2014 IEEE International Conference on Robotics & Automation (ICRA)
Hong Kong Convention and Exhibition Center
May 31 - June 7, 2014. Hong Kong, China

Robot-Dynamic Calibration Improvement by Local Identification

Nicola Pedrocchi[1], Enrico Villagrossi[1,2], *Student Member, IEEE*, Federico Vicentini[1], *Member, IEEE* and Lorenzo Molinari Tosatti[1]

***Abstract*— Notwithstanding the research on dynamic modelling of Industrial Robots (IRs hereafter) covers the last three decades, improvements are necessary to enable IRs adoption in technological tasks where high dynamics or interaction with environment is needed, *e.g.* deburring, milling, laser cutting *etc.* Indeed, this class of applications displays even more the necessity of high-accuracy tracking especially in workspace sub-regions, while common IR dynamic calibration methods often span the workspace at large (in term of positions and high velocities) resulting in an averagely fitting models. Open issues are therefore on the applicability/scalability of standard methods in workspace sub-regions and on the metrics used for the calibration performance evaluation. The paper proposes an algorithm designed to high-accuracy local dynamic identification, comparing it with the results achievable by a common IRs dynamic calibration method and by the same method scaled to a workspace sub-region. In addition, unlike from standard, the here reported experimental comparison is made by evaluating the torque prediction error for IRs robot moving along path programmed by standard/commercial IR motion planner and not along path belonging to the same template-class of trajectory used in identification phase.**

I. Introduction

It is long [1], [2], and generally accepted that the dynamic calibration of industrial robots (IRs) is of utmost importance in increasing the predictability and accuracy of model-based control strategies. Since early works [3], [4], many researchers have investigated some methodologies all involving a linear reduction of the rigid-body model into a minimum set of lumped dynamic parameters to be estimated [5], *e.g.* a complete-observable linear map from joint position, velocity and acceleration to motor torques. Such class of methods focuses on the optimization of trajectories that homogeneously excite the parameters of the model in order to attain a robust, over-constrained, well-conditioned linear system [6], [5], [7], [8], [9], [10]. The accuracy in torque prediction relies on the conditioning properties of the resulting kinematics function (regressor) that maps the to-be-estimated parameters into torques. Nevertheless, many IR applications require trajectories interpolated from a set of lines and circles, performed at constant regime velocity, in other words a class of trajectories fairly different from those commonly used in standard optimization procedures. This aspect should be critical because the maximum prediction power attained from common dynamic calibration algorithms is often for a class of trajectories rarely used in actual industrial tasks. Furthermore, considering how IRs are nowadays used, two aspects are important: (i) many IR tasks need a high dynamic accuracy only in *locally* constrained workspaces; (ii) IRs adoption in technological applications is limited and more accurate *local* dynamic models should improve the tracking accuracy also when external forces fiercely act on the tools. High-accuracy in torques prediction for bounded movements is often attained by Iterative Learning Control (ILC) techniques [11] corresponding to an implicit dynamics calibration, *i.e.* without the need of any model. Nevertheless, ILC algorithms prevent the possibility to extend (extrapolate) results to non-trained trajectories, and are rarely used in tasks where robots have to interact with the environment [12]. Higher accuracy can also been obtained with advanced methods integrating the elastic-joint model [13], [14], [15], [16], which we believe are important especially for the next generation of IRs, compliant platforms and high-dynamic technological processes done by IRs. Nonetheless, the calibration of elastic models is still extremely difficult and adaptive control strategies are often mandatory to compensate any inaccuracy in the estimation [17], [18].

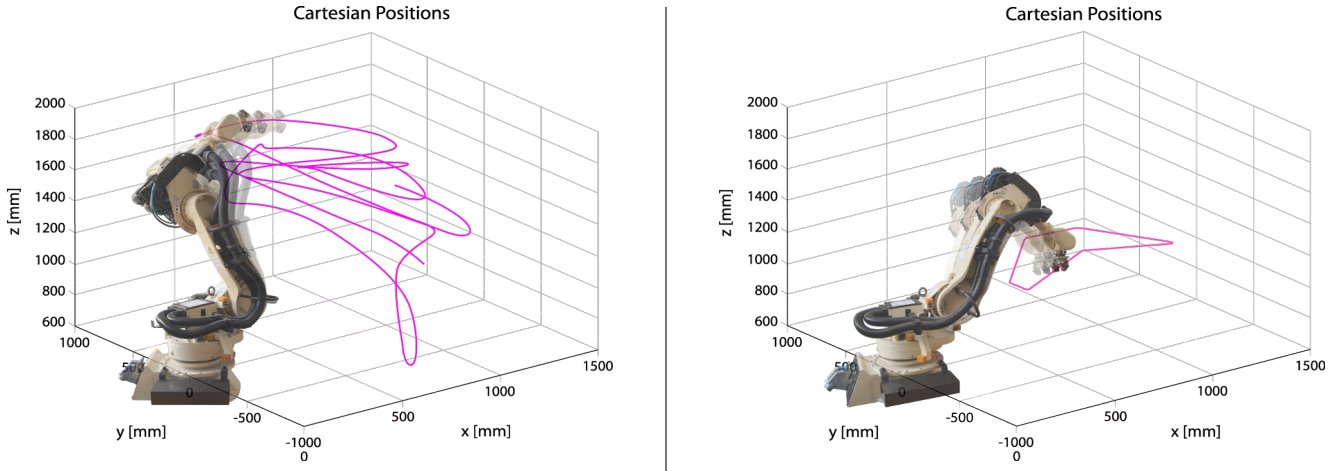
A viable solution, poorly investigated in literature, would be a down-scaling of standard global methods in workspace sub-regions, *i.e.* applying the general procedure in a limited volume. It would be however arguable that the *local* validity of a calibration procedure in a bounded sub-region could instead benefit from a dedicated excitatory pattern (*e.g.* in the likely event of the manipulator changing very little its configuration). It is, in fact, not guaranteed that the standard multi-body rigid-model with minimal reduced parameters are completely observable also in constrained sub-workspace, and no assumption can be *a priori* made on the achievable quality of the excitation. Such potential lack of scalability could be therefore amended by local methods.

In a previous work [20], we proposed a *local* dynamics parameters identification method to improve the motor-torque prediction accuracy. Such method notably employs at identification time a template-class of trajectories applied in most manufacturing tasks, *i.e.* general trajectories described by a set of discrete poses to be interpolated by the built-in IR motion planner on the basis of global user-tunable parameters (fly-by accuracy, velocity profiles, etc). Following [20], this work aims at investigating how to locally improve the accuracy of IR dynamic calibration, evaluating the different performances of three different local/global algorithms detailed in Section II applied along local/global/specialized tra-

This work is partially within FLEXICAST funded by FP7-NMP EC

[1]N. Pedrocchi, *et al.* are with the Institute of Industrial Technologies and Automation, National Research Council, via Bassini 15, 20133 Milan, Italy nicola.pedrocchi@itia.cnr.it

[2]E. Villagrossi is PhD Student with University of Brescia, Dep. of Mechanical and Industrial Engineering, via Branze 39, 25123 Brescia, Italy



(a) over the whole workspace [19] by sinusoidal trajectories

(b) over a workspace sub-region [20] by IR motion planner

Fig. 1: IRs Calibration Trajectories.

jectories. Additionally, an unified evaluation metrics has been introduced considering to the error between actual-measured and model-predicted torques along trajectories calculated by a commercial IR motion planner, such that the results would reflect the performances achievable in industrial applications.

Notation

$\mathbf{q} = \left[q^1, .., q^{dof}\right]^T$	Joint positions.
$\mathbf{q}_s = \dot{\mathbf{q}}_s, \ddot{\mathbf{q}}_s, \boldsymbol{\tau}_s$	Joint Positions Velocities, accelerations, torques at s-th sample time.
$\mathbf{Q} \equiv \{\mathbf{q}_1, .., \mathbf{q}_S\}$	Joint position time series of S different time samples.
$\{\mathbf{Q}, \dot{\mathbf{Q}}, \ddot{\mathbf{Q}}\}$	Trajectory.
$\tilde{(\cdot)}$, $\hat{(\cdot)}$, $(\cdot)^*$	Measured, estimated value and optimum estimation respectively.

II. Local and Global Calibration

Denoting $\boldsymbol{\Phi} = \boldsymbol{\Phi}(\mathbf{Q}, \dot{\mathbf{Q}}, \ddot{\mathbf{Q}})$ as the dynamics regressor [5], (*i.e.* the matrix that maps the set of reduced dynamic parameters into motor torques), the calibration consists procedurally on the identification of an optimal trajectory $\{\mathbf{Q}^\star, \dot{\mathbf{Q}}^\star, \ddot{\mathbf{Q}}^\star\}$ that minimizes an index extracted from the regressor [4], [7], [8]. The selection of the index is not unique and it reflects a *criteria* for minimizing the estimation error[1] [19].

The paper aims at displaying how (i) the design of excitatory movements for *local* dynamic calibration methods should improve motor-torque prediction accuracy, and how (ii) scaling/porting of standard procedure would not be the best modality. Hence three algorithms have been implemented and their accuracy have been measured. Algorithm A is the implementation of a well-known approach [19] and the IR dynamic model has been calibrated on the whole workspace; algorithm A_{local} is a scaling of algorithm A in a bounded workspace sub-region, and finally the algorithm B is an algorithm suggested by authors in [20] designed for locally IR dynamic model calibration.

[1]Dynamic calibration does not aim at estimating the full set of link figures (masses, inertias, friction coefficients) but to minimize the prediction error of the model w.r.t. the real behavior of the robot.

1) Algorithm A - calibration over the whole workspace using [19]: the template-class of trajectory used for the optimization is:

$$q^j = q_0^j + \Sigma_{k=1}^{W} a_k^j \sin\left(\omega_k^j t\right) \quad j = 1, \ldots, dof. \tag{1}$$

where q_0^j is the initial offset and W is a small integer. Collecting the free variables $\mathbf{a}_k = [a_k^1, ..., a_k^{dof}]^T$ and $\boldsymbol{\omega}_k = [\omega_k^1, ..., \omega_k^{dof}]^T$, the set of the decision variables of the optimization problem results $\{\mathbf{a}_1, \boldsymbol{\omega}_1, ..., \mathbf{a}_W, \boldsymbol{\omega}_W\}$ and proper constraints are to be imposed[2] coherently with the kinematics of the robot. The Cartesian path displayed by (1) is reported in Fig. 1a.

2) Algorithm B - calibration over a bounded sub-workspace using [20]: authors have suggested in [20], a paradigmatic template-class for the excitatory movements in sub-workspace. The optimization grounds on the definition of a workspace sub-region that bounds K different desired interpolated via-points at a quasi-constant velocity, $\{P_1, \ldots, P_k, V\}$ by the means of a standard IRs motion planner, MP hereafter, such that

$$\mathbf{q} = MP\left(P, \ldots, P_k, V, t\right). \tag{2}$$

Cartesian path displayed by this class of trajectory is reported in Fig. 1b.

3) Algorithm A_{local} - Algorithm A scaled over bounded sub-workspace: the template-class of trajectory used for the optimization is the same of algorithm A, but the exploration range for the parameters is bounded in a limited workspace, approximately the same of algorithm B.

On top of algorithms A, A_{local} and B, we impose the same (i) dynamic model, (ii) pseudo-inversion regressor procedure, (iii) optimization *criteria*, and (iv) similar minimum search procedure. Hence, a simpler comparison analysis is guaranteed even if each method should result under-performing with respect to an *ad-hoc* selection of (i-iv).

[2]As in [19], the constraints are $|a_k^j| < q_{max}^j/W$ and $|\omega_k^j| < \sqrt{\ddot{q}_{max}^j/q_{max}^j}$, with $j = 1, \ldots, dof$.

III. Models and Method

A. Dynamics modeling and estimation

Making use of rigid multi-bodies dynamics, the robot dynamics can be reduced [21] to:

$$\boldsymbol{\tau} = \boldsymbol{\phi}^0 \left(\ddot{\mathbf{q}}, \dot{\mathbf{q}}, \mathbf{q}\right) \boldsymbol{\pi}^0 \tag{3}$$

where $\boldsymbol{\pi}^0$ is a base set of dynamical parameters and matrix function $\boldsymbol{\phi}^0$ is a generalized accelerations. The base set $\boldsymbol{\pi}^0$ includes only combination of parameters that are observable along any excitatory trajectory that generates $\boldsymbol{\phi}^0$. The minimal size N_π of the base set $\boldsymbol{\pi}^0$ is demonstrated [5] to be 40 for the specific 6-dof anthropomorphic manipulator considered as test case in this work. In addition, other N_f coefficients of the friction model yield the compound parameters set $\boldsymbol{\pi}$. The selected friction model [22] provides the j-th joint friction torque function of three parameters, f_0^j, f_1^j, f_2^j, such that it requires $N_f = 3 \times dof$ additional parameters. Hence, friction torque is modelled as:

$$\tau_f^j = f_0^j \, sign(\dot{q}^j) + f_1^j \dot{q}^j + f_2^j \, sign(\dot{q}^j) \left(\dot{q}^j\right)^2. \tag{4}$$

B. Solving System and Regressor Pseudo-inversion

For trajectory $\{ \mathbf{Q}, \dot{\mathbf{Q}}, \ddot{\mathbf{Q}} \}$ of S-samples, (3) results:

$$\mathbf{T} \equiv \begin{bmatrix} \boldsymbol{\tau}_1 \\ \vdots \\ \boldsymbol{\tau}_S \end{bmatrix} = \begin{bmatrix} \boldsymbol{\phi}_1 \left(\ddot{\mathbf{q}}_1, \dot{\mathbf{q}}_1, \mathbf{q}_1\right) \\ \vdots \\ \boldsymbol{\phi}_S \left(\ddot{\mathbf{q}}_S, \dot{\mathbf{q}}_S, \mathbf{q}_S\right) \end{bmatrix} \boldsymbol{\pi} = \boldsymbol{\Phi}\boldsymbol{\pi}, \tag{5}$$

where $\boldsymbol{\Phi}$ is the regressor matrix. Actually, experimental sampling $\widetilde{\mathbf{T}}$, $\widetilde{\boldsymbol{\Phi}}$ includes also measurements noise $\boldsymbol{\nu}$:

$$\widetilde{\mathbf{T}} = \widetilde{\boldsymbol{\Phi}}\hat{\boldsymbol{\pi}} + \boldsymbol{\nu}, \quad \boldsymbol{\nu} \sim \mathcal{N}\left(0, \sigma_\nu\right). \tag{6}$$

Several techniques are known [23], [24] for the pseudo-inversion solution of (6) and the weighted least-squares technique [25] has been here implemented[3]. Denoting as $\boldsymbol{W}$ the weight matrix, the system is solved as

$$\hat{\boldsymbol{\pi}} = \left[(\widetilde{\boldsymbol{\Phi}}^T \boldsymbol{W} \widetilde{\boldsymbol{\Phi}})^{-1} \widetilde{\boldsymbol{\Phi}}^T \boldsymbol{W} \right] \widetilde{\mathbf{T}}. \tag{7}$$

Hence, the generation of an optimal excitation trajectory $\{ \mathbf{Q}^\star, \dot{\mathbf{Q}}^\star, \ddot{\mathbf{Q}}^\star \}$ able to provide the best regression conditions for (6) leads the optimum parameters estimation.

C. Optimization Criteria

The set of optimum parameters $\{\mathbf{a}_1^\star, \boldsymbol{\omega}_1^\star, ..., \mathbf{a}_W^\star, \boldsymbol{\omega}_W^\star\}_A$, $\{\mathbf{a}_1^\star, \boldsymbol{\omega}_1^\star, ..., \mathbf{a}_W^\star, \boldsymbol{\omega}_W^\star\}_{A_{local}}$, and $\{P_1^\star, \ldots, P_k^\star, V^\star\}$ are obtained from Genetic Algorithms (GA hereafter), and the population dimension and the generations strategies (combination of selection with elitism, recombination and mutation) have been imposed equal for the A, A_{local}, and B.

The selection of individuals is made on a fitness function, *i.e.* a scalar index, calculated on the regressor Φ scaled with respect to the axis-wise ratios of nominal-over-maximum $\overline{\tau} = \max_i(\tau_{nom}^i)$ in order to normalize contributions of different actuators. The regressor torque-normalization, $\mathbf{H}(\boldsymbol{\Phi})$, would remove biases due to the different motor contribution. Denoting as $\mathbf{d} = diag(\tau_{nom}^1/\overline{\tau} ... \tau_{nom}^6/\overline{\tau})$, it results:

$$\mathbf{H}\left(\boldsymbol{\Phi}\right) = diag\left(\mathbf{d}, \ldots, \mathbf{d}\right) \boldsymbol{\Phi}. \tag{8}$$

Finally, in this case D-optimal fitness was considered maximizing the determinant of a quadratic form associated with $\mathbf{H}(\boldsymbol{\Phi}^n)$ of each n-th individual trajectory[4], *i.e.*

$$f^n\left(\boldsymbol{\Phi}^n\right) = \log_{10} \| \det\left[\mathbf{H}^T\left(\boldsymbol{\Phi}^n\right) \mathbf{H}\left(\boldsymbol{\Phi}^n\right)\right] \|. \tag{9}$$

To make dimensionless $f^n\left(\boldsymbol{\Phi}^n\right)$ with respect to the number of samples, two strategies have been followed: for A and A_{local}, the execution time has been fixed equal to $2\pi / \min(\omega_k^j)$ $\forall j, k$, and path has been sampled with a fix number of points; in B the velocity is coded in the genes, thus the number of points has not been constrained.

TABLE I: GA results (120 generation). N denotes the trajectory samples and I the number of generations.

	I	N	$\log_{10} \| \det[\mathbf{H}^T\mathbf{H}] \|$
A	150	2000	182
A_{loc}	150	2000	160
B	150	2875	47.6

D. Experiment Design and Dependant Measure

Algorithms A, A_{local} and B have been implemented, and the three corresponding optimal trajectories have been calculated. Hence the robot has been asked to repeat 30 times each trajectory and $\hat{\boldsymbol{\pi}}_A^\star$, $\hat{\boldsymbol{\pi}}_B^\star$, and $\hat{\boldsymbol{\pi}}_{A_{local}}^\star$ have been estimated. After estimation, each set $\hat{\boldsymbol{\pi}}_{(\cdot)}^\star$ has been validated by moving an IR robot along three set of trajectories:

- *Test Case 1*: 30 random trajectories generated by the Motion Planner of the IR and spanning the workspace sub-region where $\hat{\boldsymbol{\pi}}_B^\star$ and $\hat{\boldsymbol{\pi}}_{A_{local}}^\star$ have been estimated;
- *Test Case 2*: 30 random trajectories generated as sum of sines similarly to (1) and spanning the whole workspace;
- *Test Case 3*: 30 random trajectories generated from the IR motion planner and spanning the whole workspace;

For each trajectory of each *Test Case* the measured torques $\widetilde{\mathbf{T}}$ are then compared to estimated torques $\hat{\mathbf{T}}_A$, $\hat{\mathbf{T}}_{A_{local}}$ and $\hat{\mathbf{T}}_B$ figured out of the corresponding calibration algorithm, and the vectors of the torques-error have been calculated as:

$$\begin{gathered} \mathbf{e}_A(\widetilde{\ddot{\mathbf{q}}}, \widetilde{\dot{\mathbf{q}}}, \widetilde{\mathbf{q}}) = |\widetilde{\mathbf{T}} - \Phi\hat{\boldsymbol{\pi}}_A^\star|, \, \mathbf{e}_B(\widetilde{\ddot{\mathbf{q}}}, \widetilde{\dot{\mathbf{q}}}, \widetilde{\mathbf{q}}) = |\widetilde{\mathbf{T}} - \Phi\hat{\boldsymbol{\pi}}_B^\star|, \\ \mathbf{e}_{A_{local}}(\widetilde{\ddot{\mathbf{q}}}, \widetilde{\dot{\mathbf{q}}}, \widetilde{\mathbf{q}}) = |\widetilde{\mathbf{T}} - \Phi\hat{\boldsymbol{\pi}}_{A_{local}}^\star|. \end{gathered} \tag{10}$$

Then, for each *Test Case* the three mean values μ_{e_A}, $\mu_{e_{A_{local}}}$, μ_{e_B} of the measured errors and the corresponding standard deviation σ_A, $\sigma_{A_{local}}$, and σ_B have been calculated.

E. Experimental Setup

Experiments are shown using a COMAU NS16 manipulator, with the C4GOpen controller and the virtualizer of its Motion Planner *ORL -Open Realistic Robot Library*. None mass has been added to the flange (axis 6) of the robot in order to not influence the algorithm estimation.

[3] Algorithm A in [19] implements the Kalman LS algorithm, but for our experimental setup WLS method has reached greater result.

[4] [19] displays that the optimization on conditioning number improve the results for their set-up, while our experimental outcomes are opposite.

TABLE II: Experimental Results: statistics have bene calculated over 30 different random trajectories for each test case.

	Ax.	Mean Absolute Error Nm			Percentage Error w.r.t Nominal Axis Torque %		
		$\mu_{e_A} \pm \sigma_{e_A}$	$\mu_{e_{A_{loc}}} \pm \sigma_{e_{A_{loc}}}$	$\mu_{e_B} \pm \sigma_{e_B}$	$\mu_{e_A} \pm \sigma_{e_A}$	$\mu_{e_{A_{loc}}} \pm \sigma_{e_{A_{loc}}}$	$\mu_{e_B} \pm \sigma_{e_B}$
Test Case 1 Local trajectory IR Motion Planner	1	14.3 ± 13.6	17.5 ± 14.7	$\mathbf{10.2 \pm 8.8}$	2.6 ± 2.5	3.2 ± 2.7	$\mathbf{1.9 \pm 1.6}$
	2	35.0 ± 21.7	21.4 ± 16.4	$\mathbf{15.9 \pm 15.3}$	3.7 ± 2.3	2.2 ± 1.7	$\mathbf{1.7 \pm 1.6}$
	3	12.0 ± 7.1	15.3 ± 10.5	$\mathbf{10.2 \pm 6.9}$	2.5 ± 1.4	3.1 ± 2.2	$\mathbf{2.1 \pm 1.4}$
	4	3.6 ± 5.2	5.8 ± 6.8	$\mathbf{1.6 \pm 2.2}$	7.0 ± 10.3	11.5 ± 13.4	$\mathbf{3.1 \pm 4.3}$
	5	2.0 ± 2.1	2.3 ± 2.9	$\mathbf{1.6 \pm 1.7}$	3.6 ± 3.9	4.1 ± 5.2	$\mathbf{2.9 \pm 3.0}$
	6	10.5 ± 9.4	5.8 ± 5.1	$\mathbf{1.5 \pm 1.3}$	30.1 ± 26.9	16.6 ± 14.5	$\mathbf{4.2 \pm 3.7}$
Test Case 2 Wide trajectory Sinusoidal Template	1	$\mathbf{10.4 \pm 10.6}$	11.1 ± 10.7	43.8 ± 51.5	$\mathbf{1.9 \pm 2.0}$	2.0 ± 2.0	8.0 ± 9.5
	2	$\mathbf{14.7 \pm 15.7}$	47.9 ± 41.3	43.4 ± 31.7	$\mathbf{1.5 \pm 1.6}$	5.0 ± 4.3	4.5 ± 3.3
	3	$\mathbf{6.0 \pm 5.8}$	21.2 ± 8.7	21.4 ± 27.3	$\mathbf{1.2 \pm 1.2}$	4.4 ± 1.8	4.4 ± 5.6
	4	$\mathbf{0.9 \pm 0.9}$	1.1 ± 1.0	1.4 ± 1.5	$\mathbf{1.8 \pm 1.8}$	2.2 ± 2.0	2.8 ± 2.9
	5	$\mathbf{1.1 \pm 1.5}$	1.3 ± 1.6	1.2 ± 1.8	$\mathbf{2.0 \pm 2.8}$	2.3 ± 2.8	2.2 ± 3.3
	6	2.4 ± 3.2	1.6 ± 1.9	$\mathbf{1.3 \pm 1.3}$	6.9 ± 9.0	4.6 ± 5.5	$\mathbf{3.6 \pm 3.8}$
Test Case 3 Wide trajectory IR Motion Planner	1	$\mathbf{39.4 \pm 53.4}$	49.3 ± 62.1	211.5 ± 422.9	$\mathbf{7.2 \pm 9.8}$	9.1 ± 11.4	38.8 ± 77.6
	2	$\mathbf{58.2 \pm 66.7}$	101.1 ± 93.9	99.1 ± 84.2	$\mathbf{6.1 \pm 7.0}$	10.6 ± 9.8	10.4 ± 8.8
	3	$\mathbf{21.5 \pm 19.3}$	33.1 ± 26.9	64.1 ± 75.9	$\mathbf{4.4 \pm 4.0}$	6.8 ± 5.5	13.1 ± 15.6
	4	$6.6 \pm \mathbf{4.5}$	8.9 ± 8.2	$\mathbf{5.7} \pm 5.4$	$12.9 \pm \mathbf{8.9}$	17.5 ± 16.2	$\mathbf{11.2} \pm 10.6$
	5	4.4 ± 3.7	$\mathbf{4.1 \pm 3.4}$	4.2 ± 3.9	8.0 ± 6.6	$\mathbf{7.5 \pm 6.2}$	7.5 ± 7.0
	6	7.5 ± 4.7	5.0 ± 2.6	$\mathbf{1.6 \pm 1.4}$	21.5 ± 13.5	14.3 ± 7.3	$\mathbf{4.6 \pm 3.9}$

IV. Results and Discussion

1) Test Case 1: looking at first block of Table II, B better predicts torques with respect to A and A_{loc}, with small value of μ_e correlated with small σ. Main error is on the 6^{th} axes, where mass and inertia are small and difficultly identifiable and the main contribution is due to static friction. Regarding the trend, see Fig. 2, the match between the shape of the foreseen torques and of the measured ones are good.

2) Test Case 2: looking at second block of Table II, A works better than A_{loc} and B, except for the axis 6 where the results of B is slightly better than A and A_{local} because the static friction dominates over the parameters and the movements performed with B are enough excitatory. As in the previous case, small μ_e are correlated with small σ. It is worth to show that the performance reached from A in this test are equivalent to the performance reached from B in the *Test Case 1*, and it would be due to the fact that in both cases the test trajectories belong to the same class of trajectories used in the algorithm definition.

3) Test Case 3: looking at third block of Table II, even though the optimization area of A covers the same area of the trajectories tested, the results are four/five time worse than the ones reached in *Test Case 2*. As expected, also the other two algorithms attain poor results, not only with respect to the statistics, but also the foreseen torque profile is far from the actual. Nevertheless the negative performance, it is worth to note that B displays clearly problem on the tuning of the static friction threshold (particularly evident on axis 5 of Fig. 4, where continuous change of velocity direction, produce a "step" effect on predicted torque). This issue would be a good evidence in order to improve the optimization of the B also for the local trajectories.

4) Remarks: looking at Table III, the three set of minimum dynamics parameters estimated from the three algorithms do not have physical meaning despite they guarantee good performance in the torque prediction. In fact, the optimization is based on pure mathematical consideration, D-optimization, and none physical feature is considered. Furthermore, analyzing the absolute value attained from the fitnesses, see Table I, it is worth to note that f_A and $f_{A_{loc}}$ reach higher value than f_B and, on the basis of the optimization criteria adopted, this should guarantee a better torques estimation for $\hat{\boldsymbol{\pi}}^{\star}_{A_{loc}}$ respect to $\hat{\boldsymbol{\pi}}^{\star}_{B}$ because they are defined on the same optimization region, but this is clearly in contrast with experiments. Hence, the optimization criteria used does not correspond straightforwardly to the identification of the actual parameters of the multi-body rigid model, and we would assume that once the determinant of the linear system is "huge" further increases on its value, doesn't determine more exciting trajectory for robot dynamic. In addition, the experiments display also the importance of the template-class of trajectories in the estimation with respect to the trajectories that are used as comparison. In fact, A does not reach appreciable results when it is asked to foreseen torques for movements that have acceleration and velocity profile extremely different from the ones used in the optimization, and furthermore A_{local} is far from acceptability when applied in bounded workspace sub-region. Finally, experiments seem to demonstrate that scalability of standard algorithm for local calibration is not straightforward and design of proper algorithm should improve the accuracy in the torque prediction.

On the basis of these considerations, the good performance of B in the bounded workspace sub-region could be not directly related on the optimization criteria (the maximization of the regressor determinant) but should be associated to the class of trajectories used in the optimization, and further investigation is necessary. In fact, experiments show as the integration of standard Motion Planner in the optimization phase is extremely important in order to guarantee high performances in the daily use of IRs.

TABLE III: Parameters list as in [20], an extension of [5] considering also the 18 parameters of the friction model.

Par. Id	$\hat{\pi}_A^*$	$\hat{\pi}_{A_{local}}^*$	$\hat{\pi}_B^*$	Par. Id	$\hat{\pi}_A^*$	$\hat{\pi}_{A_{local}}^*$	$\hat{\pi}_B^*$	Par. Id	$\hat{\pi}_A^*$	$\hat{\pi}_{A_{local}}^*$	$\hat{\pi}_B^*$
1	1.279	1.377	3.243	21	-0.264	-0.008	-0.047	41	95.400	53.685	34.066
2	-3.752	-1.795	108.411	22	1.830	1.000	1.604	42	0.384	0.900	1.610
3	-3.609	12.804	35.218	23	89.784	67.542	-56.797	43	68.128	46.422	73.215
4	0.747	-1.151	23.861	24	73.083	74.598	64.888	44	1.678	2.942	2.276
5	0.159	1.776	22.557	25	-41.527	-144.919	-314.818	45	38.191	39.897	45.623
6	13.140	21.054	14.482	26	-2.633	-4.421	9.635	46	1.193	1.313	1.102
7	0.031	-0.017	-0.013	27	102.558	83.196	114.217	47	5.350	4.829	5.470
8	-0.190	-0.188	-1.237	28	10.685	11.606	10.224	48	0.194	0.213	0.128
9	0.556	-0.281	5.167	29	12.654	10.404	13.079	49	9.634	10.628	9.934
10	0.761	-0.666	1.624	30	1.554	116.634	262.003	50	0.231	0.228	0.242
11	-0.097	-0.070	-0.035	31	-1.074	-1.122	-268.422	51	5.440	5.107	4.743
12	0.127	-0.649	-0.059	32	8.494	7.535	9.014	52	-0.113	-0.017	0.069
13	0.464	0.193	0.873	33	-0.106	-0.123	-0.125	53	42.751	7.788	-11.849
14	0.171	0.216	0.014	34	-0.290	-2.197	11.602	54	15.874	-11.487	30.144
15	3.656	3.281	2.664	35	-0.628	-0.102	-1.328	55	-15.013	-21.814	28.873
16	-0.033	0.003	-0.050	36	0.097	-0.277	0.495	56	0.222	0.261	5.323
17	0.012	0.009	-0.021	37	0.710	0.718	0.690	57	0.683	0.884	7.059
18	0.209	0.514	0.033	38	-0.634	-1.115	-0.311	58	0.022	-0.652	4.552
19	0.207	-0.006	-0.005	39	0.157	-0.038	0.298				
20	0.074	0.050	-0.025	40	-0.477	-0.321	0.064				

V. Conclusions

The paper have discussed about the IRs dynamic calibration and how to improve locally the estimation of multi-body rigid models. Three algorithms have been implemented and compared, testing their performances in workspace sub-region and in the whole workspace. Experimentally, algorithms designed for global dynamic calibration do not attain good results when tested along bounded trajectories, and dedicated algorithms sound technically good. The paper raises the issue on the metrics used for the performance evaluation and on the optimization criteria generally adopted.

Acknowledgments

T. Dinon, fellow research of CNR-ITIA, R. Bozzi, and J. C. Dalberto, laboratory technicians of CNR-ITIA, have been involved in setting up the experiments.

References

[1] P. K. Khosla and T. Kanade, "Experimental evaluation of nonlinear feedback and feedforward control schemes for manipulators," *The Int. J. of Robotics Research*, vol. 7, no. 1, pp. 18–28, 1988.

[2] P. Chiacchio, L. Sciavicco, and B. Siciliano, "The potential of model-based control algorithms for improving industrial robot tracking performance," in *Intelligent Motion Control, 1990. Proc. of the IEEE Int. Workshop on*, vol. 2, aug 1990, pp. 831 –836.

[3] C. G. Atkeson, C. H. An, and J. M. Hollerbach, "Estimation of inertial parameters of manipulator loads and links," *The Int. J. of Robotics Research*, vol. 5, no. 3, pp. 101–119, 1986.

[4] M. Gautier and W. Khalil, "On the identification of the inertial parameters of robots," in *Decision and Control, 1988., Proc. of the 27th IEEE Conf. on*, vol. 3, dec 1988, pp. 2264 –2269.

[5] G. Antonelli, F. Caccavale, and P. Chiacchio, "A systematic procedure for the identification of dynamic parameters of robot manipulators," *Robotica*, vol. 17, no. 04, pp. 427–435, 1999.

[6] B. Armstrong, "On finding exciting trajectories for identification experiments involving systems with nonlinear dynamics," *The Int. J. of Robotics Research*, vol. 8, no. 6, pp. 28–48, 1989.

[7] M. Gautier and W. Khalil, "Exciting trajectories for the identification of base inertial parameters of robots," *The Int. J. of Robotics Research*, vol. 11, no. 4, pp. 362–375, 1992.

[8] C. Presse and M. Gautier, "New criteria of exciting trajectories for robot identification," in *Rob. and Aut., Proc., IEEE Int. Conf. on*, may 1993, pp. 907 –912 vol.3.

[9] J. Swevers, C. Ganseman, D. Tukel, J. de Schutter, and H. Van Brussel, "Optimal robot excitation and identification," *Robotics and Automation, IEEE Transactions on*, vol. 13, no. 5, pp. 730 –740, oct 1997.

[10] K. Park, "Fourier-based optimal excitation trajectories for the dynamic identification of robots," *Robotica*, vol. 24, no. 5, pp. 625–633, 2006.

[11] D. Bristow, M. Tharayil, and A. Alleyne, "A survey of iterative learning control," *Control Systems, IEEE*, vol. 26, pp. 96–114, 2006.

[12] A. Visioli, G. Ziliani, and G. Legnani, "Iterative-learning hybrid force/velocity control for contour tracking," *Robotics, IEEE Transactions on*, vol. 26, no. 2, pp. 388 –393, april 2010.

[13] M. Spong, "Modeling and control of elastic joint robots." *J. of Dyn. Syst., Meas. and Contr., Trans. of the ASME*, vol. 109, no. 4, pp. 310–319, 1987.

[14] M. Östring, S. Gunnarsson, and M. Norrlf, "Closed-loop identification of an industrial robot containing flexibilities," *Control Engineering Practice*, vol. 11, no. 3, pp. 291–300, 2003.

[15] A. De Luca and L. Lanari, "Robots with elastic joints are linearizable via dynamic feedback," in *Decision and Control, 1995., Proc. of the 34th IEEE Conference on*, vol. 4, 1995, pp. 3895–3897 vol.4.

[16] W. He, S. S. Ge, and J. Zhang, "Dynamic modeling and system identification for the lower body of a social robot with flexible joints," in *IEEE/SICE Int. Symp. on Sys. Int.*, 2011, pp. 342–347.

[17] M. W. Spong, "Adaptive control of flexible joint manipulators," *Systems & Control Letters*, vol. 13, no. 1, pp. 15–21, 1989.

[18] A. De Luca and P. Lucibello, "A general algorithm for dynamic feedback linearization of robots with elastic joints," in *Robotics and Automation, Proc., 1998 IEEE Int. Conf. on*, vol. 1, 1998, pp. 504–510.

[19] G. Calafiore, M. Indri, and B. Bona, "Robot dynamic calibration: Optimal excitation trajectories and experimental parameter estimation," *J. of Robotic Systems*, vol. 18, no. 2, pp. 55–68, 2001.

[20] E. Villagrossi, N. Pedrocchi, F. Vicentini, and L. Molinari Tosatti, "Optimal robot dynamics local identification using genetic-based path planning in workspace subregions," in *Advanced Intelligent Mechatronics (AIM), 2013 IEEE/ASME Int. Conf. on*, 2013, pp. 932–937.

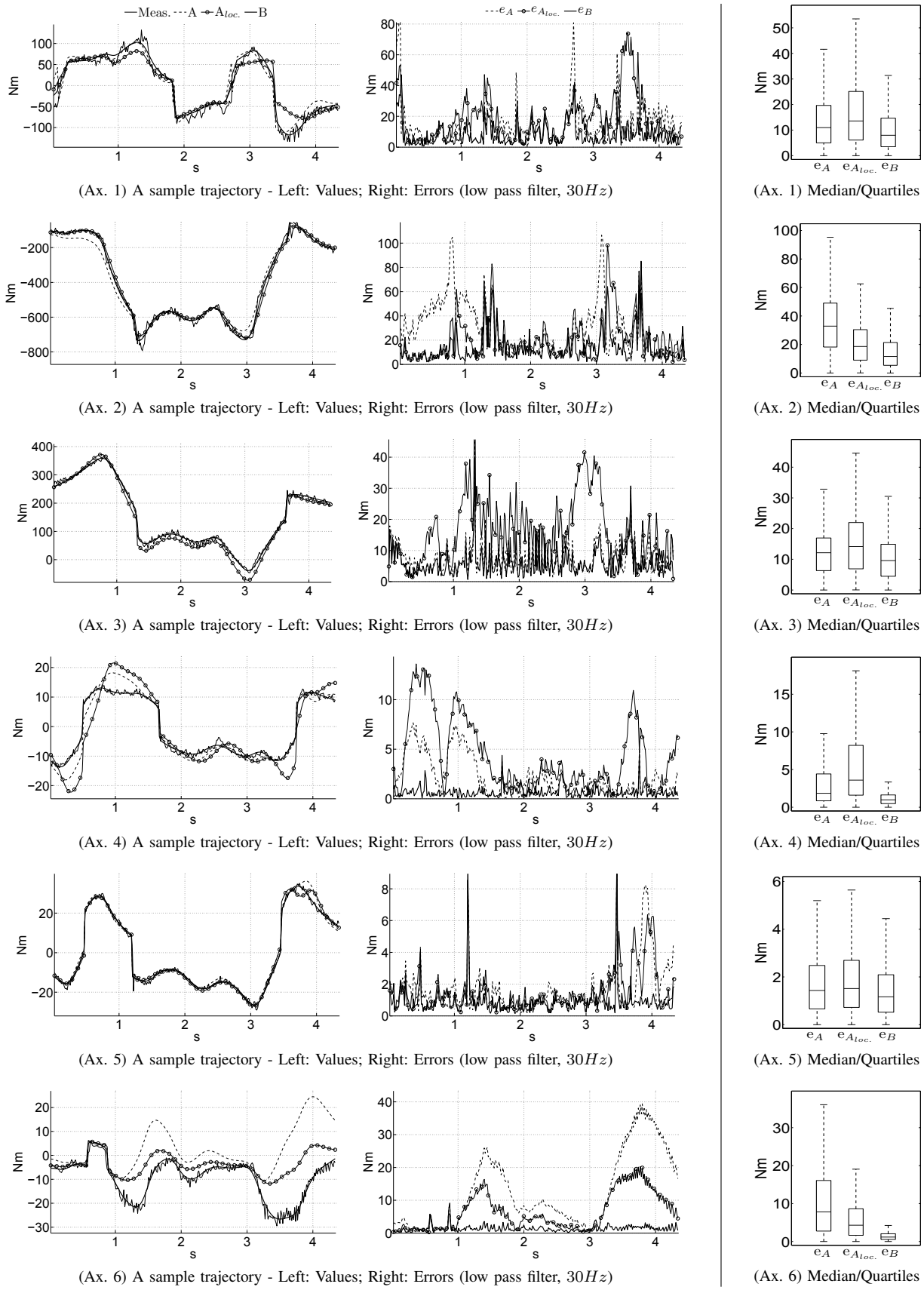
[21] B. Raucent and J. C. Samin, "Minimal parametrization of robot dynamic models," *Mechanics of Structures and Machines*, vol. 22, no. 3, pp. 371–396, 1994.

[22] M. Indri, G. Calafiore, G. Legnani, F. Jatta, and A. Visioli, "Optimized dynamic calibration of a scara robot," in *IFAC '02. 2002 IFAC Internation Federation on Automatic Control*, 2002.

[23] M. Gautier and W. Khalil, "Direct calculation of minimum inertial parameters of serial robots," *Robotics and Automation, IEEE Transactions on*, vol. 6, no. 3, pp. 368–373, 1990.

[24] F. Benimeli, V. Mata, and F. Valero, "A comparison between direct and indirect dynamic parameter identification methods in industrial robots," *Robotica*, vol. 24, no. 5, pp. 579–590, Sept. 2006.

[25] M. Gautier, "Dynamic identification of robots with power model," in *Rob. and Aut., Proc., IEEE Int. Conf. on*, vol. 3, 1997, pp. 1922 –1927.



(Ax. 1) A sample trajectory - Left: Values; Right: Errors (low pass filter, $30Hz$)

(Ax. 1) Median/Quartiles

(Ax. 2) A sample trajectory - Left: Values; Right: Errors (low pass filter, $30Hz$)

(Ax. 2) Median/Quartiles

(Ax. 3) A sample trajectory - Left: Values; Right: Errors (low pass filter, $30Hz$)

(Ax. 3) Median/Quartiles

(Ax. 4) A sample trajectory - Left: Values; Right: Errors (low pass filter, $30Hz$)

(Ax. 4) Median/Quartiles

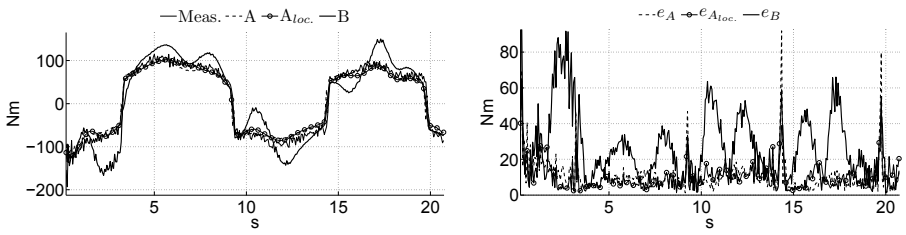
(Ax. 5) A sample trajectory - Left: Values; Right: Errors (low pass filter, $30Hz$)

(Ax. 5) Median/Quartiles

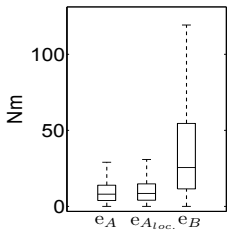
(Ax. 6) A sample trajectory - Left: Values; Right: Errors (low pass filter, $30Hz$)

(Ax. 6) Median/Quartiles

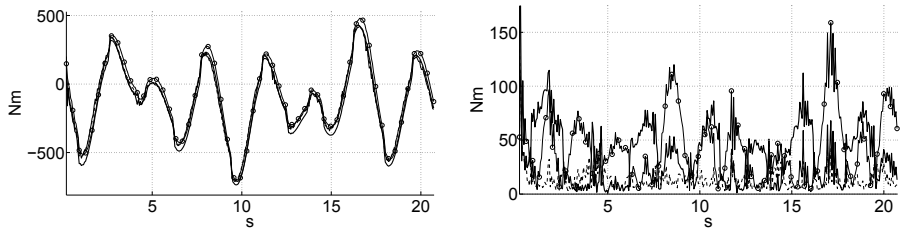
Fig. 2: Test Case 1: 30 randomly trajectories generated within bounded sub-region by the IR Motion Planner. On the Left are shown the Value Measured and Estimated with the three Algorithms A, A_{local} and B for the execution of one sample trajectory of the set, on the Right the distribution of the prediction error.



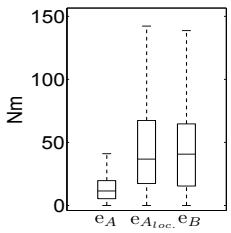
(Ax. 1) A sample trajectory - Left: Values; Right: Errors (low pass filter, $30Hz$)



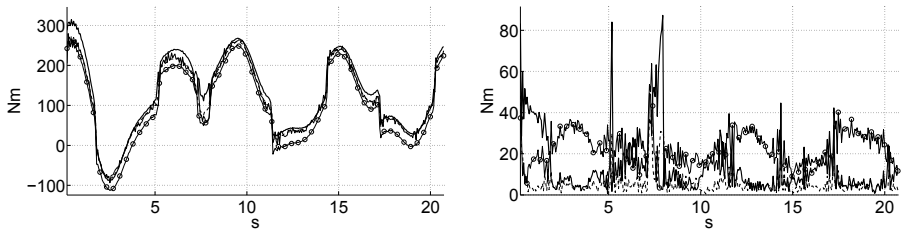
(Ax. 1) Median/Quartiles



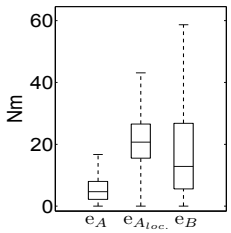
(Ax. 2) A sample trajectory - Left: Values; Right: Errors (low pass filter, $30Hz$)



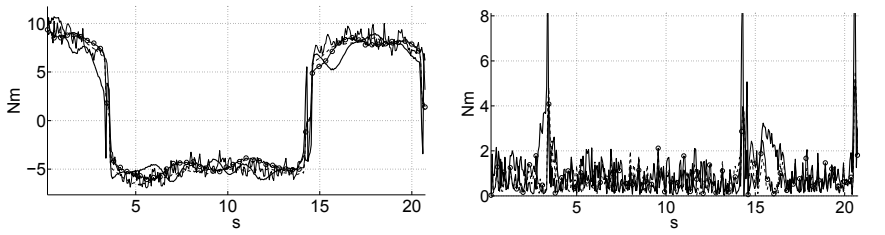
(Ax. 2) Median/Quartiles



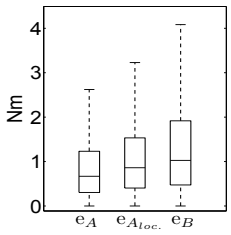
(Ax. 3) A sample trajectory - Left: Values; Right: Errors (low pass filter, $30Hz$)



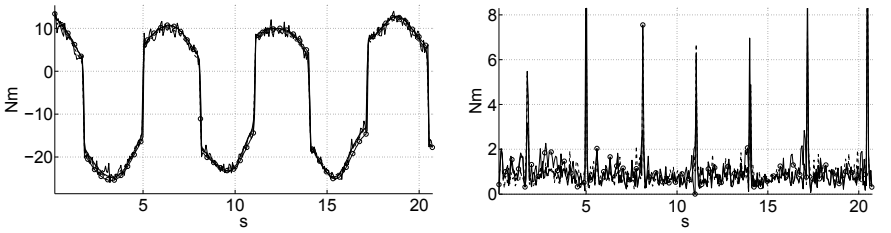
(Ax. 3) Median/Quartiles



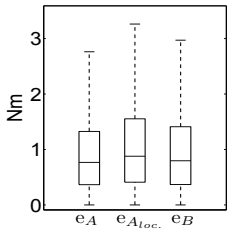
(Ax. 4) A sample trajectory - Left: Values; Right: Errors (low pass filter, $30Hz$)



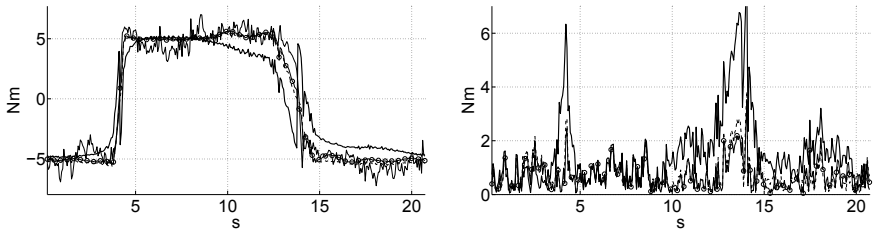
(Ax. 4) Median/Quartiles



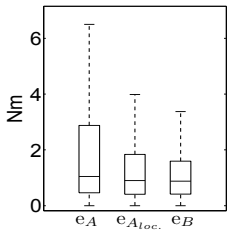
(Ax. 5) A sample trajectory - Left: Values; Right: Errors (low pass filter, $30Hz$)



(Ax. 5) Median/Quartiles

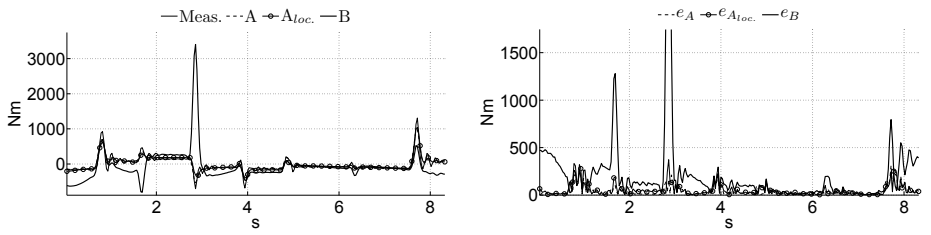


(Ax. 6) A sample trajectory - Left: Values; Right: Errors (low pass filter, $30Hz$)

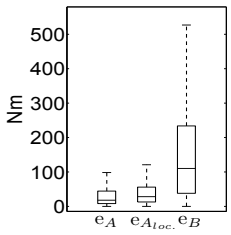


(Ax. 6) Median/Quartiles

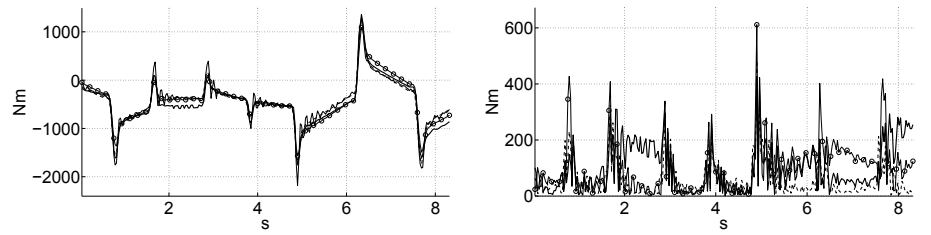
Fig. 3: Test Case 2: 30 randomly wide trajectories covering the whole workspace generated as sum of sinusoidal functions. On the Left are shown the Value Measured and Estimated with the three Algorithms A, A_{local} and B for the execution of one sample trajectory of the set, on the Right the distribution of the prediction error.



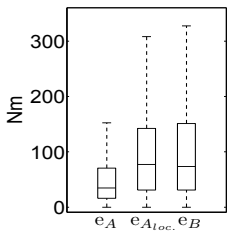
(Ax. 1) A sample trajectory - Left: Values; Right: Errors (low pass filter, $30Hz$)



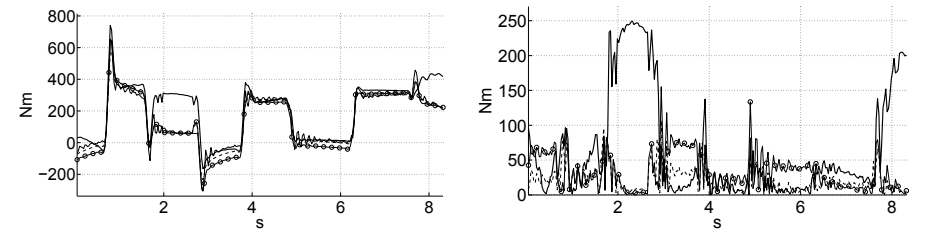
(Ax. 1) Median/Quartile



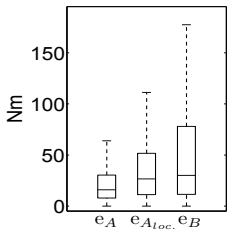
(Ax. 2) A sample trajectory - Left: Values; Right: Errors (low pass filter, $30Hz$)



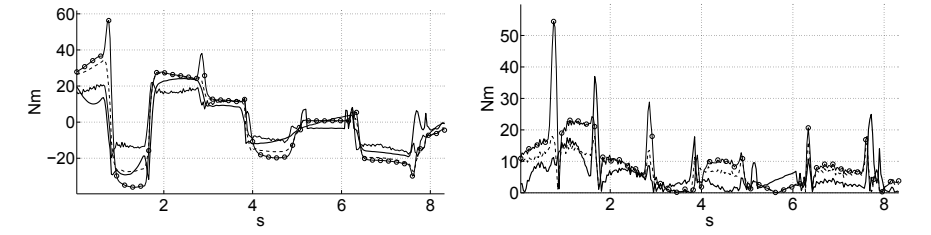
(Ax. 2) Median/Quartile



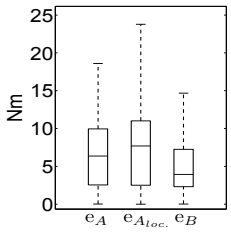
(Ax. 3) A sample trajectory - Left: Values; Right: Errors (low pass filter, $30Hz$)



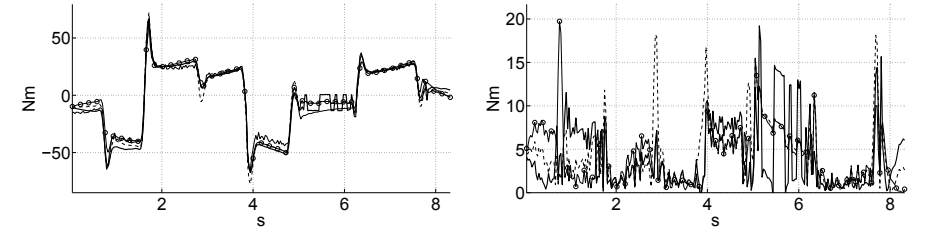
(Ax. 3) Median/Quartile



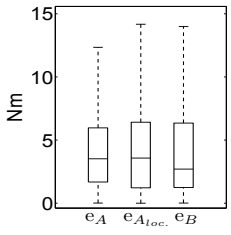
(Ax. 4) A sample trajectory - Left: Values; Right: Errors (low pass filter, $30Hz$)



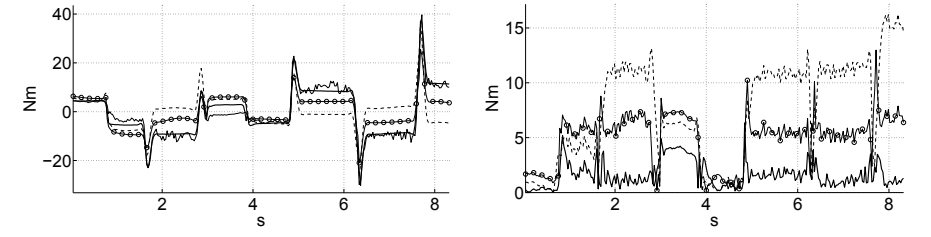
(Ax. 4) Median/Quartile



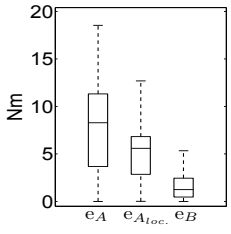
(Ax. 5) A sample trajectory - Left: Values; Right: Errors (low pass filter, $30Hz$)



(Ax. 5) Median/Quartile



(Ax. 6) A sample trajectory - Left: Values; Right: Errors (low pass filter, $30Hz$)



(Ax. 6) Median/Quartile

Fig. 4: Test Case 3: 30 randomly wide trajectories covering the whole workspace generated generated from the IR Motion Planner. On the Left are shown the Value Measured and Estimated with the three Algorithms A, A_{local} and B for the execution of one sample trajectory of the set, on the Right the distribution of the prediction error.