. & Chen, Y. C. (2017). Neural network fuzzy sliding mode control of pneumatic muscle actuators, *Engineering Applications of Artificial Intelligence*, 65, 68-86.
9. Fan, J., Zhong, J., Zhao, J. & Zhu, Y. (2015). BP neural network tuned PID controller for position tracking of a pneumatic artificial muscle, *Technology and Health Care*, 23(2), 231-238.
10. Favoreel, W., De Moor, B. & Van Overschee, P. (2000). Subspace state space system identification for industrial processes, *Journal of Process Control*, 10(2-3), 149-155.
11. Hou, Z., Liu, S. & Tian, T. (2017). Lazy-learning-based data-driven model-free adaptive predictive control for a class of discrete-time nonlinear systems, *IEEE transactions on neural networks and learning systems*, 28(8), 1914-1928.
12. Jiang, X., Wang, Z., Zhang, C. & Yang, L. (2015). Fuzzy neural network control of the rehabilitation robotic arm driven by pneumatic muscles, *Industrial Robot: An International Journal*, 42(1), 36-43.
13. Kadali, R., Huang, B. & Rossiter, A. (2003). A data-driven subspace approach to predictive controller design, *Control Engineering Practice*, 11(3), 261-278.
14. Liu, Y., Zang, X., Liu, X. & Wang, L. (2015). Design of a biped robot actuated by pneumatic artificial muscles, *Bio-Medical Materials and Engineering*, 26(1), 311-317.
15. Lu, T., Shi, Z., Shi, Q. & Wang, J. (2016). Bioinspired bicipital muscle with fiber-constrained dielectric elastomer actuator, *Extreme Mechanics Letters*, 6, 75-81.
16. Mardi, N. (2010). Data-driven subspace-based model predictive control (Thesis).
17. Pujana-Arrese, A., Mendizabal, A., Arenas, J., Prestamero, R. & Landaluze, J. (2010). Modelling in Modelica and position control of a 1-DoF set-up powered by pneumatic muscles, *Mechatronics*, 20(5), 535-552.
18. Saga, N. & Saikaw, T. (2008). Development of a pneumatic artificial muscle based on biomechanical characteristics, *Advanced Robotics*, 22(6-7), 761-770.
19. Smarra, F., Jain, A., de Rubeis, T., Ambrosini, D., D'Innocenzo, A. & Mangharam, R. (2018). Data-driven model predictive control using random forests for building energy optimization and climate control, *Applied energy*, 226, 1252-1272.
20. Sun, J., Xing, G. & Zhang, C. (2017). Data-Driven Predictive Torque Control During Mode Transition Process of Hybrid Electric Vehicles, *Energies*, 10(4), 441.
21. Todorov, E., Hu, C., Simpkins, A. & Movellan, J. (2010). Identification and

- control of a pneumatic robot. *In 2010 3rd IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics* (pp. 373-380).
22. Van Overschee, P. & De Moor, B. L. (2012). *Subspace identification for linear systems*. Springer Science Media.
23. Venkat, A. N., Hiskens, I. A., Rawlings, J. B. & Wright, S. J. (2008). Distributed MPC strategies with application to power system automatic generation control, *IEEE Transactions on Control Systems Technology*, 16(6), 1192-1206.
24. Xia, Y., Xie, W., Liu, B. & Wang, X. (2013). Data-driven predictive control for networked control systems, *Information Sciences*, 235, 45-54.
25. Zhu, X., Tao, G., Yao, B. & Cao, J. (2008). Adaptive robust posture control of a parallel manipulator driven by pneumatic muscles, *Automatica*, 44(9), 2248-2257.