



HAL
open science

Improving Recursive Dynamic Parameter Estimation of Manipulators by knowing Robot's Model integrated in the Controller

Fabio Ardiani, Mourad Benoussaad, Alexandre Janot

► **To cite this version:**

Fabio Ardiani, Mourad Benoussaad, Alexandre Janot. Improving Recursive Dynamic Parameter Estimation of Manipulators by knowing Robot's Model integrated in the Controller. IFAC MATHMOD, Jul 2022, Vienne, Austria. pp.223 - 228, 10.1016/j.ifacol.2022.09.099 . hal-03812210

HAL Id: hal-03812210

<https://hal.science/hal-03812210>

Submitted on 12 Oct 2022

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.

Improving Recursive Dynamic Parameter Estimation of Manipulators by knowing Robot's Model integrated in the Controller

Fabio Ardiani^{*,**} Mourad Benoussaad^{**} Alexandre Janot^{*}

^{*} ONERA, Toulouse, France (e-mail: ardiani@onera.fr and alexandre.janot@onera.fr)

^{**} LGP-ENIT, University of Toulouse, Tarbes, France (e-mail: mourad.benoussaad@enit.fr)

Abstract: By identifying the manipulator's model that is integrated in some industrial model-based controllers, recursive parameters' estimation algorithms can be enhanced to have a better performance in online applications. In this paper, two improvements on recursive estimation of robot's dynamic parameter estimation are addressed. Firstly, the internal model can serve to initialize the parameters in recursive estimation algorithms, as the Recursive Least-Squares (RLS) and the Recursive Instrumental Variables (RIV). Secondly, the commanded position, which is used by the controller as a reference trajectory, can replace the external simulation of the dynamic model needed for recursive algorithms as the RIV. These two improvements make recursive algorithms more suitable for online application, specially RIV, where no data filtering nor external simulation needs to be done. Offline experimental validation on the KUKA LBR iiwa R820 is carried out, showing its feasibility for online application.

Copyright © 2022 The Authors. This is an open access article under the CC BY-NC-ND license (<https://creativecommons.org/licenses/by-nc-nd/4.0/>)

Keywords: Recursive Identification; Closed Loop Identification; Grey Box Modelling; Robotics Technology; Identification Methods; Robots Manipulators.

1. INTRODUCTION

The main approach and best theoretical solution for controlling a robotic manipulator is the use of the so-called computed torque control, which consists in linearizing and decoupling the dynamic equation of motion of the robot (Khalil and Dombre (2002), Moberg (2010)). This method, as well as most of the widely used model-based controllers (Moberg (2010)), requires an accurate dynamic model of the system. The more precise this model is, the better and faster the performance of the controller will be.

Nowadays, most of manipulators available in the market come along with their respective controller designed by the manufacturer. However, they do not usually provide a dynamic model of the robot nor, for safety reasons, the tools to modify the unknown controller's structure and parameters. Although these black-box controllers are usually built by a trajectory generator and several controlling stages, this lack of information is an issue for many branches of research where full knowledge of the system is needed. Thus identifying the model of the robot that is integrated in the controller is of main importance for research. In Jubien et al. (2014), the authors achieved to identify the so-called confidential parameters of the KUKA LWR4+ controller in a reverse-engineering approach by using measurements of joint torques and motor positions, as well as sampled data of the inertia matrix and gravity torques, which are computed by the KUKA controller.

Moreover, as human-robot interaction applications and situations where robots are subjected to dynamically un-

predictably changing environments are getting more and more popular (Ajoudani et al. (2018)), online identification of parameters is becoming crucial. Monitoring the evolution of the robot's parameters is an approach to implement powerful tools as adaptive controllers (Craig et al. (1987)) and faults and collisions detection (Östring (2002)). For this purpose, recursive algorithms as the Extended Kalman Filter (EKF) (Gautier and Poignet (2001), Lightcap and Banks (2010)), Recursive Least-Squares (RLS) (Kubus et al. (2008)) and Recursive Instrumental Variables (RIV) (Brunot and Janot (2018)) have been applied. The implementation of any of these algorithms present two main challenges. Firstly, the convergence and the speed of convergence are sensitive to initial conditions. In Gautier and Poignet (2001), the authors mention several ways to choose initial values depending on the amount of a priori knowledge. Good a priori knowledge, from CAD models or previous identification processes, will lead to faster identification results. Secondly, the algorithm computation needs to be fast enough to satisfy the online application speed. This is specially an issue for some recursive algorithms that make use of the dynamic model simulation, which is not only time-demanding, but sometimes not computationally possible. In order to solve this issue, batch techniques have been developed (Barfoot (2017)), but it can still be too slow for many applications in which the dynamics change quickly during task execution (e.g. human physical interaction). Therefore, authors in Brunot and Janot (2018) proposed a new way to build the instrumental matrix in the RIV method making use of

the reference trajectory and its derivatives, avoiding the simulation of the model.

The objective of this work is to tackle the two mentioned issues of recursive parameter identification of robots by using the knowledge of the identified robot model that is integrated in the controller and of its assumed structure. Indeed, knowing this model is a tool to initialize parameters in recursive algorithms. Moreover, thanks to the signals that some controllers make available and to the knowledge of its structure, RIV can be applied without the need of simulation of the dynamic model, making it more suitable for online application. This is an extension to industrial redundant collaborative manipulators of the work done in Brunot and Janot (2018) where tests of the RIV were carried out on a 2 dof SCARA manipulator. Although experimental tests of RLS and RIV are carried out on the KUKA LBR iiwa 14 R820 in an offline way to show results, the methods are also suitable and an interesting approach for online applications.

Section 2 reviews the structure and the model of collaborative manipulators, while Section 3 describes the algorithms and the methodology for dynamic parameter identification. Details on the experimental setup of the KUKA LBR iiwa 14 R820, as well as the design of trajectories, data acquisition and data processing are shown in Section 4. Finally, Section 5 discusses the experimental results and improvements on the performance of recursive identification methods, and Section 6 gives some concluding remarks.

2. MODELING

2.1 Structure of Collaborative Manipulators

The structure of a collaborative robot (Albu-Schäffer et al. (2007)) is graphically presented in Fig. 1. As these robots are intended to work alongside with humans, many times sharing not only objectives but also tasks, there are sensors measuring torques acting on the link side (*sensed torque*) (Albu-Schäffer et al. (2007)). They are also equipped with position sensors on the motor-side, and by knowing the gearbox characteristics, manufacturers usually make the link position available for users (*sensed position*).

As mentioned previously, model-based controllers are commonly used, and it could be assumed that the *commanded position* and *commanded torque* given by these controllers, as KUKA controllers, are the input and output of the dynamic model of the robot that is integrated in them.

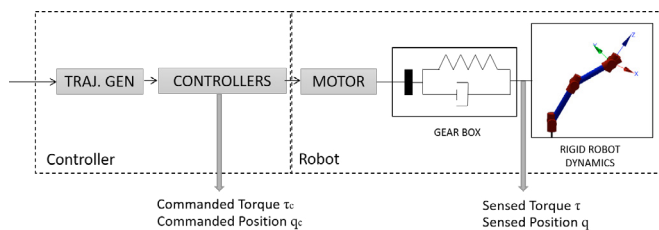


Fig. 1. KUKA Industrial Collaborative Manipulator Structure (Albu-Schäffer et al. (2007)).

2.2 Equations of Motion

The equation describing the inverse dynamic model (IDM) of a n-dof serial manipulator without considering external forces, elasticity nor couplings between links and motors is (Khalil and Dombre (2002)):

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

where $\boldsymbol{\tau} \in \mathbb{R}^n$ is the vector of the link-side torques; $\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}} \in \mathbb{R}^n$ are the joint positions, velocities and accelerations, respectively; $\mathbf{H}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} \in \mathbb{R}^n$ is the vector of the Coriolis and centrifugal forces; $\mathbf{G}(\mathbf{q}) \in \mathbb{R}^n$ stands for gravity torques; $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{n \times n}$ is the symmetric positive definite inertia matrix; and $\boldsymbol{\tau}_f \in \mathbb{R}^n$ are the friction torques. The friction model of joint i used here is the one described in Hamon et al. (2011):

$$\tau_{fi} = F_{si} \text{sign}(\dot{q}_i) + F_{vi} \dot{q}_i + \tau_{offi}, \quad (2)$$

where τ_{fi} is the i th component of the friction vector; F_{si} and F_{vi} are Coulomb and viscous friction parameters respectively; and τ_{offi} is an offset in torque measurements regrouping amplifiers and sensors offsets and the asymmetrical Coulomb friction coefficient.

The IDM in (1) can be expressed in a linear way w.r.t. a set of standard dynamic parameters $\boldsymbol{\chi}_{st}$ as:

$$\boldsymbol{\tau} = \mathbf{IDM}_{st}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\boldsymbol{\chi}_{st}, \quad (3)$$

where $\mathbf{IDM}_{st}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$ is the regressor matrix. $\boldsymbol{\chi}_{st}$ is composed by 13 standard parameters from each link i :

$$\boldsymbol{\chi}_{st}^i = [XX_i \ XY_i \ XZ_i \ YY_i \ YZ_i \ ZZ_i \ MX_i \ MY_i \ MZ_i \ M_i \ F_{si} \ F_{vi} \ \tau_{offi}]^T. \quad (4)$$

where $XX_i, XY_i, XZ_i, YY_i, YZ_i$ and ZZ_i are the 6 components of the inertia tensor considered from the origin of the link; MX_i, MY_i and MZ_i are the components of its first moment of inertia; and M_i is its mass.

Furthermore, the base parameters are commonly defined as a minimum set of identifiable parameters with which the dynamic model can be calculated. It is obtained from the reduction of the set of standard parameters by, eliminating some which have no effect on the model and regrouping others which are not identifiable separately (Gautier and Khalil (1990)). The model in terms of base parameters can be expressed in the same way as (3), with the corresponding changes in the regressor matrix and the parameters' vector.

After sampling the IDM during a period of time, an over-determined linear system with all the measurements can be constructed in the following way:

$$\mathbf{Y} = \mathbf{W}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\mathbf{X} + \boldsymbol{\rho}, \quad (5)$$

where \mathbf{Y} is the vector of torques; $\mathbf{W}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$ is the observation matrix; \mathbf{X} is the set of base parameters; and $\boldsymbol{\rho}$ refers to the vector of errors due to noise and not modeled effects. From these base parameters, there may be some of them that, given the measured data, do not substantially affect the model. If they are deleted, the set of essential parameters is obtained (Pham and Gautier (1991)). In Janot et al. (2014), it is suggested to ignore those base parameters with a relative standard deviation higher than 20%.

3. IDENTIFICATION METHODOLOGY

First, an analysis of the commanded signals (*commanded position* and *commanded torque*) and the sensed signals (*sensed position* and *sensed torque*) is required to show that the controller is one of a model-based type. Then, the parameters of the robot's model that is integrated in the controller can be identified offline using the commanded signals and the LS method. This identified model will be used for the initialization of the recursive algorithms RLS and RIV in order to identify the real robot's parameters. RLS will be applied using sensed signals, and RIV using *commanded position* for the instrumental matrix generation and sensed signals. In this section, the three mentioned identification methods are briefly detailed.

3.1 Least-Squares

The LS solution $\hat{\mathbf{X}}_{LS}$ minimizes the squared 2-norm of the vector of errors:

$$\hat{\mathbf{X}}_{LS} = (\mathbf{W}^T \mathbf{W})^{-1} \mathbf{W}^T \mathbf{Y}. \quad (6)$$

The covariance matrix of the estimation error is:

$$\mathbf{C}_{\hat{\mathbf{X}}_{LS}} = \frac{\|\mathbf{Y} - \mathbf{W} \hat{\mathbf{X}}_{LS}\|}{r - n_b} (\mathbf{W}^T \mathbf{W})^{-1}, \quad (7)$$

where r is the product between the amount of measurements n_m and the number of joints n , and n_b is the amount of base parameters. Then, the relative standard deviation of the i th estimate \hat{X}_{LS}^i is given by:

$$\% \hat{\sigma}_{\hat{X}_{LS}^i} = 100 \frac{\sqrt{C_{\hat{X}_{LS}}^{ii}}}{\hat{X}_{LS}^i}, \quad (8)$$

where $C_{\hat{X}_{LS}}^{ii}$ are the diagonal coefficients of $\mathbf{C}_{\hat{\mathbf{X}}_{LS}}$.

The main challenge for this method, lays in the data processing and the way to ensure noise properties. $\hat{\mathbf{X}}_{LS}$ is considered to be the best linear unbiased estimator if the error terms have zero mean, they are homoscedastic (constant finite variance), they do not present multicollinearity and they are not autocorrelated.

3.2 Recursive Least-Squares

LS is a method to be applied once all the measurements are made. In order to be able to implement it online, as measurements are carried out, its recursive version is necessary. The RLS solution $\hat{\mathbf{X}}_{RLS}$ is given by the following algorithm (Young (2011)):

$$\begin{aligned} \mathbf{K}_k &= \mathbf{P}_{k-1} \mathbf{W}_k^T (\mathbf{I}_n + \mathbf{W}_k \mathbf{P}_{k-1} \mathbf{W}_k^T)^{-1} \\ \mathbf{P}_k &= \mathbf{P}_{k-1} - \mathbf{K}_k \mathbf{W}_k \mathbf{P}_{k-1} \\ \hat{\mathbf{X}}_{RLS}^k &= \hat{\mathbf{X}}_{RLS}^{k-1} + \mathbf{K}_k (\mathbf{Y}_k - \mathbf{W}_k \hat{\mathbf{X}}_{RLS}^{k-1}) \end{aligned} \quad (9)$$

where \mathbf{P}_k , \mathbf{K}_k , \mathbf{W}_k , \mathbf{Y}_k and $\hat{\mathbf{X}}_{RLS}^k$ are the covariance matrix, the estimator gain matrix, the observation matrix, the vector of torques and the vector of RLS estimates respectively, at iteration k ($k = 1, \dots, n_m$) being n_m the amount of measurements, and \mathbf{I}_n is the identity matrix of order n .

In general, when no a priori information is available, $\hat{\mathbf{X}}_{RLS}$ should be started at zero and \mathbf{P} as a diagonal matrix

with large elements. In robotics, this brings numerical problems, and an alternative is to initialize them with low values and the adequate sign. The choice of these initial values affects the stability and speed convergence of the algorithm (Gautier and Poignet (2001)). Therefore, the best way to initialize values is with proper a priori knowledge, either by having the CAD model or by carrying out another identification process before applying the recursive algorithm.

3.3 Recursive Instrumental Variables

The RIV solution $\hat{\mathbf{X}}_{RIV}$ is closely related to $\hat{\mathbf{X}}_{RLS}$ (Young (2011)):

$$\begin{aligned} \mathbf{K}_k &= \mathbf{P}_{k-1} \mathbf{Z}_k^T (\mathbf{I}_n + \mathbf{W}_k \mathbf{P}_{k-1} \mathbf{Z}_k^T)^{-1} \\ \mathbf{P}_k &= \mathbf{P}_{k-1} - \mathbf{K}_k \mathbf{W}_k \mathbf{P}_{k-1} \\ \hat{\mathbf{X}}_{RIV}^k &= \hat{\mathbf{X}}_{RIV}^{k-1} + \mathbf{K}_k (\mathbf{Y}_k - \mathbf{W}_k \hat{\mathbf{X}}_{RIV}^{k-1}) \end{aligned} \quad (10)$$

being \mathbf{Z}_k and $\hat{\mathbf{X}}_{RIV}^k$ are the instrumental matrix and the RIV estimates at iteration k , respectively.

This method is more robust against noise than RLS, but has the problem of how to choose \mathbf{Z} for optimal results. As mentioned in Janot et al. (2014), in robotics, a valid auxiliary model for the generation of the instrumental matrix is the dynamic model. However, computational speed and numerical issues, makes the choice not suitable for online application. To remove the need to simulate the robot's model online, the authors in Brunot and Janot (2018) proved that the instrumental matrix can be also built with the reference trajectories of position, velocity and acceleration sent to the robot, which are usually computed before the robot's movement.

4. EXPERIMENTAL SETUP

4.1 Robotic Manipulator

To validate the approach, the 7 dof lightweight manipulator KUKA LBR iiwa R820 was used (KUKA (2019)). Its kinematic parameters following the modified Denavit-Hartenberg (MDH) convention are shown and summarized in Fig. 2. Considering the dynamic model in (1) and (2), and the closed-loop rules for model reduction from Gautier and Khalil (1990), 64 base parameters are derived from the 91 standard parameters. Table 1 shows the 17 base parameters which are the ones obtained from the regrouping of standard parameters. The other 47 base parameters are the ones related to friction (F_{si} , F_{vi} and τ_{offi} with $i = 1, \dots, 7$), and other inertial parameters (\mathbf{ZZ}_7 , \mathbf{MY}_7 and \mathbf{MX}_i , \mathbf{XY}_i , \mathbf{XZ}_i and \mathbf{YZ}_i , with $i = 2, \dots, 7$).

The KUKA iiwa brings a load identification routine that allows the controller to compensate the dynamics of the load. As the robot is equipped with the "Media Flange Touch Electrical" on its tip (KUKA (2015)), this routine has been performed before carrying out any experiment.

4.2 Trajectories, Data Acquisition and Processing

50 configurations of the robot are chosen in the useful work-space of the robot in order to generate trajectories. Constraints regarding joint limits and platform position

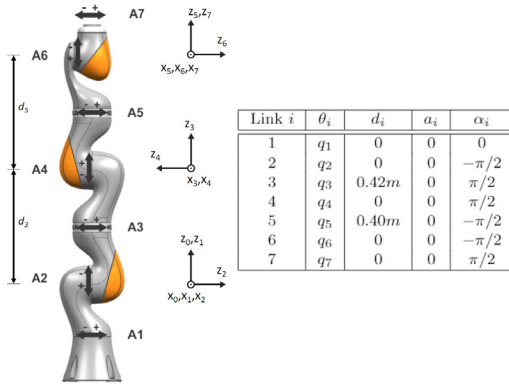


Fig. 2. MDH parameters and link frames of the KUKA LBR iiwa 14 R820.

Table 1. Regrouped base parameters of the KUKA LBR iiwa following the MDH convention stated in Fig. 2.

$$\begin{aligned}
 \overline{ZZR}_1 &= \overline{ZZ}_1 + \overline{YY}_2 \\
 \overline{XXR}_2 &= \overline{XX}_2 - \overline{YY}_2 + \overline{YY}_3 + 2d_3 \overline{MZ}_3 + (M_3 + M_4 + M_5 + M_6 + M_7)d_3^2 \\
 \overline{ZZR}_2 &= \overline{ZZ}_2 + \overline{YY}_3 + 2d_3 \overline{MZ}_3 + (M_3 + M_4 + M_5 + M_6 + M_7)d_3^2 \\
 \overline{MYR}_2 &= \overline{MY}_2 - \overline{MZ}_3 - (M_3 + M_4 + M_5 + M_6 + M_7)d_3 \\
 \overline{XXR}_3 &= \overline{XX}_3 - \overline{YY}_3 + \overline{YY}_4 \\
 \overline{ZZR}_3 &= \overline{ZZ}_3 + \overline{YY}_4 \\
 \overline{MYR}_3 &= \overline{MY}_3 - \overline{MZ}_4 \\
 \overline{XXR}_4 &= \overline{XX}_4 - \overline{YY}_4 + \overline{YY}_5 + 2d_5 \overline{MZ}_5 + (M_5 + M_6 + M_7)d_5^2 \\
 \overline{ZZR}_4 &= \overline{ZZ}_4 + \overline{YY}_5 + 2d_5 \overline{MZ}_5 + (M_5 + M_6 + M_7)d_5^2 \\
 \overline{MYR}_4 &= \overline{MY}_4 + \overline{MZ}_5 + (M_5 + M_6 + M_7)d_5 \\
 \overline{XXR}_5 &= \overline{XX}_5 - \overline{YY}_5 + \overline{YY}_6 \\
 \overline{ZZR}_5 &= \overline{ZZ}_5 + \overline{YY}_6 \\
 \overline{MYR}_5 &= \overline{MY}_5 + \overline{MZ}_6 \\
 \overline{XXR}_6 &= \overline{XX}_6 - \overline{YY}_6 + \overline{YY}_7 \\
 \overline{ZZR}_6 &= \overline{ZZ}_6 + \overline{YY}_7 \\
 \overline{MYR}_6 &= \overline{MY}_6 - \overline{MZ}_7 \\
 \overline{XXR}_7 &= \overline{XX}_7 - \overline{YY}_7
 \end{aligned}$$

are considered to avoid collisions. We used the KUKA Sunrise OS to interpolate these points, and to create the position, velocity and acceleration profiles with the Spline and PTP motion types (KUKA (2017)). This method of generating trajectories (Jubien et al. (2014)) yields good results in terms of the condition number of the regressor matrix without any time and resource consuming non-linear optimization (Khalil and Dombre (2002)). This kind of trajectory is more real application-like, which makes it more suitable for online application. Indeed, trajectories specifically generated for parameters identification may not be applied when executing a task.

The *Fast Robot Interface* (FRI), a library provided by KUKA, allows the user to continuously exchange data in real time between the robot controller and a C++ client application on an external system (Schreiber et al. (2010)). In this work, the client application records data from the robot at its highest possible rate of 1000 Hz.

After data acquisition, the data is processed. For the commanded signals, the data is first downsampled to 50 Hz and then velocities and accelerations are calculated from the *commanded position*. In addition, when using the sensed signals for RLS, as noise conditions (see Section 3.1) cannot be ensured, a filter is needed. Therefore, a 2nd order digital Butterworth filter with cutoff frequency of 3.5 Hz in both directions to avoid time-lag is applied before the

downsample process. Moreover, points where the absolute value of the velocity of any joint is lower than 0.01 rad/s are deleted to avoid problems due to the discontinuity around zero velocity of the friction model in (2).

5. RESULTS & DISCUSSION

5.1 Commanded vs Sensed Signals

To validate the assumption regarding the controller being model-based, a comparison between the commanded and sensed signals that the KUKA controller makes available is done. This is important in order to analyze the behavior of the controller, and verify if the decoupling of the non-linear model is well-done by KUKA.

In Fig. 3, the respective signals from joint 4 and their difference are depicted. Results from joint 4 are chosen to be shown as it is a central link, with a lower relative effect of noise and payload uncertainties w.r.t. applied torque. This is not the case of links that are at the end of the kinematic chain, where noise has more relative impact on the behavior.

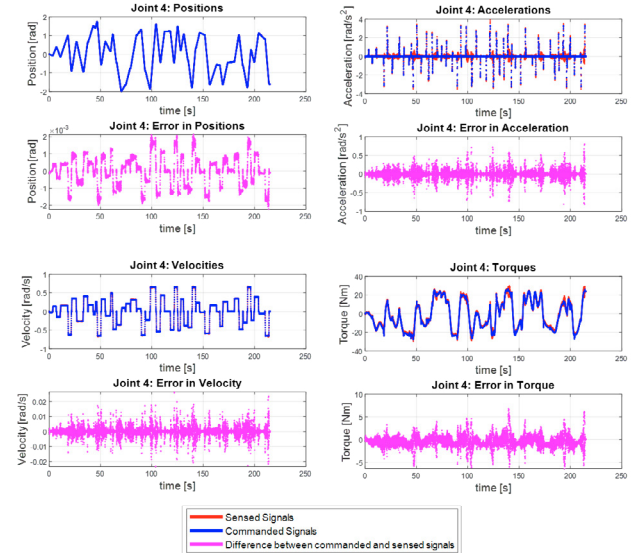


Fig. 3. Commanded Signals, Sensed Signals, and difference between both for joint 4.

The error between the sensed position and commanded position is lower than 1%, showing good performance of the KUKA controller. Moreover, the position error is bounded and has a profile similar to the one of the velocity. This means that, for model-based controllers, the inertial effects are well decoupled and compensated with the internal model. As explained in Khalil and Dombre (2002), an adequate linearization and decoupling of the equations of the model, provides a uniform dynamic behavior for every joint whatever the configuration of the robot. Moreover, it can be seen in Fig. 3 that noise has significant effects in the torque and acceleration signals.

5.2 Parameters Identification of Manipulator's Model Integrated in Controller with LS

The set of essential parameters obtained by applying the LS method using the non-filtered commanded position

and the non-filtered commanded torque, and by removing those parameters with a relative standard deviation higher than 20% (Janot et al. (2014)), is shown in Table 2. 21 out of the 64 base parameters are identified. The friction parameters are not identified in this set, which means that, a priori, friction is not included in the controller's model. All the main inertial parameters (ZZR_i for $i = 1, \dots, 7$; MYR_i for $i = 2, \dots, 6$; and XXR_i for links with movement against gravity, thus 2, 4 and 6) are identified with low relative standard deviation. Moreover, all the identified parameters in Table 2 are physically consistent, which allows the simulation and control of the manipulator.

Table 2. Identified set of essential parameters and their respective relative standard deviation.

Param.	Value	$\% \sigma$	Param.	Value	$\% \sigma$
ZZR_1	0.0419	5.17	ZZR_5	0.0094	10.39
XXR_2	2.3452	0.30	YZ_5	-0.0046	10.39
ZZR_2	2.3963	0.21	MYR_5	-0.0741	0.54
MYR_2	-5.9076	0.02	XXR_6	0.0196	4.94
ZZR_3	0.0753	2.35	ZZR_6	0.0185	4.20
XZ_3	-0.0065	18.32	MYR_6	-0.1439	0.25
MYR_3	-0.0159	3.07	ZZ_7	0.0073	7.60
XXR_4	0.8064	0.23	XY_7	-0.0030	12.17
ZZR_4	0.8259	0.1634	YZ_7	-0.0026	13.98
MYR_4	2.4083	0.02	MX_7	0.0244	1.04
			MY_7	0.0259	1.08

Moreover, to validate this model, Table 3 shows the percent error of the torque reconstruction using two new validation trajectories. A percent error lower than 11% for all joints validates the model obtained. It can also be seen that central links have a torque reconstruction percent error lower than 3.5%, being higher for links 1 and 7 due to the mentioned higher effect of noise.

Table 3. Percent error [%] of commanded torque reconstruction using the model in Table 2 and two new validation trajectories.

Joint	1	2	3	4	5	6	7
Traj. 1	3.44	0.45	0.56	0.61	1.80	1.68	3.83
Traj. 2	10.07	1.14	3.26	1.25	2.87	2.51	5.31

5.3 Parameter Initialization Effect on RLS

To evaluate the effect of parameters initialization in RLS two situations are compared: when no a priori information is available and when the set of essential parameters from Table 2 is considered. Fig. 4 shows the evolution of some identified parameters of joint 4 for these two situations. In the case of initializing the parameters with no a priori information, all values were started at 0 with a variance of 100. When having a priori information, the essential parameters were started as in Table 2, and all the non-essential parameters (inertial and friction) at 0. Variance values were started at 0.01 for essential parameters, at 1 for the non-essential inertial parameters and at 100 for friction, answering to the different degrees of parameters' knowledge. The choice of the magnitude of the variance locks or lets free the parameters' variation (Gautier and Poignet (2001)). Moreover, the filtering process described in Section 4 was carried out before the identification

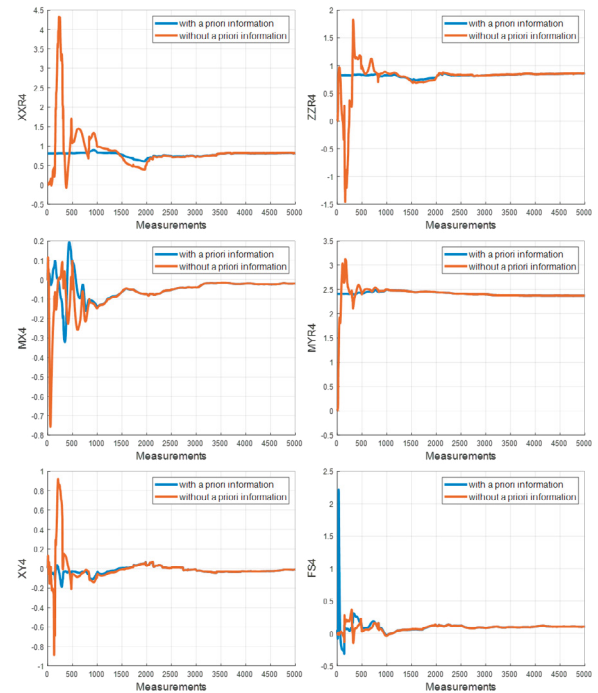


Fig. 4. Evolution of some base parameters of joint 4 using RLS with and without a priori information.

process, as assumptions mentioned in Section 3.1 regarding the noise cannot be ensured.

In between 1000 and 2000 measurements, both methods converge to the same results. As it was expected, main inertial parameters' estimations converge faster when initializing the parameters with the identified model from Section 5.2. This is a substantial advantage for the beginning of the execution, reducing the risk of having a non-computable dynamic model. However, friction parameters (FS4 in Fig.4) vary significantly more, as they are initialized with much higher variance than the other parameters. Nevertheless, it cannot cause problems in terms of calculation because this issue comes from the non-invertibility of the inertia matrix $\mathbf{M}(\mathbf{q})$ in (1) which friction does not affect.

These results show two things. First, initializing parameters in recursive algorithms with the values identified of the model that is integrated in the controller is better than initializing them with no a priori knowledge for an online identification context. This method also avoids the need of CAD models, which are often not provided by manufacturers. Furthermore, although the real robot parameters are not exactly the same as the ones inside the model of the controller, they are a good approximation, and they can be used to control the robot.

5.4 RIV with Commanded Position

To compare with RLS results obtained in the previous subsection, RIV results are analyzed. Parameter identification using RLS with filtered sensed signals, and using RIV with non-filtered sensed signals and commanded position for the instrumental matrix generation, are shown in Fig. 5. For simplicity, some of the parameters of joint 4 shown in the

previous subsection are depicted. Initial values for both methods are the ones shown in Table 2 used in Section 5.3.

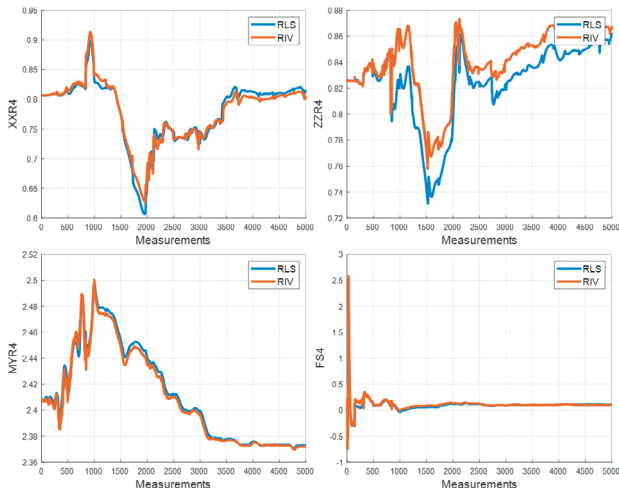


Fig. 5. Evolution of some base parameters of joint 4 using RLS and RIV methods.

Results are very close for both methods, and it is not possible to demonstrate with this study which is more precise, as the real values are unknown. However, the fact that results have a difference of less than 2% between each other, and that no filtering process was made to the signals in RIV, presents a huge advantage of this method over others for online application.

6. CONCLUSIONS

In this work, two improvements for recursive identification of dynamic parameters of robotic manipulators using the knowledge of the model that is integrated in the controller were discussed. First, after analyzing that the commanded signals given by the KUKA LBR iiwa 14 R820 controller are the input and output from the robot's model that is integrated in it, this model was identified using LS method. These values were used to initialize values in RLS and RIV methods, and it was shown that this approach is a proper alternative to arbitrary or CAD-based initialization of parameters. Secondly, RIV method was successfully applied by generating the instrumental matrix with the commanded position signal given by the KUKA controller. This method is an interesting approach for recursive online estimation, as it does not use any additional simulation of the dynamic model nor filtering.

REFERENCES

- Ajoudani, A., Zanchettin, A.M., Ivaldi, S., Albu-Schäffer, A., Kosuge, K., and Khatib, O. (2018). Progress and prospects of the human–robot collaboration. *Autonomous Robots*, 42(5), 957–975.
- Albu-Schäffer, A., Haddadin, S., Ott, C., Stemmer, A., Wimböck, T., and Hirzinger, G. (2007). The dlr lightweight robot: design and control concepts for robots in human environments. *Industrial Robot: an international journal*.
- Barfoot, T. (2017). *State estimation for robotics*. Cambridge University Press.
- Brunot, M. and Janot, A. (2018). A new recursive instrumental variables approach for robot identification. *IFAC-PapersOnLine*, 51, 132–137.
- Craig, J.J., Hsu, P., and Sastry, S.S. (1987). Adaptive control of mechanical manipulators. *The International Journal of Robotics Research*, 6(2), 16–28.
- Gautier, M. and Poignet, P. (2001). Extended kalman filtering and weighted least squares dynamic identification of robot. *Control Engineering Practice*, 9(12), 1361–1372.
- Gautier, M. and Khalil, W. (1990). Direct calculation of minimum set of inertial parameters of serial robots. *IEEE Transactions on robotics and Automation*, 6(3), 368–373.
- Hamon, P., Gautier, M., and Garrec, P. (2011). New dry friction model with load- and velocity-dependence and dynamic identification of multi-dof robots. In *2011 IEEE International Conference on Robotics and Automation*, 1077–1084.
- Janot, A., Vandanjon, P.O., and Gautier, M. (2014). A generic instrumental variable approach for industrial robot identification. *IEEE Transactions on Control Systems Technology*, 22(1), 132–145.
- Jubien, A., Gautier, M., and Janot, A. (2014). Dynamic identification of the kuka lightweight robot: Comparison between actual and confidential kuka's parameters. In *IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, 483–488.
- Khalil, W. and Dombre, E. (2002). *Modeling identification and control of robots*. CRC Press.
- Kubus, D., Kroger, T., and Wahl, F.M. (2008). On-line estimation of inertial parameters using a recursive total least-squares approach. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 3845–3852.
- KUKA (2015). *KUKA Robots Media Flange for Product Family LBR iiwa*. Pub Option Medien-Flansch, KUKA Augsburg, Germany.
- KUKA (2017). *KUKA Sunrise OS 1.13*. Pub Sunrise OS, KUKA Augsburg, Germany.
- KUKA (2019). *KUKA Robots LBR iiwa 7 R800, LBR iiwa 14 R820*. Pub Spez LBR, KUKA Augsburg, Germany.
- Lightcap, C.A. and Banks, S.A. (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), 91–103.
- Moberg, S. (2010). *Modeling and control of flexible manipulators*. Ph.D. thesis, Linköping University Electronic Press.
- Östring, M. (2002). Identification, diagnosis, and control of a flexible robot arm.
- Pham, C. and Gautier, M. (1991). Essential parameters of robots. In *Proceedings of the 30th IEEE Conference on Decision and Control*, 2769–2774.
- Schreiber, G., Stemmer, A., and Bischoff, R. (2010). The fast research interface for the kuka lightweight robot. In *IEEE workshop on innovative robot control architectures for demanding (Research) applications how to modify and enhance commercial controllers (ICRA)*, 15–21.
- Young, P.C. (2011). *Recursive estimation and time-series analysis: An introduction for the student and practitioner*. Springer Science & Business Media.