Ellipsoidal Estimation of Parallel Robot Dynamic Parameters

Philippe Poignet, Nacim Ramdani, Oscar Andrès Vivas

Proceedings of the 2003 IEEE/RSJ
Intl. Conference on Intelligent Robots and Systems
Las Vegas, Nevada · October 2003

ELLIPSOIDAL ESTIMATION OF PARALLEL ROBOT DYNAMIC PARAMETERS

Philippe Poignet[1], Nacim Ramdani[2], Oscar Andrès Vivas[1]

[1] *Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier (LIRMM), UMR CNRS 5506, University Montpellier 2*
161 Rue Ada, 34392 Montpellier cedex 5, France.
<vivas, poignet>@lirmm.fr

[2] *Centre d'Etude et de Recherche en Thermique, Energétique et Systèmes, Université Paris XII- Val de Marne, ave G. de Gaulle, 94000 Créteil.*
ramdani@univ-paris12.fr

Abstract - This paper presents the application of an ellipsoidal method for robust dynamic identification of parallel robots. The robot is modelled with classical Lagrange equation which leads to an inverse dynamic model linear with respect to the parameters. Assuming the error additive on input (motor torque), the problem is expressed in a bounded error context. The ellipsoidal method is applied in a factorised form in order to guarantee numerical stability. Experimental results are exhibited for a fully parallel robot with 4 degrees of freedom.

I. INTRODUCTION

Since recently robust estimation algorithms for robot manipulators identification are extensively investigated. In [1], they formulated an approach based on the maximum-likelihood parameter estimation, but in practical case, considered additive noise, which leads to the classical weighted least square (WLS) estimation. In [2], the authors compared WLS estimation to extended Kalman filtering. Within a statistical framework, the maximum likelihood estimator makes it possible to derive confidence intervals for the identified parameters. However, such techniques suffer some major weaknesses as it is difficult to check that the assumed statistical assumptions are verified and even to evaluate the fact that they are not. In addition, the models used often encompass significant structural errors that cannot be accounted for by random variables. In fact, the problem may be expressed in a bounded error context. Therefore, robust estimation can be performed through bounded error methods [3], [4].

In this paper, we focus on the implementation of ellipsoidal methods for robust and guaranteed parameter identification in a bounded error context. To ensure numerical stability when used with experimental data, the algorithm presented in [5] requires the use of a factorized form as proposed in [6]. Experimental results are exhibited for a 4-dof parallel mechanism – the H4 robot – [7], [8]. Fig. 1 shows a photography of the H4 parallel robot. This machine is based on 4 independent active chains between the base and the nacelle; each chain is actuated by a brushless direct drive motor fixed on the base and equipped with an incremental position encoder.

Fig. 1 H4 robot

Thanks to its design, the mechanism is able to provide high performances. However in order to achieve high speed and acceleration for pick-and-place applications or precise motion in machining tasks, advanced model based robust controllers are often required to increase the performances of the robot, which justify these works on guaranteed estimation.

The paper is organized as follows : Section II is dedicated to the geometric, kinematics and dynamic modelling of the H4 robot. Section III details the ellipsoidal technique and the factorization. Section IV exhibits major experimental results on a fully parallel robot. Finally, conclusions are given in section V.

II. MODELLING

A. Geometric and kinematics modelling

The Jacobian matrix and the forward geometric model are required to compute the dynamic model (see section II.B) [9]. Therefore we briefly present the way of computing the different relationship necessary to obtain these model and matrix. The design parameters of the robot are described in Fig. 2 where the following parameters have been chosen:

$\alpha_1 = 0;\ \alpha_2 = \pi;\ \alpha_3 = 3\pi/2;\ \alpha_4 = 3\pi/2$
$\boldsymbol{u}_1 = \boldsymbol{u}_y;\ \boldsymbol{u}_2 = -\boldsymbol{u}_y;\ \boldsymbol{u}_3 = \boldsymbol{u}_x;\ \boldsymbol{u}_4 = \boldsymbol{u}_x$

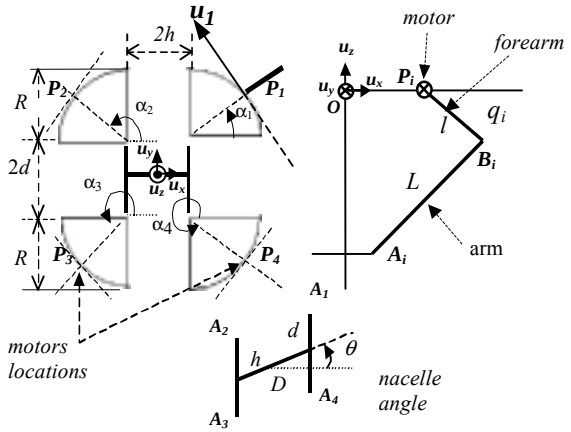

Fig. 2 Design parameters

The angles α_i describe the position of the four motors, L is the length of arms, l is the length of the forearms, θ the nacelle's angle, and d and h are the half lengths of the "H" forming the nacelle. $\boldsymbol{O}$ is the origin of the base frame and D is the origin of the nacelle frame. R gives the motor's position. The $\boldsymbol{A_iB_i}$ segments represent the arms of the robot and $\boldsymbol{P_iB_i}$ the forearm segments. The joint positions are represented by q_i.

To obtain the geometric model, we need to express the different points of the mechanical system with respect to the origin $\boldsymbol{O}$. The origin is fixed in the middle of the nacelle with the coordinates (x, y, z). In the Cartesian space, the end effector position is given by (x, y, z, θ).

$$\mathbf{OD} = \begin{bmatrix} x & y & z \end{bmatrix}^T \tag{1}$$

The vector that joins the absolute origin O and all of the forearms to the nacelle is:

$$\boldsymbol{OA_i} = \boldsymbol{OD} + \boldsymbol{DA_i} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \boldsymbol{DA_i} \tag{2}$$

The $\boldsymbol{DA_i}$ segments can be expressed as:

$$\boldsymbol{DA_1} = \begin{bmatrix} h\cos\theta \\ h\sin\theta + d \\ 0 \end{bmatrix}; \boldsymbol{DA_2} = \begin{bmatrix} -h\cos\theta \\ -h\sin\theta + d \\ 0 \end{bmatrix} \tag{3}$$

$$\boldsymbol{DA_3} = \begin{bmatrix} -h\cos\theta \\ -h\sin\theta - d \\ 0 \end{bmatrix}; \boldsymbol{DA_4} = \begin{bmatrix} h\cos\theta \\ h\sin\theta - d \\ 0 \end{bmatrix} \tag{4}$$

Moreover, the vector that links the absolute origin and all of the arms to the forearms is:

$$\boldsymbol{OB_i} = \boldsymbol{OP_i} + \boldsymbol{P_iB_i} \tag{5}$$

with:

$$\boldsymbol{P_iB_i} = \begin{bmatrix} l\cos q_i \cos\alpha_i \\ l\cos q_i \sin\alpha_i \\ -l\sin q_i \end{bmatrix} \tag{6}$$

and actuator locations are:

$$\boldsymbol{OP_1} = \begin{bmatrix} h + R\cos\alpha_1 \\ d + R\sin\alpha_1 \\ 0 \end{bmatrix}; \boldsymbol{OP_2} = \begin{bmatrix} -h + R\cos\alpha_2 \\ d + R\sin\alpha_2 \\ 0 \end{bmatrix} \tag{7}$$

$$\boldsymbol{OP_3} = \begin{bmatrix} -h + R\cos\alpha_3 \\ -d + R\sin\alpha_3 \\ 0 \end{bmatrix}; \boldsymbol{OP_4} = \begin{bmatrix} h + R\cos\alpha_4 \\ -d + R\sin\alpha_4 \\ 0 \end{bmatrix} \tag{8}$$

Finally, arms coordinates are given by:

$$\mathbf{A_iB_i} = \mathbf{A_iO} + \mathbf{OB_i} \tag{9}$$

The analytical forward position relationship is difficult to compute. Up to now, the simplest model we've got is a 8th degree polynomial equation. The forward model is then computed iteratively using the classical formula:

$$\mathbf{x}_{n+1} = \mathbf{x}_n + \mathbf{J}(\mathbf{x}_n, \mathbf{q}_n)\,[\mathbf{q} - \mathbf{q}_n] \tag{10}$$

Where $\mathbf{q}$ is the convergence point and $\mathbf{J}$ is the robot Jacobian matrix. If the mechanism is not in a singular configuration, this expression is derived as follows [7], [8]:

$$\mathbf{J} = \mathbf{J}_x^{-1}\,\mathbf{J}_q \tag{11}$$

Where:

$$\boldsymbol{J_x} = \begin{bmatrix} A_1B_{1x} & A_1B_{1y} & A_1B_{1z} & (\mathbf{DC}_1 \mathrm{x} A_1B_1)_z \\ A_2B_{2x} & A_2B_{2y} & A_2B_{2z} & (\mathbf{DC}_2 \mathrm{x} A_2B_2)_z \\ A_3B_{3x} & A_3B_{3y} & A_3B_{3z} & (\mathbf{DC}_3 \mathrm{x} A_3B_3)_z \\ A_4B_{4x} & A_4B_{4y} & A_4B_{4z} & (\mathbf{DC}_4 \mathrm{x} A_4B_4)_z \end{bmatrix} \tag{12}$$

$$\mathbf{J_q} = diag((\mathbf{P_iB_i} \times \mathbf{A_iB_i}).u_{mi}) \quad , \quad i = 1,\ldots 4 \tag{13}$$

$\boldsymbol{DC_i}$ is the distance between the centre of the nacelle and the centre of the half lengths of the "H" that forms the nacelle.

B. Dynamic modelling

In first approximation, the dynamic model is computed by considering physical dynamics. Indeed, the drive torques are mainly used to move the motor inertia, the fore-arms and the arms and the nacelle equipped with a machining

tool. Because of the design, the fore-arm inertia can be considered as a part of the motor inertia and the arm (manufacturing in carbon materials) effects are neglected [7], [8]. A simple friction model is added considering viscous and Coulomb friction.

If $\mathbf{\Gamma_{mot}}$ is the (4x1) actuator torque vector, the basic equation of dynamics can be written as :

$$\Gamma_{mot} = \mathbf{I}_{mot}\ddot{\mathbf{q}} + \mathbf{J}^T\mathbf{M}(\ddot{\mathbf{x}} - G) + \mathbf{F}_v\dot{\mathbf{q}} + \mathbf{F}_s sign(\dot{\mathbf{q}}) \quad (14)$$

where $\mathbf{I_{mot}}$ represents the motor's inertia matrix including the forearm's inertia, $\mathbf{M}$ a matrix containing the mass of the nacelle and its inertia, $\dot{\mathbf{q}}$ is the (4x1) joint velocity vector, $\ddot{\mathbf{q}}$ is the (4x1) joint acceleration vector , $\ddot{\mathbf{x}}$ is the (4x1) vector of cartesian accelerations $\left[\ddot{x} \quad \ddot{y} \quad \ddot{z} \quad \ddot{\theta}\right]^T$, and G the gravity constant. Thanks to the design, the forearm's inertia is taken into account in the motor's inertia. $\mathbf{F_v}$ are the viscous friction coefficients and $\mathbf{F_c}$ are the Coulomb friction.

With:

$$\mathbf{I}_{mot} = \begin{bmatrix} I_{mot1} & 0 & 0 & 0 \\ 0 & I_{mot2} & 0 & 0 \\ 0 & 0 & I_{mot3} & 0 \\ 0 & 0 & 0 & I_{mot4} \end{bmatrix} \quad (15)$$

$$\mathbf{M}_{mot} = \begin{bmatrix} M_{nac} & 0 & 0 & 0 \\ 0 & M_{nac} & 0 & 0 \\ 0 & 0 & M_{nac} & 0 \\ 0 & 0 & 0 & I_{bc} \end{bmatrix} \quad (16)$$

The dynamic equation can be rewritten in a relation linear to the dynamic parameters. By introducing $\mathbf{J}^{\mathrm{T}} = \left[\mathbf{J}_{43} \quad \mathbf{J}_4\right]$, it follows:

$$\mathbf{\Gamma}_{mot} = \begin{bmatrix} \ddot{\mathbf{q}} & \mathbf{J}_{43}\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} - G \end{bmatrix} & \mathbf{J}_4\ddot{\theta} & \dot{\mathbf{q}} & sign(\dot{\mathbf{q}}) \end{bmatrix}\boldsymbol{\theta} \quad (17)$$

where $\boldsymbol{\theta}$ is the vector of parameters:

$$\boldsymbol{\theta} = [I_{mot1} \quad I_{mot2} \quad I_{mot3} \quad I_{mot4} \quad M_{nac} \quad I_{bc} \quad F_{v1} \quad F_{v2} \quad F_{v3} \quad F_{v4} \quad F_{s1} \quad F_{s2} \quad F_{s3} \quad F_{s4}]^T \quad (18)$$

Only the torque input $\mathbf{\Gamma_{mot}}$ and motor position $\mathbf{q}$ are directly measured. As acceleration measurement $\ddot{\mathbf{x}}$ is not available, $\ddot{\mathbf{x}}$ is evaluated by:

$$\ddot{\mathbf{x}} = \mathbf{J}\ddot{\mathbf{q}} + \dot{\mathbf{J}}\dot{\mathbf{q}} \quad (19)$$

where $\mathbf{J}$ depends on $\mathbf{x}$ and $\mathbf{q}$, is computed using a central difference algorithm.

III. ELLIPSOIDAL METHOD

When the statistical properties of the random variable used to model the actual disturbances acting on model inputs or outputs remain unattainable, it is still possible to compute values for the bounds between the output of a model $y_k^m(\cdot)$ and some actual measurements y_k. Indeed, the sensors used for data measurements are frequently characterized with a prior maximum measurement error. Under the hypothesis of additive noise, actual model output can be related to actual data as follows:

$$y_k = y_k^m\left(\boldsymbol{\theta}^*\right) + \varepsilon_k^* \ , \ k = 1 \ldots N \quad (20)$$

where N is the number of observations and $\boldsymbol{\theta}^*$ is the unknown *true* parameter vector to be identified and $\left\{\varepsilon^*\right\}$ an output error sequence assumed to be stationary, uncorrelated and bounded but otherwise unknown. The error sequence thus satisfies the following inequality:

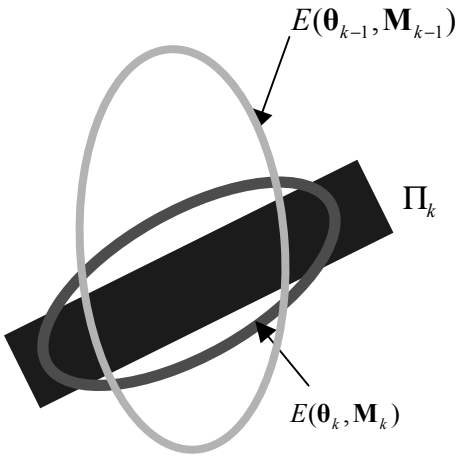

Fig. 3 OBE principle

$$\forall k = 1 \ldots N, \ -1 \le \varepsilon_k^* \le 1 \quad (21)$$

This writing is known as the standard form with a normalized error, and it is always possible to transform the case where the upper and lower prior error bounds are different to such a form.

A parameter vector $\boldsymbol{\theta}$ is said acceptable, if and only if the output error is enclosed in the prior bounds. Consequently, the issue of the bounded-error set estimation is to compute the set, known as *the posterior feasible set*, defined as:

$$S = \left\{\boldsymbol{\theta} \in Q \middle| \ \forall k = 1 \ldots N, \ -1 \le y_k - y_k^m\left(\boldsymbol{\theta}\right) \le 1\right\} \quad (22)$$

where the prior parameter search space $Q \subseteq \mathbb{R}^p$.

When the model is linear in parameter, which is the case

of the robot inverse dynamic model, it is written as:

$$y_{m,k} = \mathbf{d}_k^T \boldsymbol{\theta} \qquad (23)$$

The parameter set compatible with the datum at observation k is a strip Π_k defined by:

$$\Pi_k = \left\{ \boldsymbol{\theta} \in \mathbb{R}^p \;\middle|\; \left\| \mathbf{y}_k - \mathbf{d}_k^T \boldsymbol{\theta} \right\|^2 \le 1 \right\} \qquad (24)$$

The posterior feasible set is the intersection of a prior search space $\mathbf{Q}$ and N strips $\boldsymbol{\Pi}_k$:

$$S = \left\{ \boldsymbol{\theta} \in Q \;\middle|\; \forall k = 1 \ldots N,\; -1 \le y_k - \mathbf{d}_k^T \boldsymbol{\theta} \le 1 \right\} \qquad (25)$$

It is a convex polyhedron, which is clearly complex when N is large. In particular, it becomes computationally expensive and demands large memory resources. Therefore, several ways have been investigated in order to approach this polyhedron with simple-shaped forms, such as ellipsoids or parallelotopes [3]-[5], [10]. In the next section, a recursive algorithm will be described for computing the smallest ellipsoid, which outer-bounds the posterior feasible set.

A. The standard approach

The Optimal Bounding Ellipsoid (OBE) algorithms are a class of bounded-error methods which aim at superscribing the exact polytope of feasible parameters by an hyperellipsoid. One of the claimed advantages of the ellipsoids is that they can be concisely described by a vector specifying the centre of the ellipsoid and a positive-definite matrix which specifies its size and orientation.
Two families of algorithms were developed in order to determine the optimal outer-bounding ellipsoid, one in estimation theory, known as Optimal Bounding Ellipsoid algorithm (OBE) and the other in linear programming, known as Ellipsoid with Parallel Cuts algorithms (EPC) [3], [4]. The derived algorithms are structured to be computationally efficient. In the following, the OBE algorithm will be used and its main steps of the OBE algorithm are as follows. After processing the k-1 first observations, the ellipsoid $E(\hat{\boldsymbol{\theta}}_{k-1}, \mathbf{M}_{k-1})$ outer-bounding the posterior feasible set consistent with the observations is characterised by:

$$E(\hat{\boldsymbol{\theta}}_{k-1}, \mathbf{M}_{k-1}) = \left\{ \boldsymbol{\theta} \in \mathbb{R}^p \;\middle|\; \left(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}_{k-1}\right)^T \mathbf{M}_{k-1} \left(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}_{k-1}\right) < 1 \right\} \qquad (26)$$

where $\hat{\boldsymbol{\theta}}_{k-1}$ is the centre of the ellipsoid and $\mathbf{M}_{k-1}$ is an information matrix which the inverse defines the shape and the orientation of the ellipsoid. Given the new observation at time k, the updated ellipsoid at time k, which outer-bounds the intersection of the ellipsoid estimated up to the observation time k-1 and the strip defined by the new observation datum, Π_k, satisfies the following relationship (see also fig. 3):

$$E(\hat{\boldsymbol{\theta}}_k, \mathbf{M}_k) \supseteq E(\hat{\boldsymbol{\theta}}_{k-1}, \mathbf{M}_{k-1}) \cap \Pi_k \qquad (27)$$

where Π_k is given by equation (24).
This equation can be equivalently written as the following inequality: (28)

$$\boldsymbol{\theta} \in E\left(\hat{\boldsymbol{\theta}}_{k-1}, \mathbf{M}_{k-1}\right) \Rightarrow \forall \alpha \in]0,1[,$$
$$\alpha \left(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}_{k-1}\right)^T \mathbf{M}_{k-1} \left(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}_{k-1}\right) + (1-\alpha) \left\| y_k - \mathbf{d}_k \boldsymbol{\theta} \right\|^2 \le 1$$

which defines a family of ellipsoids parameterised with α parameter. The value of the latter is chosen in order to minimize the volume of the new ellipsoid $E(\hat{\boldsymbol{\theta}}_k, \mathbf{P}_k)$:

$$\hat{\alpha} = \arg\min \left(\log \det \left(\mathbf{M}_k^{-1} \right) \right) \qquad (29)$$

The interested reader will find in [3], [5] the whole details of the computations and an explicit solution for $\hat{\alpha}$.

The on-line algorithm for computing the best ellipsoid outer-bounding the posterior parameter set is as follows:

$$\begin{cases} \mathbf{N} = \hat{\alpha} \mathbf{M}_{k-1} + (1-\hat{\alpha}) \mathbf{d}_k \mathbf{d}_k^T \\ \hat{\boldsymbol{\theta}}_k = \mathbf{N}^{-1} \left(\hat{\alpha} \mathbf{M}_{k-1} \hat{\boldsymbol{\theta}}_{k-1} + (1-\hat{\alpha}) \mathbf{d}_k \mathbf{y}_k \right) \\ \hat{\delta} = \hat{\alpha} \hat{\boldsymbol{\theta}}_{k-1}^T \mathbf{M}_{k-1} \hat{\boldsymbol{\theta}}_{k-1} + (1-\hat{\alpha}) y_k^2 - \hat{\boldsymbol{\theta}}_k^T \mathbf{N} \hat{\boldsymbol{\theta}}_k \\ \mathbf{M}_k = \mathbf{N} \Big/ \left(1 - \hat{\delta}\right) \end{cases} \qquad (30)$$

The on-line ellipsoidal procedure given in (30) is potentially numerically instable because they use the normal equations of least square [6]. This instability is due to the fact that the information matrix $\mathbf{M}$ may become non definite positive.

B. The factorized form

As an alternate solution for an efficient numerical implementation, Lesecq and Barraud [6] propose using a factorized form of (30).

The main idea is to regard the determination of the ellipsoid $E(\hat{\boldsymbol{\theta}}_k, \mathbf{M}_k)$ as an optimisation problem:

$$\hat{\boldsymbol{\theta}} = \arg\min \left[f(\boldsymbol{\theta}) \right] \qquad (31)$$

where the objective function is given by:

$$f(\boldsymbol{\theta}) = (1-\hat{\alpha}) \left\| y_k - \mathbf{d}_k^T \boldsymbol{\theta} \right\|^2 + \hat{\alpha} \left(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}_{k-1}\right)^T \mathbf{M}_{k-1} \left(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}_{k-1}\right) \qquad (32)$$

Introducing a Cholesky factorisation of $\hat{\alpha}\mathbf{M}$ and the following vectors

$$\begin{cases} \tilde{\mathbf{X}} \; s.t. \; \mathbf{M}_k = \tilde{\mathbf{X}}_k^T \tilde{\mathbf{X}}_k \\ \mathbf{X}_k = \sqrt{\hat{\alpha}}\tilde{\mathbf{X}}_k \\ v = \sqrt{1-\hat{\alpha}}\mathbf{d}_k \\ w = \sqrt{1-\hat{\alpha}}\, y_k \end{cases} \quad (33)$$

the functional (32) can be equivalently written as:

$$f(\boldsymbol{\theta}) = \left\| w - v^T \boldsymbol{\theta} \right\|^2 = \left\| \begin{bmatrix} \mathbf{X}_k \\ v^T \end{bmatrix} \boldsymbol{\theta} - \begin{bmatrix} \mathbf{X}_k \hat{\boldsymbol{\theta}}_k \\ w \end{bmatrix} \right\|^2 \quad (34)$$

Equation (34) has the form of a classical least square problem, the resolution of which can be done through orthogonal factorisation [11].

The new algorithm in a factorized form can then be derived for the recursive updating of the outer-bounding ellipsoid. In [6], they demonstrate that computing the algorithm (30) is completely equivalent to:

$$\begin{cases} \text{build} \quad \mathbf{P} = \begin{bmatrix} \mathbf{X}_{k-1} & \mathbf{X}_{k-1}\hat{\boldsymbol{\theta}}_{k-1} \\ v^T & w \end{bmatrix} \\ \text{compute a triangular form of } \mathbf{P}, \\ \qquad \mathbf{QP} = \begin{bmatrix} \mathbf{U} & \mathbf{u} \\ 0 & \tau \end{bmatrix} \\ \text{compute } \hat{\boldsymbol{\theta}}_k \text{ by solving the triangular system,} \\ \qquad \mathbf{U}\hat{\boldsymbol{\theta}}_k = \mathbf{u} \\ \text{compute } \tilde{\mathbf{X}}_{k-1} = \mathbf{U} / \sqrt{1-\tau^2} \end{cases} \quad (35)$$

Square Root Information Ellipsoid Filter algorithm: The factorized form algorithm is derived from (35) and is written as follows :

- Initialize : $\tilde{X}_0, z_0 = \tilde{X}_0 \hat{\theta}_0$
- Recurse over time :
  - Computation of $\hat{\alpha}$
  - Factorisation QR :

$$Q \begin{bmatrix} \sqrt{\hat{\alpha}} \begin{bmatrix} \tilde{\mathbf{X}}_{k-1} & z_{k-1} \end{bmatrix} \\ \sqrt{1-\hat{\alpha}} \begin{bmatrix} \mathbf{d}_k^T & y_k \end{bmatrix} \end{bmatrix} = \begin{bmatrix} \mathbf{U} & \mathbf{u} \\ 0 & \tau \end{bmatrix}$$

  - Solve $\mathbf{U}\hat{\boldsymbol{\theta}}_k = \mathbf{u}$
  - $\begin{bmatrix} \tilde{\mathbf{X}}_k & z_k \end{bmatrix} = \frac{1}{\sqrt{1-\tau^2}} \begin{bmatrix} \mathbf{U} & \mathbf{u} \end{bmatrix}$

The algorithm ensures numerical stability of the computation and makes the latter simpler as the determination of the centre and the information matrix are performed independently.

IV. EXPERIMENTAL RESULTS

A. Experimental data

Joint position $\mathbf{q}$ and the current reference $\mathbf{V_T}$ (the control input) are collected at a 1000Hz sample rate while the robot is tracking exciting trajectories containing both slow (for friction) and high dynamics (for inertia). These trajectories ensure a good condition number. The identification is performed by using a closed-loop joint PI control. The torques are computed using a linear relationship between torque $\boldsymbol{\Gamma}_{\mathbf{mot}}$ and voltage $\mathbf{V_T}$ where G_T is the amplifier gain:

$$\boldsymbol{\Gamma}_{mot} = G_T V_T \quad (36)$$

Joint velocities and accelerations for computing the regressor are estimated by a band pass filtering of the position. The band pass filtering is obtained by the product of a low pass filter in both the forward and the reverse direction (Butterworth) and a derivative filter obtained by a central difference algorithm, without phase shift. The cut-off frequency of the low pass filter should be chosen to avoid any distortion of magnitude on the filtered signals in the range [0 ω_{dyn}] where ω_{dyn} is the bandwidth of the position closed loop. A parallel filtering is implemented to reject the high frequency ripples of the measured motor torques. Practical aspects of the derivative estimation and data filtering are completely detailed in [2].

B. Estimated parameters

The prior bounds on motor torques are tuned by taking into account prior information on motors. They were chosen prior to the computation as 5% of measurement range of the torques (±15Nm). Then they were increased to 15% in order to ensure that the number of outliers remains negligible. This increase can be explained by the fact that the estimated value for the Coulomb friction parameter is around 1 Nm.

Table I contains the estimated centre $\hat{\theta}$ of the outer-bounding ellipsoid. Prior values for motor inertia and nacelle mass and inertia are known by design. The components of the centre of the estimated ellipsoid are close to prior values. Motor inertia are larger than the prior ones because they actually encompass the inertia of the fore-arm, which were neglected.

Table I. Estimated parameters (SI Units)

Parameter	Estimated ellipsoid centre $\hat{\theta}$	*A priori* values
I_{mot1}	0.0199	0.012
I_{mot2}	0.0133	0.012
I_{mot3}	0.0189	0.012
I_{mot4}	0.0250	0.012
M_{nac}	0.7914	1.0
I_{bc}	0.0005	0.0008
F_{v1}	0.3093	/

F_{v2}	0.2166	/
F_{v3}	0.1235	/
F_{v4}	0.0752	/
F_{c1}	0.8126	/
F_{c2}	0.9390	/
F_{c3}	0.5360	/
F_{c4}	0.9868	/

The uncertainty volume is given by $\det\left(\hat{\mathbf{M}}^{-1}\right)$=1.63 10^{-21} where $\hat{\mathbf{M}}$ is the ellipsoid estimated Information matrix after that the whole data have been introduced. The number of outliers is less than 0.03%.

C. Cross-validation

The cross-validation is performed with trajectories different from the ones used for the estimation step. Fig. 4 shows the estimated torque bounds computed by the following formulae:

$$y_k^{\pm} = d_k^T \hat{\theta} \pm \sqrt{d_k^T \hat{\mathbf{M}}^{-1} d_k} \qquad (37)$$

which not only accounts for the uncertainty in the estimated parameters, but also for the dependence between them. This dependence appears mathematically through the non-diagonal elements of the estimated covariance matrix $\hat{\mathbf{M}}^{-1}$ which are significant in this study.

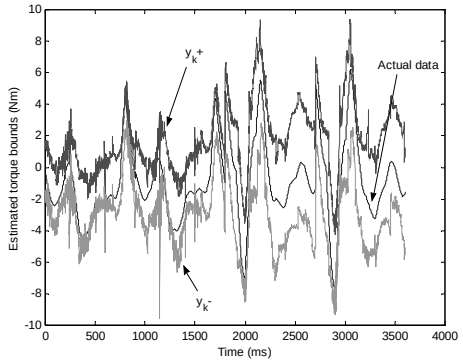

Fig. 4 Cross-validation
Estimated torque bounds / actual data

V. CONCLUSION

This paper exhibits relevant results for robust dynamic identification of parallel robots with an ellipsoidal method. To ensure numerical stability, the factorized form has been used. The estimated ellipsoid centre is close to prior values. Estimated torque bounds issued from the cross-validation demonstrate good performances of the ellipsoidal method. The estimation may be compared to the weighted least square experimental results given in [12]. The quality of estimation have been checked through model based control scheme. Experimental results are detailed in [13]. Further works will concern the choice of other criteria (trace, volume or mixed techniques) in order to decrease the size of the outer-bounding ellipsoid and the estimated torque bounds.

VI. REFERENCES

[1] J. Swevers, C. Ganseman, D.B. Tükel, J.D. Schutter, and H. Van Brussel,"Optimal robot excitation and Identification, *IEEE Transactions on Robotics and Automation,* vol. 13, n°5, pp.730-740, 1997.

[2] Ph. Poignet, M. Gautier,"Extended Kalman filtering and weighted least squares dynamic identification of robots", *Control Engineering Practice,* vol. 9, n°12, pp. 1361-1372, 2001.

[3] E. Walter and L. Pronzato,. *Identification of parametric models from experimental data*, Springer, London, 1997.

[4] D.G. Maksarov, J.P. Norton, "Computationally efficient algorithms for state estimation with ellipsoidal approximations", *Int. J. Adapt. Control Signal Process.*, vol. 16, n°6, pp.411-434, 2002.

[5] C. Durieu, E. Walter, B. Polyak, "Multi-Input Multi-Output Ellipsoidal State Bounding", *Journal of Optimization Theory and Applications*, vol. 111, n°2, pp.273-303, 2001.

[6] S. Lesecq and A. Barraud, « Une approche factorisée plus simple et numériquement stable pour l'estimation ensembliste ellipsoïdale », Journal Européen des Systèmes Automatisés, vol. 36, n°4, pp. 505-518, 2002.

[7] O. Company and F. Pierrot, "A new 3T-1R parallel robot", in *Proceedings of ICAR'99,* Tokyo, Japan, October 25-27, pp. 557-562.

[8] F. Pierrot, F. Marquet, O. Company and T. Gil, "H4 Parallel Robot: Modeling, Design and Preliminary Experiments", in *Proceedings of the 2001 IEEE International Conference on Robotics & Automation,* Seoul, Korea, May 21-26.

[9] W. Khalil and E. Dombre *Modeling, Identification and Control of Robots.* Hermes Penton Science, London, 2002.

[10] A. Vicino, G. Zappa, "Sequential Approximation of feasible parameter sets for identification with set membership uncertainty", *IEEE Trans. on Automatic Control*, 1996, vol. 41, n°6, pp. 774-785.

[11] C. Lawson and C. Hanson, *Solving Least Squares Problems*, Prentice-Hall, 1974.

[12] O.A. Vivas, Ph. Poignet, F. Marquet, F. Pierrot, M. Gautier, "Experimental dynamic identification of a fully parallel robot", in *Proceedings of the 2003 IEEE International Conference on Robotics & Automation,* Taïpei, Taïwan, September 14-19.

[13] O.A. Vivas, Ph. Poignet, "Model based predictive control of a fully parallel robot", in *Proceedings of the 7th IFAC Symposium on Robot Control,* Wroclaw, Poland, September 1-3, 2003.