Online identification using Markov coefficients: application to a DC motor

Simona-Daiana Stiole
Department of Automation
Technical University of Cluj-Napoca
Cluj-Napoca, Romania
simona.sim@aut.utcluj.ro

Zsófia Lendek
Department of Automation
Technical University of Cluj-Napoca
Cluj-Napoca, Romania
zsofia.lendek@aut.utcluj.ro

Vlad-Mihai Mihaly
Department of Automation
Technical University of Cluj-Napoca
Cluj-Napoca, Romania
vlad.mihaly@aut.utcluj.ro

Mircea Susca
Department of Automation
Technical University of Cluj-Napoca
Cluj-Napoca, Romania
mircea.susca@aut.utcluj.ro

Attila Medgyesi
Department of Automation
Technical University of Cluj-Napoca
Cluj-Napoca, Romania
medgyesi.is.attila@student.utcluj.ro

Anca Elena Pica
Department of Automation
Technical University of Cluj-Napoca
Cluj-Napoca, Romania
anca.pica@campus.utcluj.ro

Doina Pisla
CESTER
Technical University of Cluj-Napoca
Cluj-Napoca, Romania
doina.pisla@mep.utcluj.ro

Calin Liviu Vaida
CESTER
Technical University of Cluj-Napoca
Cluj-Napoca, Romania
calin.vaida@mep.utcluj.ro

Petru Dobra
Department of Automation
Technical University of Cluj-Napoca
Cluj-Napoca, Romania
petru.dobra@aut.utcluj.ro

***Abstract*—This paper presents the online identification, based on finite impulse response filter coefficients, of a DC motor. The coefficients are obtained based on least mean-square identification using experimental data. They are used to construct the Hankel matrix based on which the mathematical model is determined. The results are subsequently compared with a standard method in Matlab. The method is applied in real time directly on a low-cost development board, where it successfully replicates the identification process previously tested in simulation. The results obtained online confirm the accuracy and reliability of the approach in a real-time setting. This work bridges the gap between theoretical system identification and practical real-time implementation, enabling motor identification with accessible hardware. The proposed approach supports rapid prototyping, educational use, and cost-effective industrial applications, particularly in scenarios that require real-time system monitoring.**

Keywords—identification, adaptive filters, Markov coefficients

I. INTRODUCTION

System identification [1] builds mathematical models of dynamic systems using measurements of the input and output of the system. The steps usually taken in order to identify a mathematical model are: 1) input-output data acquisition, 2) selecting or estimating a model structure, 3) estimating the model parameters and 4) validation of the identified model.

System identification is essential for subsequent model-based analysis and synthesis of dynamic systems. Various identification methods exist in the literature that can be parametric [1], non-parametric [2], neural network identification etc. The identification of linear time invariant (LTI) systems is important both in signal processing fields such as seismic deconvolution [3], channel equalization [4] (in communications), radar [5], and systems and control theory [6], [7]. Many parameter estimation methods for linear systems, such as multiple innovation parameter identification methods [8], [9], iterative estimation methods [1], data filtering-based estimation methods and gradient-based estimation methods [8], assume that the system order or structure is known. For instance, in [10], a third order cumulants-based algorithm has been proposed for the identification of a linear system with minimum and nonminimum phase, excited by non-Gaussian, independent identically distributed sequences.

Digital filters present a significant interest, as they have a multitude of applications, see [11]. One application is the representation of the response of a process to impulse by finite impulse response (FIR) filter coefficients. This can be considered as an identification method [12] and has received considerable attention [8]. FIR filters can be implemented in continuous time or in discrete time. In digital signal processing a FIR filter is a filter [13] whose impulse response has a well-defined period and settles to zero in a finite time. This is frequently contrasted with IIR (Infinite Impulse Response) filters, which have the potential for internal feedback and respond indefinitely. An Nth order discrete time FIR filter's impulse response takes precisely N+1 samples to reach zero. [14] shows that it is possible to construct filter matrices that preserve the impulse response of arbitrary linear time-invariant (LTI) systems. [2] focuses on the Kronecker product decomposition of impulse responses, together with low-rank approximations. It proposes two (iterative/recursive) algorithms: an iterative Wiener filter, with improved performance compared to the conventional Wiener filter, and a recursive adaptive least-squares algorithm.

Although deriving a mathematical model based on impulse response has been well-documented in the literature [1], its real-time application has not been sufficiently explored. In this work, we will provide a concrete example

This research was supported by the project New frontiers in adaptive modular robotics for patient-centered medical rehabilitation– ASKLEPIOS, funded by European Union – NextGenerationEU and Romanian Government, under National Recovery and Resilience Plan for Romania, contract no. 760071/23.05.2023, code CF 121/15.11.2022, with Romanian Ministry of Research, Innovation and Digitalization, within Component 9, investment I8.

using a DC motor with unknown parameters, subjected to real-time testing.

Adaptive subband [12] filtering is of considerable interest because it reduces computational complexity and improves the convergence speed in adaptive signal processing applications. An adaptive filter has an associated adaptive algorithm for updating the filter coefficients to allow the filter to operate in unknown and changing environments. Adaptive system identification has been widely applied in active noise control [15], digital control [16], communications [17] etc. Online system identification, or real-time identification, is also an important application of adaptive algorithms, allowing models to dynamically adjust to changing conditions. In this work we will utilize algorithms based on FIR filters, which offer advantages in stability and computational simplicity for real-time implementation. Using these adaptive filters, both identification and control can be achieved in real time for multiple-input multiple-output (MIMO) systems, as shown in [18]; however, this requires high-processing-power computers.

In this paper we use FIR filters for the identification of a DC motor. The choice of a DC motor is motivated by the simplicity of its mathematical model, the ease of access to its internal states, and its availability for testing in laboratory environment. However, the identification algorithm can be applied to any system or process where the input can be controlled and the output can be measured. It is important to note that the process should not be too fast, ensuring that it can be processed in real-time by the low-cost development board with limited specifications. This broadens the algorithm's applicability while maintaining feasibility within constrained hardware environments. For estimating the Markov coefficients, which are essential for constructing the Hankel matrix, we selected least mean-square (LMS) based on processing cost and error performance as shown in our previous work in [11]. Once these Markov coefficients - also known as FIR coefficients - are obtained, we construct the Hankel matrix and subsequently derive the motor's mathematical model, specifically its transfer function.

The identification is performed online and validated experimentally. A comparative analysis with existing methods is conducted. The novelty of this work lies in the real-time implementation of the identification process. By using only a low-cost development board, we demonstrate the feasibility of deriving a mathematical model for a system with unknown parameters.

The main contribution of this paper is bridging the gap between theoretical system identification and practical real-time implementation, using only low-cost, accessible hardware. We show that accurate model identification for a DC motor can be achieved in real time and under significant resource constraints, directly enabling cost-effective scenarios where immediate system characterization is needed. While our study is demonstrated on a SISO system, the approach is readily applicable to other contexts requiring real-time system monitoring.

This paper is structured as follows. Section II presents aspects related to the motor, the development board and the acquisition of input-output data. The identification of the coefficients of the FIR filter and the steps to obtain the mathematical model are presented in Section III. A comparison between the performed identification and a classic method is described in Section IV. Section V presents some conclusions and starting points for further developments.

II. SETUP

The case study we consider is the identification of the direct current motor in Fig.1. For data acquisition the Nucleo-64P development board, see Fig. 2, is used.

A. The motor

The motor whose dynamics is identified is the direct current motor in Fig. 1. The parameters are not known.

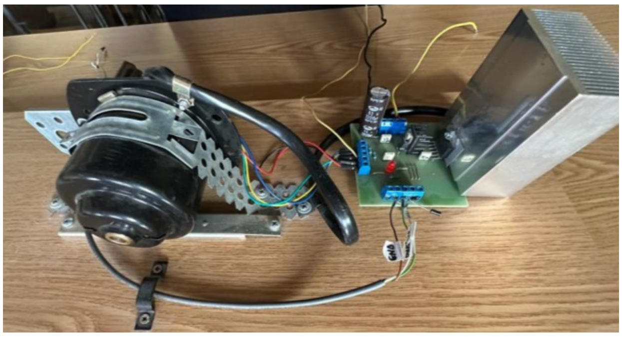

Fig. 1. The motor used for identification

The motor's dynamics are [6]:

$$\begin{pmatrix} \frac{di_a(t)}{dt} \\ \frac{d\omega_m(t)}{dt} \\ \frac{d\theta_m(t)}{dt} \end{pmatrix} = \begin{pmatrix} \frac{-R_a}{L_a} & \frac{-K_e}{L_a} & 0 \\ \frac{K_t}{J_m} & \frac{-D_m}{J_m} & 0 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} i_a \\ \omega_m \\ \theta_m \end{pmatrix} + \begin{pmatrix} \frac{1}{L_a} \\ 0 \\ 0 \end{pmatrix} u_a(t) \quad (1)$$

where $i_a(t)$ represents the rotor's current, $\omega_m(t)$ is the angular velocity, $\theta_m(t)$ is the angular position, R_a represents the rotor's resistance, L_a is the rotor's inductance, K_e is a proportionality constant, K_t is torque constant, D_m is the viscous damping, J_m represents the equivalent moment of inertia and $u_a(t)$ is the voltage applied to the armature.

Model (1) is in state-space form, where each variable has a clear physical meaning. In our case, the measured output is the angular velocity ω_m of the motor, thus $y = \omega_m$. Although our goal is to identify the motor described by (1), the identified model will not retain this exact form, as it will be obtained in discrete time and represents an approximation of the original system.

B. Data acquisition

The STM32 Nucleo 64-P development board (see Fig.2.) was used to generate the pseudo-random binary sequence (PRBS) input and to acquire data from the motor.

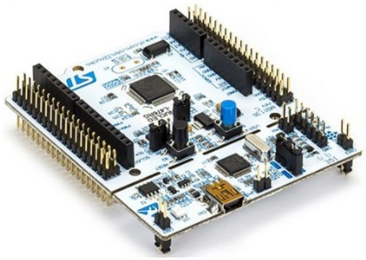

Fig. 2. The development board used in the data acquisition process

The setup is presented in Fig. 3. The input is applied, and the output is acquired using the STM32 Nucleo-64P microcontroller. The sampling period is $T_s = 5 \cdot 10^{-3}$ $[sec]$. The sampling period was chosen to be as small as possible to avoid altering the captured signal, yet large enough to allow for the data to be transmitted and processed efficiently, i.e. to balance identification accuracy with the practical constraints of real-time implementation. A smaller sampling period would introduce a higher computational load, which could hinder performance, especially if additional processing (e.g., filtering, control) is required in parallel. In fact, 5 ms approaches the practical lower limit for our low-cost development board when considering concurrent processing tasks.

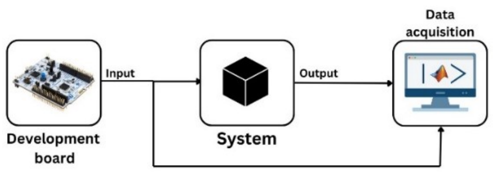

Fig. 3. Hardware setup for data acquisition

A PRBS signal is used as input for the motor in order to capture its response characteristics. This type of signal is commanly used in system identification [1], telecommunications [4], and signal processing to test and analyze system responses to complex inputs. It serves as a duty cycle for the motor and is easily generated by a microcontroller. The period of the PRBS is also 5 ms and it ensures that even for the shortest segment of the input signal, several data points could be obtained for both input and output signals. The measured output is the angular velocity of the motor, more precisely its frequency, measured by the encoder (1000 pulses/revolution). Both the PRBS input and the motor output are acquired in real time and can be seen in Fig. 4.

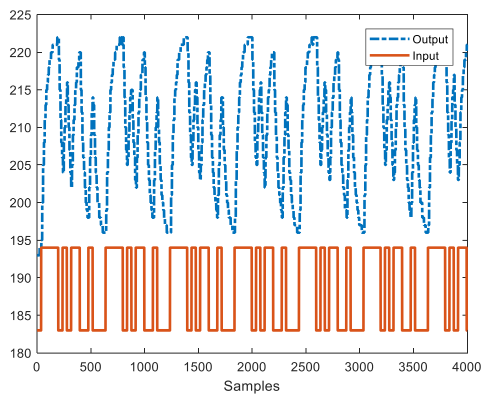

Fig. 4. Acquired signals

III. System Identification

In this section we discuss the steps taken to identify the motor. Adaptive algorithms will be used to obtain FIR coefficients, after which the Hankel matrix is formed, and then reduced.

A. Data preprocessing and Markov coefficients

Most signals of practical interest are analog. To process them using numerical processors, they must be sampled. Our goal is to accurately identify the process despite these aspects. Markov coefficients, known also as FIR coefficients, represent the impulse response of an LTI system and characterize the system's output at each discrete time step following an impulse input. To identify them we use subband adaptive filters that return an imposed number of coefficients.

An adaptive system identification (or modeling) structure is shown in Fig. 5, where the adaptive filter is placed in parallel with an unknown system (or plant) to be identified. The adaptive filter provides a linear model. The excitation signal $u(n)$ serves as input to both the unknown system and the adaptive filter, while $\eta(n)$ represents the disturbance (or plant noise) occurring in the unknown system. The objective is to model the unknown system so that the adaptive filter output $y(n)$ closely matches the unknown system output $d(n)$. This is achieved by minimizing the error signal $e(n)$, which is the difference between the physical response $d(n)$ and the model response $y(n)$.

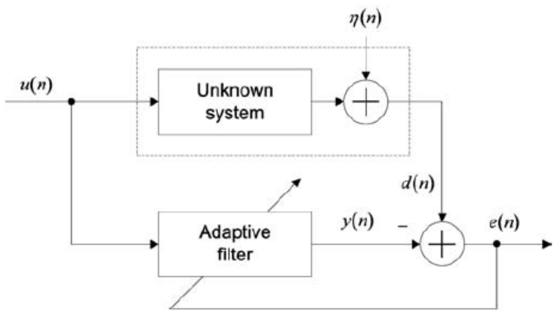

Fig. 5. Block diagram of adaptive system identification

Several algorithms for obtaining the coefficients have been tested using collected data from input and output, among which Least Means Square (LMS), Normalized LMS (NLMS), Affine Projection (AP) and Recursive Least Squares (RLS). A comparison of these methods was carried out in [19], where we evaluated their performance in estimating the Markov coefficients. Among them, the LMS algorithm provided the best trade-off between simplicity and accuracy, yielding the lowest estimation error. In this paper, we build upon those results and use the LMS-estimated coefficients for constructing the Hankel matrix, benefiting from the algorithm's computational efficiency and reliable performance in real-time scenarios.

B. Hankel matrix decomposition

A Hankel matrix [20] is a square matrix that is constant on any diagonal that is orthogonal to the main one. The Hankel matrix has been found useful [6], [21] for decomposing non-stationary signals and representing time-frequency.

Hankel matrices are formed when, given a sequence of input-output data, a realization of an underlying state space model is desired. Singular value decomposition of the Hankel matrix provides a means to compute the A, B, and C matrices of the linear state space realization.

After obtaining the Markov coefficients $g_i, , i = 0, \dots, M$, as described in Section III A, the Hankel matrix is constructed as:

$$H_p = \begin{pmatrix} g_1 & g_2 & \cdots & g_p \\ g_2 & g_3 & & g_{p+1} \\ \vdots & & \ddots & \vdots \\ g_p & g_{p+2} & \cdots & g_{2p} \end{pmatrix} \qquad (2)$$

where p represents the number of rows and columns of the composite matrix. For our experiments, we use $M = 33$, determined experimentally.

Next, we compute the singular values. To determine the order of the system that best reproduces its response, it is necessary to consider the order as the number of the highest singular values of the Hankel matrix. In this sense, according to Fig. 6 which presents these values, we may consider order 5 (the first 5 highest values can be observed), but, as model (1) is of order 3, we choose order 3.

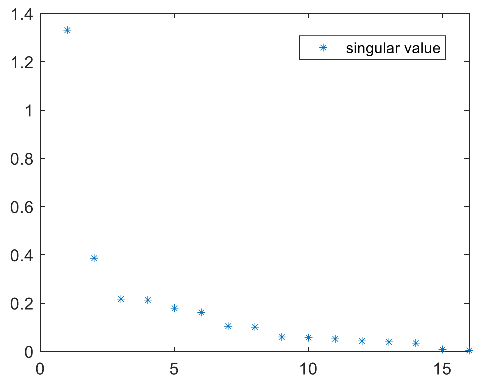

Fig. 6. The singular values obtained

Thus, we consider a reduction to the 3rd order. For this, H is decomposed as:

$$H = \begin{pmatrix} U_s & U_o \end{pmatrix} \begin{pmatrix} \Sigma_s & 0 \\ 0 & \Sigma_o \end{pmatrix} \begin{pmatrix} V_s^T \\ V_o^T \end{pmatrix} \qquad (3)$$

where Σ_s has the largest singular values on the main diagonal.

The state, input and output matrices are computed as [21]:

$$A = (J_1 U_s)^{\dagger} J_2 U_s \qquad (4)$$

$$C = J_3 U_s \qquad (5)$$

$$B = (I - A^{2N}) \Sigma_s V_s^T J_4 \qquad (6)$$

$$D = g_0 - C A^{2N-1} (I - A^{2N})^{-1} B \qquad (7)$$

where

$$J_1 = \begin{pmatrix} I_{(p-1)n_y} & 0_{(p-1)n_y \times n_y} \end{pmatrix} \qquad (8)$$

$$J_2 = \begin{pmatrix} 0_{(p-1)n_y \times n_y} & I_{(p-1)n_y} \end{pmatrix} \qquad (9)$$

$$J_3 = \begin{pmatrix} I_{n_y} & 0_{n_y \times (p-1)n_y} \end{pmatrix} \qquad (10)$$

$$J_3 = \begin{pmatrix} I_{n_u} \\ 0_{(p-1)n_{uy} \times n_u} \end{pmatrix} \qquad (11)$$

I and 0 represent the identity and 0 matrics of corresponding dimension; $Z^{\dagger}$ is the pseudo-inverse of matrix Z; n_y is the number of outputs and n_u represents the number of inputs. In our case, they are both 1; g_0 represents the 0th Markov coefficient, which has not been used in constructing the Hankel matrix.

C. *Identified model*

After reducing the model to order 3, we obtain the discrete state space representation (12), the discrete time transfer function (13) and by Tustin approximation, with sampling period $T_s = 5 \cdot 10^{-3}$ $[sec]$ the continuous time one in (14):

$$A = \begin{pmatrix} 0.94 & 0.3 & -0.01 \\ -0.08 & 0.53 & 0.18 \\ 0.003 & 0.09 & -0.92 \end{pmatrix} \quad B = \begin{pmatrix} 0.45 \\ 0.27 \\ -0.02 \end{pmatrix} \qquad (12)$$

$$C = \begin{pmatrix} 0.33 & -0.7 & 0.1 \end{pmatrix} \qquad D = (0.07)$$

$$H_{FIR}(z) = \frac{0.07448\, z^3 - 0.07989\, z^2 + 0.06108\, z + 0.1678}{z^3 - 0.545\, z^2 - 0.8581\, z + 0.5056} \qquad (13)$$

$$H_{FIR}(s) = \frac{0.2627\, s^3 + 1645\, s^2 - 3.714 \cdot 10^5\, s + 7.887 \cdot 10^7}{s^3 + 1.306 \cdot 10^4\, s^2 + 1.585 \cdot 10^6 s + 3.616 \cdot 10^7} \qquad (14)$$

For evaluating the model, we use the root mean square error (RMSE).

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N} (y_i - d_i)^2}{N}} \qquad (15)$$

where N is the number of samples, y is the output given by the model and d is the measured output.

The comparison of the output of (13) and the original signal can be seen in Fig.7. The RMSE for this model is 5.2247.

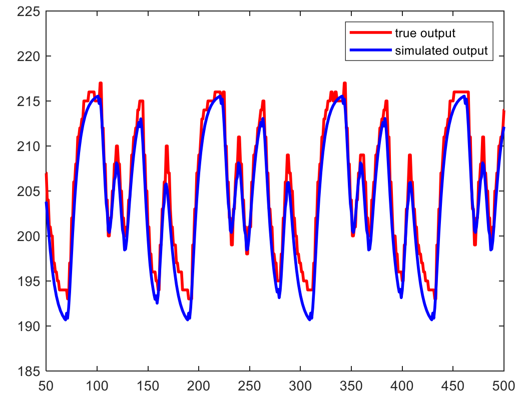

Fig. 7. Measured and simulated output

IV. Results and Discussion

As baseline, we consider n4sid (numerical subspace state space system identification) [22] implemented in Matlab. Using a 3rd order model an approximation of over 87% (with a RMSE = 2.5641) was obtained. The result can be seen in Fig. 8.

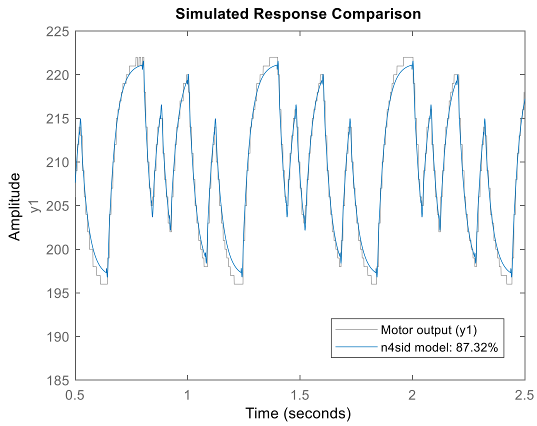

Fig. 8. Simulated output with n4sid

The obtained transfer function is given in (16) and in continuous time in (17).

$$H_{n4sid}(z) = \frac{0.01622\, z^2 + 0.01199\, z + 0.196}{z^3 - 0.8016\, z^2 - 0.3286\, z + 0.2322} \quad (16)$$

$$H_{n4sid}(s) = \frac{-0.16\, s^3 + 180.5\, s^2 - 7.53 \cdot 10^4\, s + 1.15 \cdot 10^7}{s^3 + 1556\, s^2 + 2.36 \cdot 10^5\, s + 5.264 \cdot 10^6} \quad (17)$$

Fig. 9 shows the response of the transfer functions (13) and (16) compared to the motor output for the PRBS input (yellow line). Both return satisfactory results and close to the motor output. Fig. 10 emphasizes the differences in response.

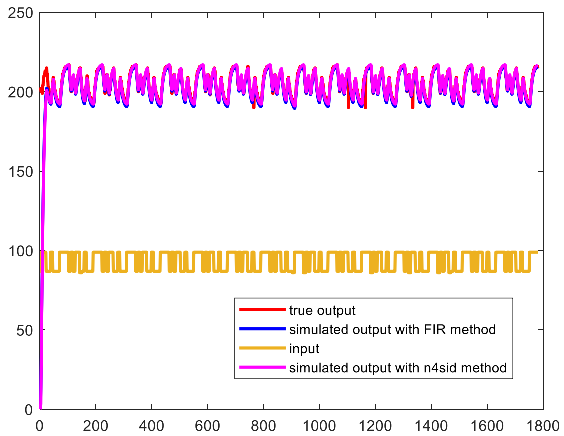

Fig. 9. Comparison between n4sid and FIR identification

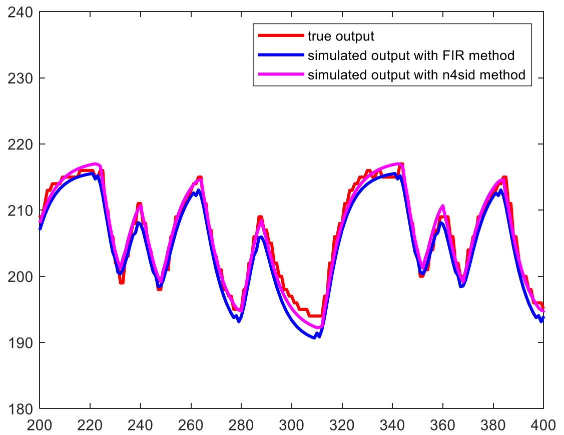

Fig. 10. Detailed comparison

V. Real-Time Results

Before real-time implementation, a series of timing experiments were conducted in Matlab. The execution time of the proposed method was approximately 0.020 seconds, while Matlab's built-in n4sid function required around 0.30 seconds to identify a model of the same order. Even though the built-in function is highly optimized, its execution time remains significantly higher. A manual implementation of the n4sid algorithm was also developed, similar to our method. This version achieved a model fit (77.05% vs. 87.32% for the Matlab version) and reduced execution time to approximately 0.055 seconds. However, this still represents more than double the execution time of our method. Given the computational limitations of low-cost embedded platforms, such latency differences become critical when frequent model updates are required. These results confirm that, although real-time subspace identification is theoretically feasible, the proposed FIR-based approach remains more practical for fast and repeated updates. Consequently, the identification steps for FIR method were implemented in real time on the development board.

The execution time to process a measurement is 1.5 ms. Then 24 ms are needed to calculate the FIR parameters and another 14 ms for matrices A, B, C, D. The matrices could be recalculated each 13 samples, corresponding to 65 ms, but the system is slowly varying and to allow for extra computation time (or parallel tests), the matrices are recalculated only every 100 ms, corresponding to 20 samples.

The transfer function obtained in real-time is:

$$H_{real-time}(z) = \frac{0.075\, z^3 - 0.080\, z^2 + 0.061\, z + 0.168}{z^3 - 0.545\, z^2 - 0.858\, z + 0.506} \quad (18)$$

which differs minimally from (13).

Comparing the Markov coefficients obtained in Matlab and the real-time development board, respectively, can be seen in Table 1. As can be seen, the difference is of the order 10^{-5}.

TABLE I. Comparison between Matlab and STM32

idx	**Markov Coefficients**		**idx**	**Markov Coefficients**	
	Matlab results	**Real-time results**		**Matlab results**	**Real-time results**
1	-0.00077	-0.00077	**18**	0.001017	0.001018
2	0.000762	0.000762	**19**	0.001275	0.001275
3	0.000651	0.00065	**20**	0.001378	0.001378
4	0.000363	0.000364	**21**	0.0000207	0.000021
5	0.001292	0.001292	**22**	0.000416	0.000416
6	0.001661	0.001661	**23**	0.000357	0.000358
7	0.001191	0.001191	**24**	0.000952	0.000952
8	0.002062	0.002062	**25**	0.000479	0.000478
9	0.001603	0.001603	**26**	0.000336	0.000337
10	0.002543	0.002542	**27**	0.000423	0.000422
11	0.002549	0.002548	**28**	0.000223	0.000224
12	0.001653	0.001653	**29**	0.0000693	0.000069
13	0.001226	0.001225	**30**	0.000552	0.000552
14	0.000903	0.000904	**31**	0.0004782	0.000478
15	0.00039	0.00039	**32**	0.000386	0.000386
16	0.001386	0.001385	**33**	0.000645	0.000644
17	0.000498	0.000497			

Simulating the system using these values with the same input, the output is indistinguishable from the result in Fig. 7. This is because the small differences between the Markov parameters obtained in real-time and those computed in MATLAB cause only minor variations in the system's response and transfer function.

VI. Conclusions

This paper presented a real time practical application to online identify a motor based on Markov coefficients, which can yield rapid results suitable for dynamic systems in a resource-constrained environment. The acquired data was processed using numerical filters. Adaptive sub-band filters were used to obtain the low-order finite impulse response filter coefficients. The state-space model was obtained starting from the Hankel matrix.

Real-time testing has been satisfactory, correctly identifying the parameters by obtaining the mathematical model using only the development board and acquired data.

Directions for further development include comparing the models obtained by other adaptive algorithms such as NLMS, RLS and AP; creating an interface for the user in choosing the adaptive algorithm for obtaining the Markov coefficients, for their number and for the order of the identified system. Another direction of development is to apply this algorithm to other: mechanical, electrical, etc systems. Moreover, it is desired to obtain a model from a larger number of FIR coefficients to improve approximation. This methodology can be extended to any system for which the input and output are known. Real-time identification directly from the development board is particularly valuable in scenarios where rapid adaptation to changing system dynamics is crucial, such as in control systems, fault detection, and system changes.

REFERENCES

[1] K. Karel J., System Identification: An Introduction, Springer, 2011.

[2] I.-D. Fîciu, J. Benesty, L. Dogariu, C. Paleologu and S. Ciochina, "Efficient Algorithms for Linear System Identification with Particular Symmetric Filters," *Applied Sciences,* vol. 12, pp. 42-63, 2022.

[3] V. Sandor, G. Keith R. and V. Peter, Numerical deconvolution using system identification methods, Springer, 1998.

[4] D. Kutluyıl, Partial-Update Adaptive Signal Processing. Design, Analysis and Implementation, Academic Press, 2009.

[5] R. K. Prasanth, "State Space System Identification Approach to Radar Data Processing," *IEEE Transactions on Signal Processing,* vol. 59, no. 8, pp. 3675-3684, 2011.

[6] P. Dobra, "Lecture notes: System's Theory," UTCN.

[7] D. Puangdownreong and S. Sarawut, "Image Based System Identification," *Int. Conference on Automatic Control, Modeling and Simulation,,* 2006.

[8] T. Arun K., Principles of System Identification: Theory and Practice, CRC Press, 2014.

[9] B. Wise and N. Ricker, "Identification of finite impulse response models by principal components regression: Frequency-response properties," *Process Control and Quality,* vol. 4, pp. 77-86, 1992.

[10] M. Zidane, S. Said, M. SABRI and A. Boumezzough, "Impulse Response Identification of Minimum and Non Minimum Phase Channels," *International Journal of Advanced Science and Technology,* vol. 64, pp. 59-72, 2014.

[11] S.-D. Sim, Z. Lendek and P. Dobra, "Implementation and testing of digital filters on STM32 Nucleo-64P," in *IEEE International Conference on Automation, Quality and Testing, Robotics*, Cluj-Napoca, Romania, 2022.

[12] K. A. Lee, W. S. Gan and S. M. Kuo, Subband adaptive filtering, John Wiley & Sons, 2009.

[13] S. Zahoor and S. Naseem, "Design and implementation of an efficient FIR," *Cogent Engineering,* vol. 4, 2017.

[14] T. Münker, J. Belz and O. Nelles, "Improved Incorporation of Prior Knowledge for Regularized FIR Model Identification," in *2018 Annual American Control Conference (ACC)*, Milwaukee, WI, USA, 2018.

[15] S. M. Kuo and D. R. Morgan, Active Noise Control Systems: Algorithms and DSP Implementations, New York: John Wiley & Sons, Inc., 1996.

[16] B. Widrow and E. Walach, Adaptive Inverse Control, Prentice Hall, 1996.

[17] J. R. Treichler, C. R. Johnson and M. G. Larimore, Theory and Design of Adaptive Filters, Prentice Hall, 2001.

[18] G. Yang and J. E. Meng, "Online adaptive fuzzy neural identification and control of a class of MIMO nonlinear systems," *IEEE Transactions on Fuzzy Systems,* vol. 11, no. 4, pp. 462-477, 2003.

[19] S.-D. Sim, A. Medgyesi, Z. Lendek, M. Susca, V. Mihaly, D. Pisla, P. Dobra, P. Dobra and A. Pica, "Adaptive filters for DC motor identification: a case study," in *IEEE International Conference on Automation, Quality and Testing, Robotics*, Cluj-Napoca, 2024.

[20] H. Yin, Z. Zhu and F. Ding, "Model order determination using the Hankel matrix of impulse responses," *Appl. Math. Lett.,* vol. 24, pp. 797-802, 2011.

[21] A. Damen, P. Van den Hof and A. Hajdasinski, "Approximate realization based upon an alternative to the Hankel matrix: the Page matrix," *Systems & Control Letters,* vol. 2, no. 4, pp. 202-208, 1982.

[22] L. Ljung, System Identification: Theory for the User, Upper Saddle River, NJ: Prentice Hall PTR, 1999, p. 132–134.