



HAL
open science

A New Recursive Instrumental Variables Approach for Robot Identification

Mathieu Brunot, Alexandre Janot

► **To cite this version:**

Mathieu Brunot, Alexandre Janot. A New Recursive Instrumental Variables Approach for Robot Identification. IFAC-PapersOnLine, 2018, 51 (15), pp.132-137. 10.1016/j.ifacol.2018.09.103 . hal-02472764

HAL Id: hal-02472764

<https://hal.science/hal-02472764>

Submitted on 10 Feb 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A New Recursive Instrumental Variables Approach for Robot Identification

M. Brunot and A. Janot

ONERA DTIS, 2 Avenue Edouard Belin, 31055 Toulouse, France (alexandre.janot@onera.fr and mathieu.brunot@onera.fr)

Abstract: The work presented in this paper focus on robot identification and presents a method based on the use of instrumental variables (IV). When dealing with *en-bloc* and offline identification of robots, the instrumental matrix constructed with the inverse dynamic model (IDM) and simulated data obtained from the simulation of the direct dynamic model (DDM). In this paper, a new recursive IV approach relevant for robot identification is presented. The instrumental matrix is constructed with the IDM and the references and their derivatives which are previously filtered by the transfer function of the position closed loop. This new way of building the instrumental matrix avoids the simulation of the DDM and offers some perspectives for online identification and real-time implementation. This recursive IV method termed IDIM-RIV (Inverse Dynamic Identification Model Recursive Instrumental Variables) is experimentally validated on the two degrees-of-freedom SCARA robot. Finally, some hints for real-time implementation are provided.

Keywords: closed-loop identification, robot identification, recursive instrumental variables

1. INTRODUCTION

Dynamic models used in robotics are continuous-time models resulting from law of Newton and, like all mechanical systems, robots exhibit a double-integrator behavior. They must be identified while they are operating in closed loop (Khalil & Dombre 2002), (Swevers *et al.* 2007) and (Gautier *et al.* 2013) among others. The standard identification method makes use of the inverse model (IDM) which is linear in relation to the dynamic parameters and the Least-Squares (LS) estimation. This standard method is termed IDIM-LS (Inverse Dynamic Identification Model Least Squares). Although good results can be obtained provided that an appropriate data filtering is used (Swevers *et al.* 2007) and (Gautier *et al.* 2013), the approach based on an instrumental variables (IV) method proposed in (Janot *et al.* 2014 a), termed IDIM-IV, should be preferred. Indeed, it is known that LS estimates are biased when the system is identified in closed loop (Van den Hof 1998) and (Gilson *et al.* 2011). However, both the IDIM-LS and IDIM-IV methods mentioned above are *en-bloc* techniques and their recursive variants were not discussed nor mentioned.

For robot identification, the Extended Kalman Filter (EKF) is one the most popular algorithms for recursive identification of the dynamic parameters, see e.g. (Gautier and Poignet 2001), (Lightcap and Banks 2010), (Ulrich and Sasiadek 2011) and (Masi *et al.* 2012). However, as pinpointed out in (Gautier and Poignet 2001), the EKF does not outperform the standard approach while the friction parameters are not well identified. This may be due to the fact that using the direct dynamic model (DDM) in order to estimate the dynamic parameters is not relevant because of the lack of sensitivity of the simulated joint positions, velocities and accelerations with respect to parameters' variations. This was emphasized in (Janot *et al.* 2014 b). In automatic control, recursive IV schemes have been deeply investigated and applied to linear

systems i.e. systems being linear in relation to both the states and parameters (Young 1970, 1981), (Söderström & Stoica 1983), (Söderström & Stoica 1989) and (Young 2011). Unfortunately, such recursive approaches are not popular in robotics. This may be explained by the fact that the systems considered in these references are linear which means that the researchers and engineers do not see a clear and straightforward way to apply them. In (Boeren *et al.* 2017), the authors present an interesting recursive IV approach that makes use of the IDM in order to identify the mechanical parameters of a nanopositioning system and they made some relationships with (Janot *et al.* 2014 a). However, the authors focused on a batching strategy i.e. the IV estimates are calculating over a whole trajectory and not at each time sample. Further, the system was still linear with respect to states and the parameters.

Motivated by the results presented in (Boeren *et al.* 2017), a recursive IV approach relevant for robot identification termed IDIM-RIV (Inverse Dynamic Identification Model Recursive Instrumental Variables) is presented in this paper. The instrumental matrix makes use of the IDM filled with the reference trajectories and their derivatives which are previously filtered by the transfer function of the position closed loop. Because simulated data are not used, this way of constructing the instrumental matrix avoids the simulation of the DDM and this offers interesting perspectives for online identification and real-time implementation. The experimental results obtained with a 2-DOF planar SCARA robot for online and online executions show the relevance of the IDIM-RIV method.

The paper is organized as follows: Section 2 reviews the background of robotics while Section 3 presents the IDIM-RIV method. The experimental results are given in Section 4 and Section 5 gives some concluding remarks.

2. BACKGROUND OF ROBOTICS

2.1 Inverse dynamic model of robots

The IDM of robots with n moving links calculates the $(n \times 1)$ joint torques vector $\boldsymbol{\tau}$ as a function of generalized coordinates and their derivatives (Khalil & Dombre 2002)

$$\boldsymbol{\tau} = \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{N}(\mathbf{q}, \dot{\mathbf{q}}), \quad (1)$$

where \mathbf{q} , $\dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$ are respectively the $(n \times 1)$ vectors of generalized joint positions, velocities and accelerations; $\mathbf{M}(\mathbf{q})$ is the $(n \times n)$ inertia matrix; $\mathbf{N}(\mathbf{q}, \dot{\mathbf{q}})$ is the $(n \times 1)$ vector of centrifugal, Coriolis, gravitational and friction torques. The modified Denavit and Hartenberg (DHM) notation allows the user to obtain an IDM which is linear in relation to the $(b \times 1)$ vector of base parameters $\boldsymbol{\beta}$, i.e.

$$\boldsymbol{\tau} = \mathbf{IDM}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\boldsymbol{\beta}, \quad (2)$$

where $\mathbf{IDM}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$ is the $(n \times b)$ matrix of basis functions of bodies dynamics; and b is the number of base parameters. Relation (2) represents the Inverse Dynamic Identification Model (IDIM). It is recalled that the base parameters are the minimum number of the dynamic parameters from which the IDM can be calculated. They are obtained from standard dynamic parameters by regrouping some of them with linear relations (Gautier & Khalil 1990). The standard parameters of a link j are: XX_j , XY_j , XZ_j , YY_j , YZ_j and ZZ_j the six components of the inertia matrix of link j at the origin of frame j ; MX_j , MY_j and MZ_j the components of the first moment of link j ; M_j the mass of link j ; Ia_j a total inertia moment for rotor and gears of actuator j ; Fv_j and Fc_j the viscous and Coulomb friction parameters of joint j .

2.2 Direct dynamic model of robots

The DDM of a robot expresses the joint accelerations as function of the joint torques, positions and velocities (Khalil & Dombre 2002). From Newton's law, we have

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} = \boldsymbol{\tau} - \mathbf{N}(\mathbf{q}, \dot{\mathbf{q}}). \quad (3)$$

Relation (3) shows that the DDM is nonlinear in relation to the states and the dynamic parameters. This is the reason why the DDM is rarely used in robot identification (Gautier *et al.* 2013).

2.3 Control and data acquisition

In robotics, Proportional-Derivative (PD) and Proportional-Integral-Derivative (PID) controls are often used to identify $\boldsymbol{\beta}$. The joint j torque, τ_j , can be thus written as

$$\tau_j = g_{\tau_j} v_j = g_{\tau_j} C_j(s)(q_{r_j} - q_j), \quad (4)$$

where $C_j(s)$ is the transfer function of the joint j

controller; q_{r_j} is the joint j position reference; q_j is the joint j position; g_{τ_j} is the joint j drive gain; v the joint j control signal; and s is the Laplace's variable. The data available from the controller are \mathbf{q} and the $(n \times 1)$ vector of the control signals \mathbf{v} , (Gautier *et al.* 2013).

2.4. The IDIM-LS identification method

The IDIM given by (2) is sampled at a measurement frequency deonted f_m while robot is tracking the reference trajectories $(\mathbf{q}_r, \dot{\mathbf{q}}_r, \ddot{\mathbf{q}}_r)$. In (2), \mathbf{q} is estimated with $\hat{\mathbf{q}}$ obtained by filtering the measurements of \mathbf{q} through a lowpass Butterworth filter in both the forward and reverse directions. $(\hat{\mathbf{q}}, \dot{\hat{\mathbf{q}}})$ are then calculated with a central differentiation algorithm of $\hat{\mathbf{q}}$. Finally, $\boldsymbol{\tau}$ being perturbed by high-frequency disturbances, a parallel decimation procedure is applied to eliminate torque ripples. All the details are given in (Gautier *et al.* 2013). After the data processing described previously, the following over-determined linear system is obtained

$$\mathbf{y}(\boldsymbol{\tau}) = \mathbf{X}(\hat{\mathbf{q}}, \dot{\hat{\mathbf{q}}}, \ddot{\hat{\mathbf{q}}})\boldsymbol{\beta} + \boldsymbol{\varepsilon}, \quad (5)$$

where $\mathbf{y}(\boldsymbol{\tau})$ is the $(r \times 1)$ measurements vector built from actual torques $\boldsymbol{\tau}$; $\mathbf{X}(\hat{\mathbf{q}}, \dot{\hat{\mathbf{q}}}, \ddot{\hat{\mathbf{q}}})$ is the $(r \times b)$ observation matrix built from $\mathbf{IDM}(\hat{\mathbf{q}}, \dot{\hat{\mathbf{q}}}, \ddot{\hat{\mathbf{q}}})$; $\boldsymbol{\varepsilon}$ is the $(r \times 1)$ vector of error terms; $r = n_e \cdot n$ is the number of rows in (5); and n_e is the number of samples of each joint j . The IDIM-LS estimates are given by

$$\hat{\boldsymbol{\beta}}_{LS} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}. \quad (6)$$

The statistics of the IDIM-LS method is detailed in (Janot *et al.* 2014 a).

2.5. The IDIM-IV method

The IDIM-LS method can be improved by adopting an IV approach that requires the construction of an instrumental matrix denoted \mathbf{Z} . To construct \mathbf{Z} , the DDM given by (3) is simulated with the previous IV estimates denoted as $\hat{\boldsymbol{\beta}}_{IV}^{i-1}$ and assuming the same references and the same control law structure for both the actual and the simulated robots. $\ddot{\mathbf{q}}_s$, the vector of the simulated joint accelerations, is given by $\mathbf{M}(\mathbf{q}_s, \dot{\hat{\boldsymbol{\beta}}}_{IV}^{i-1})\ddot{\mathbf{q}}_s = \boldsymbol{\tau}_s - \mathbf{N}(\mathbf{q}_s, \dot{\mathbf{q}}_s, \hat{\boldsymbol{\beta}}_{IV}^{i-1})$ where $\mathbf{q}_s, \dot{\mathbf{q}}_s$ are respectively the $(n \times 1)$ vectors of the simulated joint positions and velocities calculated by numerical integration of the DDM while $\boldsymbol{\tau}_s$ is the $(n \times 1)$ vector of simulated torques with τ_{s_j} , the j^{th} element of $\boldsymbol{\tau}_s$, is given by

$\tau_{s_j} = g_{\tau_j} C_j(s) (q_{r_j} - q_{s_j})$. The instrumental matrix \mathbf{Z} is given by

$$\mathbf{Z} = \mathbf{X}(\mathbf{q}_s, \dot{\mathbf{q}}_s, \ddot{\mathbf{q}}_s, \hat{\mathbf{p}}_{IV}^{it-1}), \quad (7)$$

where $\mathbf{X}(\mathbf{q}_s, \dot{\mathbf{q}}_s, \ddot{\mathbf{q}}_s, \hat{\mathbf{p}}_{IV}^{it-1})$ is the $(r \times b)$ sampled matrix of $\mathbf{IDM}(\mathbf{q}_s, \dot{\mathbf{q}}_s, \ddot{\mathbf{q}}_s, \hat{\mathbf{p}}_{IV}^{it-1})$. At iteration it , the IV estimates are given by

$$\hat{\mathbf{p}}_{IV}^{it} = (\mathbf{Z}^T \mathbf{X})^{-1} \mathbf{Z}^T \mathbf{y}. \quad (8)$$

This iterative process is run until convergence. The statistics of the IDIM-IV method is detailed in (Janot *et al.* 2014 a).

3. A RECURSIVE INSTRUMENTAL VARIABLES APPROACH FOR ROBOT IDENTIFICATION

In this section, a Recursive IV approach, termed IDIM-RIV, relevant for robot identification is presented. This section is the theoretical contribution of this paper.

3.1. Choice of the set of instruments

To be implementable online, the IDIM-RIV must avoid the simulation of the DDM. Otherwise, an IV variant based on a batching strategy must be preferred and adopted, as in (Boren *et al.* 2017).

To do so, let us first consider a joint j control illustrated in Fig. 1. In robotics, it is convenient to consider the nonlinear plant of robots as decoupled linear models. The joint j decoupled linear model denoted $P_j(s)$ is a double-integrator system, i.e. $P_j(s) = 1/J_j s^2$, while the nonlinear coupling term p_j is considered as a perturbation given by

$p_j = -\sum_{i \neq j}^n M_{j,i}(\mathbf{q}) \ddot{q}_i - N_j(q, \dot{q})$ where $M_{j,i}(\mathbf{q})$ is approximated by a constant inertia denoted J_j given by $J_j = ZZ_j + Ia_j + \max_q (M_{j,i}(\mathbf{q}) - ZZ_j - Ia_j)$. Let us now consider the closed-loop relations that are given by

$$q_j = H_j(s) q_{r_j} + D_{p_j}(s) p_j + D_{\varepsilon_j}(s) \varepsilon_{q_j}, \quad (9)$$

where $H_j(s) = g_{\tau_j} C_j(s) P_j(s) / Den(s)$,

$$D_{p_j}(s) = P_j(s) / Den(s), \quad D_{\varepsilon_j}(s) = 1 / Den(s), \quad \text{and}$$

$$Den(s) = 1 + g_{\tau_j} C_j(s) P_j(s).$$

From relation (6), q_{nf_j} , the noise-free part of q_j , is given by

$$q_{nf_j} = H_j(s) q_{r_j} + D_{p_j}(s) p_j, \quad (10)$$

and it follows that q_{nf_j} given by (10) must be utilized to construct \mathbf{Z} . However, because p_j still depends on the base

parameter, so does q_{nf_j} . It follows that choosing (10) to construct \mathbf{Z} still requires the simulation of the DDM or at least the calculation of p_j .

Let ω_{n_j} be the bandwidth of the joint j position closed loop.

In robotics, the gains within $C_j(s)$ are tuned such that the influence of p_j proves to be negligible in order to obtain a good tracking (Khalil & Khalil 2002). Hence, below ω_{n_j} , the following approximation holds

$$q_{nf_j} \approx H_j(s) q_{r_j} = q_{r_j}^f, \quad (11)$$

leading to construct \mathbf{Z} with (7) where \mathbf{q}_s and its derivatives is replaced with \mathbf{q}_r^f , the $(n \times 1)$ vector of filtered references and its derivatives. It yields that for the IDIM-RIV method, the instrumental matrix is then given by

$$\mathbf{Z} = \mathbf{X}(\mathbf{q}_r^f, \dot{\mathbf{q}}_r^f, \ddot{\mathbf{q}}_r^f). \quad (12)$$

Loosely speaking, the instrumental matrix constructed with (12) is the IDM filled with the sampled vector of reference trajectories filtered by the position closed loop. Like the IDIM-IV method, the IDIM-RIV approach still makes use of the IDM. However, unlike the IDIM-IV method, the IDIM-RIV technique does not require the simulation of the DDM. Indeed, the controller $C_j(s)$ as well as the value of J_j being known, the reference trajectories being designed and imposed by the user, and the basis functions of the IDM being literal expressions that can be explicitly calculated with the SYMORO+ software, it is therefore no longer necessary to simulate the DDM to construct \mathbf{Z} . It results that choosing $q_{r_j}^f$ defined by (11) instead of q_{s_j} allows the user to construct \mathbf{Z} before running experiments which means that \mathbf{Z} is constructed once and for all. As we shall see later, this result is particularly promising for real-time implementation of the IDIM-RIV method. Interestingly, in (Boren *et al.* 2017), the same kind of instrumental matrix is obtained in a somewhat different context.

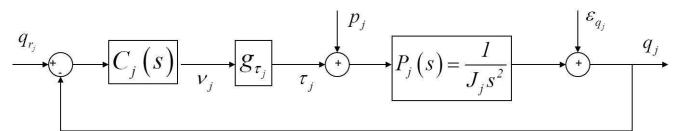


Fig. 1. Actual joint j

3.2. Scheme of the IDIM-RIV algorithm

The IDIM-RIV method is based on a standard recursive IV algorithm described by the following equations (see e.g. (Söderström and Stoica 1983) or (Young 2011) for the details) where \mathbf{x}_k (resp. \mathbf{z}_k) is the k th $(n \times b)$ line of \mathbf{X} (resp. \mathbf{Z}), \mathbf{k}_k is the $(b \times n)$ vector of correction at time k ;

and \mathbf{P}_k is the ($b \times b$) covariance of the IDIM-RIV estimates at time k :

- update of the gain at time k

$$\mathbf{k}_k = \mathbf{P}_{k-1} \mathbf{z}_{k-1}^T (\mathbf{I}_n + \mathbf{x}_{k-1} \mathbf{P}_{k-1} \mathbf{z}_{k-1}^T)^{-1}, \quad (13)$$

- update of the covariance matrix at time k

$$\mathbf{P}_k = \mathbf{P}_{k-1} - \mathbf{k}_k \mathbf{x}_{k-1} \mathbf{P}_{k-1}, \quad (14)$$

- prediction of the parameters at time k

$$\hat{\boldsymbol{\beta}}_k = \hat{\boldsymbol{\beta}}_{k-1} + \mathbf{k}_{k-1} (\mathbf{y}_k - \mathbf{x}_{k-1} \hat{\boldsymbol{\beta}}_{k-1}). \quad (15)$$

$\boldsymbol{\beta}$ is initialized with the Computer Aided Design values (CAD values) which are usually given by the manufacturers. It is also possible to get an initial value of $\boldsymbol{\beta}$ by running the IDIM-IV method which is not sensitive the initialization of $\boldsymbol{\beta}$. This way of doing is quite usual, see e.g. (Young 2011) p. 173.

3.3. Some hints for real-time implementation for embedded systems

As indicated in the previous sections, the IDIM-RIV method does not require the simulation of the DDM resulting in an *easy-to-implement* recursive IV approach suitable for embedded systems such as electric motors, mobile robots UAV's... Indeed, most of today microcontrollers are equipped of *large-size* volatile and non-volatile memory blocks as well as Floating Point Unit (FPU). It is recalled that volatile memory requires power to maintain the stored information (typically Random Access Memory - RAM) whereas non-volatile memory does not require power to maintain the information (typically Read Only Memory - ROM). Since the user has all the information needed to construct \mathbf{Z} in advance, it is thus possible to store \mathbf{Z} in a non-volatile memory and once the system powered on, to relocate it into a block of RAM in order to get an effective computing time. This relocation is needed because there are some *wait states* (or delays) when fetching the information stored in the non-volatile memory. This produces latencies while the code is running. Naturally, microcontroller must be chosen accordingly to the size of the matrices and vectors that are involved in the recursive IV algorithm; the larger the sizes of those matrices and vectors are, the more powerful microcontroller must be.

In addition, instructions including matrix operations (addition, multiplication, inversion...) can be dealt with optimized libraries that are provided by the manufacturers. However, when the number of parameters is limited (typically less than 5 which is the number of base parameters of a 1-dof robot), the instructions can be expanded in order to obtain classical algebraic expressions. This way of doing is quite usual in real-time programming and limits the number of *jumps* that produce some delays. It should be stressed that nonlinear functions (e.g. sine, cosine, square and sign that are

often present in the basis functions of the IDM) are also treated with appropriate and optimized libraries delivered by the manufactures. It is no longer required to approximate those nonlinear functions by a Taylor series.

As an indication, let us consider the TMS320F28069 manufactured and sold by Texas Instruments which has a clock rate of 90 MHz and is equipped of a FPU. The equations (13), (14) and (15) were implemented and executed in order to simulate the behavior of the EMPS prototype which has the following features $b=4$ and $n=1$. It takes only 80.10^{-6} s to execute the RIV algorithm.

4. EXPERIMENTAL VALIDATION

This section presents the experimental validations obtained with the SCARA robot. This section contents the experimental contribution of this paper.

4.1 Presentation of the SCARA robot

The SCARA robot presented Fig. 2 is a 2-dof planar robot. The SCARA robot is controlled with a simple PD controller whose the desired natural frequency ${}^d\omega_{n_j}$ is chosen according to the driving capacity without saturation of the joint drive. For this robot, we obtain a full bandwidth with ${}^d\omega_{n_1} = 1$ rad/s and ${}^d\omega_{n_2} = 10$ rad/s. It is worth noting that several controls including PID control and feedforward velocity/acceleration which give better tracking accuracy were tried and similar results have been obtained. This shows that the IDIM-RIV method is not sensitive to the control structure.



Fig. 2. SCARA robot

The data measurement frequency is $f_m = 200$ Hz. The torque data are calculated with (4) while the positions are obtained through incremental encoders (2000 and 5000 (lines/rev) for joint 1 and 2, respectively) with a 4-fold subdivision of each encoder line (8000 and 20000 (pulses/rev) for joint 1 and 2 respectively).

The reference trajectories, $(\mathbf{q}_r, \dot{\mathbf{q}}_r, \ddot{\mathbf{q}}_r)$, are fifth order polynomials and their duration is 12s. Since one obtains $\text{cond}(\mathbf{X}(\hat{\mathbf{q}}, \hat{\dot{\mathbf{q}}})) = 25$, the parameters are well excited (Gautier and Khalil 1992).

4.2 Offline experimental results

The IDIM-RIV approach is evaluated offline by comparing its results with those obtained with the *en-bloc* IDIM-IV method. The two methods are carried out with the filtered positions, $\hat{\mathbf{q}}$, calculated with a 10 Hz forward and reverse Butterworth filter and with the velocities, $\hat{\dot{\mathbf{q}}}$, the accelerations, $\hat{\ddot{\mathbf{q}}}$, calculated with a central differentiation algorithm of $\hat{\mathbf{q}}$. The maximum bandwidth for the second joint being $\omega_{dyn} = {}^d\omega_{n_2} = 10$ rad/s, this leads to choose $\omega_{fq} \geq 5\omega_{dyn}$, $\omega_{fq} \geq 50$ rad/s $\omega_{fq} = 8$ Hz. It is important to note that we can use a forward and reverse filter because the methods are evaluated offline; such filters cannot be used for online schemes. Finally, we choose a 10 Hz cutoff frequency. The parallel decimation is carried out with a lowpass Tchebyshev filter with a cutoff frequency $\omega_{fp} \geq 2\omega_{dyn}$, $\omega_{fp} \geq 20$ rad/s $\omega_{fp} = 3.18$ Hz. Then we choose a 4 Hz cutoff frequency. The sample rate f_m is divided by a factor $n_d = 20$. The IDIM-IV method starts with the regular initialization. The IDIM-RIV estimates are initialized with the CAD values while $\mathbf{P}_0 = 10^6 \mathbf{I}_8$. The relative deviations denoted $\% \sigma_{\hat{\beta}_j}$ are calculated as $\% \sigma_{\hat{\beta}_j} = 100 \cdot Cov(j, j) / |\hat{\beta}_j|$ where $Cov(j, j)$ is the j th diagonal element of the covariance matrix of the IDIM-IV or IDIM-RIV estimates. Finally, the relative errors denoted $\|\hat{\boldsymbol{\varepsilon}}\|/\|\mathbf{y}\|$ are computed with $\hat{\boldsymbol{\varepsilon}} = \mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}_{IDIM-IV}$ (resp. $\hat{\boldsymbol{\varepsilon}} = \mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}_{IDIM-RIV}$) for the IDIM-IV method (resp. IDIM-RIV method).

Table 1. IDIM-IV estimates and IDIM-RIV estimates averaged over the 10 last seconds

| Parameters | IDM-IV | $\% \sigma_{\hat{\beta}_j}$ | IDM-RIV | $\% \sigma_{\hat{\beta}_j}$ |
|---|--------|-----------------------------|---------|-----------------------------|
| ZZ1 | 3.446 | 1.2% | 3.445 | 1.3% |
| FV1 | 0.010 | 59.5% | 0.005 | 231.0% |
| FC1 | 0.833 | 2.5% | 0.815 | 5.3% |
| ZZ2 | 0.063 | 2.1% | 0.062 | 2.2% |
| MX2 | 0.249 | 1.2% | 0.249 | 1.4% |
| MY2 | 0.010 | 452.3% | 0.015 | 942.1% |
| FV2 | 0.021 | 6.3% | 0.020 | 9.5% |
| FC2 | 0.129 | 5.1% | 0.135 | 6.2% |
| $\ \hat{\boldsymbol{\varepsilon}}\ /\ \mathbf{y}\ $ | 3.9% | | 4.8% | |

The IDIM-IV estimates and the average values of the IDIM-RIV estimates calculated over the 10 last seconds i.e. when the variations of the IDIM-RIV estimates prove to be negligible are gathered in Table 1. Likewise their relative deviations and the relative errors, they are averaged over the ten last seconds. The IDIM-RIV estimates and their 95% interval bounds are plotted in Fig. 3 while in Fig. 4 are plotted the actual joint torque and the torque predicted with the IDIM-RIV estimates (direct comparisons). The IDIM-RIV estimates being comparable with the IDIM-IV estimates, the relative error being less than 5%, the IDIM-RIV estimates

converging rapidly to their steady-state values and the reconstructed torques matching the measured ones, it can be concluded that the IDIM-RIV method provides satisfactory results.

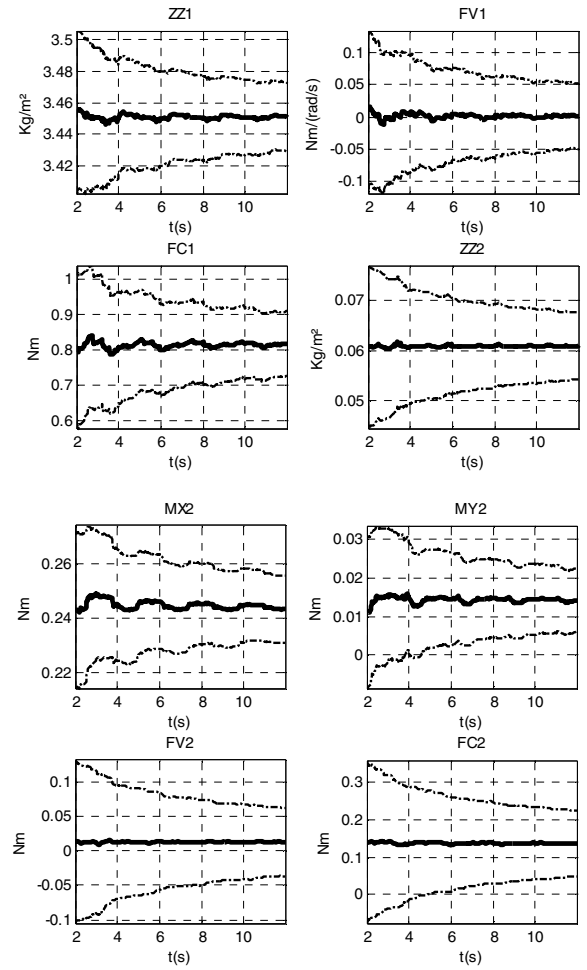


Fig. 3. Convergence of the parameters ZZ1, FV1, FC1, ZZ2, MX2, MY2, FV2 and FC2 – 95% interval confidence

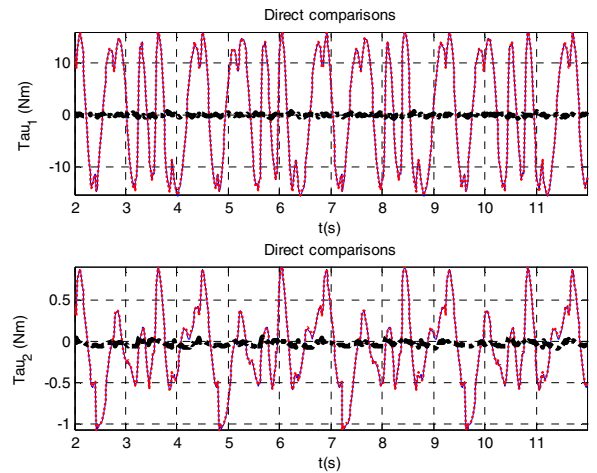


Fig. 4. Direct comparisons: blue, measurement; red, estimation and black, error

4.3. Online experimental results

The IDIM-RIV is now evaluated online. Because forward and reverse filters cannot be used, the data captured at time k i.e. the (2×1) vector of positions, \mathbf{q} , and the (2×1) vector of control signal, \mathbf{v} , are filtered with a stable low-pass filter with an unitary gain and a bandwidth of $\omega_{fq}=8$ Hz. $(\hat{\mathbf{q}}, \hat{\dot{\mathbf{q}}})$ are then calculated with a backward differentiation algorithm of $\hat{\mathbf{q}}$ i.e. $\hat{\dot{\mathbf{q}}}_k = (\hat{\mathbf{q}}_k - \hat{\mathbf{q}}_{k-1})/dt$ where dt is the sampling time with $dt = 1/f_m$. We still have $f_m = 200$ Hz. The IDIM-RIV estimates are initialized with the CAD values while $\mathbf{P}_0 = 10^6 \mathbf{I}_8$. Finally, the IDIM-RIV algorithm is executed via a Dspace card.

The results obtained with the online IDIM-RIV method are very close to those exposed in Section 4.2.: the IDIM-RIV estimates are comparable with the IDIM-IV estimates, the relative error are less than 5%, the IDIM-RIV estimates converge rapidly to their steady-state values and the reconstructed torques match the measured ones. They are not re-shown here for sake of clarity. The IDIM-RIV method executed online provides satisfactory results which means that the IDIM-RIV method presented in Section 3 can be applied for online identification of robots.

5. CONCLUSION

In this paper, a recursive instrumental variables approach suitable for identification of rigid industrial robots, termed IDIM-RIV, has been presented and validated offline and online with a 2 degrees-of-freedom SCARA robot. The instrumental matrix makes use of the inverse dynamic model and is constructed with the references and their derivatives filtered by the transfer function of the position closed loop. In so doing, the instrumental matrix is constructed once for all since the simulation of an auxiliary model is no longer necessary. This makes the IDIM-RIV method suitable for embedded systems. The experimental results showed that the IDIM-RIV provides estimates that are comparable with those obtained with the *en bloc* IDIM-IV method and has a rapid convergence.

Future works concern the use of the IDIM-RIV method to identify a 6-DOF industrial robot as well as its implantation and real-time programming for embedded systems. Finally, statistics will be addressed in order to obtain estimates with variances as small as possible.

REFERENCES

- Boeren, F., Bruijnen, D., and Oomen, T. (2017). Enhancing feedforward controller tuning via instrumental variables: with application to nanopositioning. *International Journal of Control*, Vol. 90(4), 2017
- Gautier, M. and Khalil, W. Exciting trajectories for the identification of the inertial parameters of robots. (1992) *International Journal of Robotics Research*, vol. 11, Aug. 1992, pp. 362-375.
- Gautier, M., and Poignet, P. (2001) Extended Kalman Filtering and Weighted Least-squares Dynamic Identification of Robot. *Control Engineering Practice*, Vol. 9, pp. 1361-1372.
- Gautier, M. Janot, A. and Vandanjon, P.O. (2013). A new closed-loop output error method for parameter identification of robot dynamics. *IEEE Transactions on Control System Technology*, Vol. 21(2), 2013, pp.428-444.
- Gilson M., Garnier H., Young P.C. and Van den Hof P. (2011). Optimal instrumental variable method for closed-loop identification. *IET on Control Theory & Applications*, vol. 5, no. 10, pp. 1147 – 1154.
- Janot, A., Vandanjon, P.O., and Gautier, M. (2014 a). A generic instrumental variable approach for industrial robots identification. *IEEE Transactions on Control Systems Technology*, 22(1), 2014.
- Janot, A., Gautier, M., Jubien A, Vandanjon, P.-O. (2014 b) Comparison between the CLOE Method and the DIDIM Method for Robots Identification. *IEEE Transactions on Control Systems Technology*, Vol. 22(5), pp. 1935-1941.
- Khalil, W. and Dombre, E. (2002). *Modeling, Identification and Control of Robots*. Hermes Penton, London, 2002.
- Lightcap, C.A. and Banks, A.B. (2010). An Extended Kalman Filter for Real-Time Estimation and Control of a Rigid-Link Flexible-Joint Manipulator. *IEEE Transactions on Control Systems Technology*, 18(1), 2010.
- Masi, A., Butcher, M., Martino, M., and Picatoste, R. (2012). An Application of the Extended Kalman Filter for a Sensorless Stepper Motor Drive Working With Long Cables. *IEEE Transactions on Industrial Electronics*, 59(11), 2012.
- Söderström, T. and Stoica, P. (1989). *System Identification*. Prentice Hall, 1989.
- Söderström, T., and Stoica, P. (1983). *Instrumental Variable Methods for System Identification*. Springer-Verlag, Berlin Heidelberg, 1983
- Swevers J, Verdonck J and De Schutter J. (2007). Dynamic model identification for industrial robots. *IEEE Control Systems Magazine*, pp. 58-71, Oct. 2007.
- Ulrich, S., and Sasiadek, J. Z. (2011). Extended Kalman filtering for flexible joint space robot control. *American Control Conference 2011 (ACC 2011)*, 29 June – 1 July 2011, San Francisco, CA, USA.
- Van den Hof, P.M.J. (1998). Closed loop issues in system identification. *Annual Reviews in Control*, 22, 173-186.
- Young, P.C. (2011). *Recursive Estimation and Time-Series Analysis: An Introduction for the Student and Practitioner*. Springer Verlag, 2nd Edition, 2011.
- Young, P.C. (1970) An instrumental variable method for real-time identification of a noisy process. *Automatica*, Vol. 6(2), pp. 271–287.
- Young, P.C. (1981) Parameter estimation for continuous-time models - a survey. *Automatica*, Vol. 17(1), pp. 23–39.