Robot Dynamic Calibration: Optimal Excitation Trajectories and Experimental Parameter Estimation

G. Calafiore, M. Indri,* and B. Bona

Dipartimento di Automatica e Informatica

Politecnico di Torino

Corso Duca degli Abruzzi, 24-10129 Torino, Italy

e-mail: indri@polito.it

Received 17 April 1999; revised 13 October 2000

Advanced robot control schemes require an accurate knowledge of the dynamic parameters of the manipulator. This article examines various issues related to robot dynamic calibration, from generation of optimal excitation trajectories to data acquisition and filtering, and experimental inertial and friction parameter estimation. In particular, a new method is developed for the determination of optimal joint trajectories for the calibration experiment, which is based on evolutionary optimization techniques. A genetic algorithm is used to determine excitation trajectories that minimize either the condition number of the regression matrix or the logarithmic determinant of the Fisher information matrix. All the calibration steps have been carried out on a SCARA two-link planar manipulator, and the experimental results are discussed. © 2001 John Wiley & Sons, Inc.

1. INTRODUCTION

In robotics, the implementation of advanced control algorithms that take into account the manipulator nonlinearities (like the inverse dynamics method) requires a good knowledge of the dynamic model of the robot. While the structure of the motion differential equations of a manipulator is well known, the correct values of the involved parameters are not always available, because they are often not directly measurable. In particular, the determination of the

dynamic parameters can be achieved only by means of a proper calibration procedure, which estimates their values from input and output data provided by sensors and internal measurement devices, or reconstructed via software.

Two different calibration procedures are commonly used to estimate the dynamic parameters of a manipulator: the first one is based on the recursive formulation of the Newton–Euler motion equations^{1,2} the second one is based on the Lagrangian energy approach.³ Both approaches lead to dynamic equations that are linear in a minimally reduced set of identifiable parameters (the base parameter set),

To whom correspondence should be addressed.

which can be determined by applying the reduction procedure proposed by Gautier and Khalil.⁴ The preliminary aspects related to manipulator modeling for dynamic calibration are recalled in Section 2.

Estimation of the identifiable parameters is usually achieved by application of a least-squares algorithm to the motion or the energy equations of the robot. The validity of the provided estimate strongly depends on the quality of the available input-output signals and on the choice of the imposed reference trajectories, which must properly "excite" the robot dynamics. The quality of the used input-output signals can be enhanced by using filters and/or observers (see, e.g., refs. 5 and 29).

The problem of finding exciting trajectories for the identification has been discussed in several papers. The work presented in Armstrong⁷ is among the oldest and it proposed a functional approach, in which the calculus of variations is used to solve a nonlinear path optimization problem: An optimal acceleration sequence is determined, and then velocities and positions are computed by numerical integration. The weak points of such an approach are the difficulty of including trajectory feasibility constraints in the optimization procedure and high computational burden due to the large number of decision variables required when the method is discretized.

The methods proposed in refs. 8 and 9 are both based on the determination of a sequence of optimal joint position-velocity couples. A continuous smooth "optimal" trajectory is then found by interpolating them via fixed order polynomials; trajectory constraints are only checked at the end. The number of decision variables and, consequently, the computational effort in this case are dependent on the number of samples along the trajectory: Gautier and Khalil⁹ reported numerical intractability of the optimization problem in their test cases for a number of samples greater than 50.

An alternative solution to the above-described methods, which are gradient based (and thus require time-consuming computations to exactly determine or estimate the gradient of the objective function), is proposed in ref. 10, where a random search approach is used to determine the set of optimal points to be interpolated.

The method proposed in this article is based on the parameterization of the class of joint position time functions. The key idea underlying this method is to restrict attention to some parameterized family

of possible trajectories, thus reducing the original infinite-dimensional problem to a more tractable finitely parameterized optimization problem. This is, of course, done at the expense that the solution will be optimal in the selected class, i.e., suboptimal with respect to the original problem. However, what is of interest in practice is to determine a "good" (albeit not globally optimal) excitation trajectory that leads to an informative identification experiment. In this respect, the above drawback does not seem to be critical in an experimental or industrial context, as long as the family of allowable trajectories is chosen to be "rich enough," according to engineering judgment and experimental trials. In this article, in particular, we consider the class of harmonic functions with a fixed, finite number of harmonic terms. The introduced parameterization takes into account the frequency band constraints on the input excitation, and the number of decision parameters of the resulting optimization problem is independent of the number of time samples along the trajectory. Harmonic parameterization also was used in this context by Swevers et al.¹¹

The trajectory parameterization reduces the problem to a nonlinear parametric optimization that is solved here using a genetic algorithm (GA). The convergence of the GAs to the global optimum is not, in general, guaranteed, but in practice such algorithms have proved to be particularly efficient and robust in the global search of solutions of nonlinear and multimodal optimization problems.^{12,13} In robotics, genetic algorithms have been used, for instance, for motion planning of mobile manipulators¹⁴ and for estimation of kinematic parameters.¹⁵ In our case, the use of GAs in conjunction with the introduced trajectory parameterization allows us to take into account the physical constraints on joint positions, velocities, and accelerations directly in the decoding phase of the *genotype* of the *individuals*, i.e., of the strings containing the binary codes of the decision variables, so that the resulting trajectories are certainly feasible.

In the experimental part of this article, the complete calibration procedure is performed on a SCARA two-link manipulator produced by IMI (Integrated Motions Inc., Berkeley, CA), starting from trajectories generation according to the proposed method, through issues related to data acquisition and filtering, to the estimation of the inertial and friction parameters of the manipulator and model validation.

2. ROBOT CALIBRATION MODEL

Consider an n degrees-of-freedom (dof) manipulator, having only revolute joints for the sake of simplicity (the subsequent developments can be easily extended to the most general case). Let the kinematics of the robot be described by introducing Denavit–Hartenberg notation to define local frames attached to the links. According to such notation, frame $\mathcal{R}_{i-1}$, $i = 1, \dots, n$, is the reference frame located in correspondence to joint i , with the z axis coinciding with the rotational axis of the respective joint ($\mathcal{R}_0$ is the inertial frame), whereas $\mathcal{R}_n$ is the frame attached to the end effector. Matrix $\mathbf{R}_i^{i-1}$ is the rotation matrix from frame $\mathcal{R}_{i-1}$ to frame $\mathcal{R}_i$. In addition, let $\mathbf{q}$ be the joint position vector, where each coordinate q_i represents the angular position of joint i , defined according to Denavit–Hartenberg rules.

The motion equations of the manipulator can be determined following the recursive Newton–Euler formulation, by using two transformation laws: the first one is represented by the *forward* transformation law of the robot kinematics from the base to the end effector, while the second one is given by the *backward* computation of the joint torques from the end effector to the base. For a manipulator that has only revolute joints, the recursive Newton–Euler equations are (dependence on time is omitted)

$$\boldsymbol{\omega}_{i+1} = \mathbf{R}_i^{i+1} [\boldsymbol{\omega}_i + \mathbf{z}_i \dot{q}_{i+1}], \quad (1a)$$

$$\dot{\boldsymbol{\omega}}_{i+1} = \mathbf{R}_i^{i+1} [\dot{\boldsymbol{\omega}}_i + \mathbf{z}_i \ddot{q}_{i+1} + \boldsymbol{\omega}_i \times (\mathbf{z}_i \dot{q}_{i+1})], \quad (1b)$$

$$\dot{\mathbf{v}}_{i+1} = \mathbf{R}_i^{i+1} \dot{\mathbf{v}}_i + \dot{\boldsymbol{\omega}}_{i+1} \times \mathbf{p}_{i+1} + \boldsymbol{\omega}_{i+1} \times (\boldsymbol{\omega}_{i+1} \times \mathbf{p}_{i+1}), \quad (1c)$$

$$\mathbf{f}_i = \mathbf{R}_{i+1}^i \mathbf{f}_{i+1} + m_i \dot{\mathbf{v}}_i + \dot{\boldsymbol{\omega}}_i \times m_i \mathbf{s}_i + \boldsymbol{\omega}_i \times (\boldsymbol{\omega}_i \times m_i \mathbf{s}_i), \quad (2a)$$

$$\mathbf{n}_i = \mathbf{R}_{i+1}^i \mathbf{n}_{i+1} + \mathbf{p}_i \times \mathbf{f}_i + \Gamma_i \dot{\boldsymbol{\omega}}_i + \boldsymbol{\omega}_i \times (\Gamma_i \boldsymbol{\omega}_i) + m_i \mathbf{s}_i \times \dot{\mathbf{v}}_i, \quad (2b)$$

$$\boldsymbol{\tau}_i = \mathbf{n}_i^T \mathbf{R}_{i-1}^i \mathbf{z}_{i-1} + \tau_{fi}, \quad (2c)$$

where $\boldsymbol{\omega}_i$ is the angular velocity of joint i , $\mathbf{v}_i$ is the linear velocity of the origin of frame $\mathcal{R}_i$, $\mathbf{f}_i$ and $\mathbf{n}_i$ are, respectively, the force and the moment applied to link i by link $i-1$, τ_i is the torque applied to joint i , τ_{fi} is the friction torque acting on joint i , $\mathbf{z}_i$ denotes the z axis of frame $\mathcal{R}_i$ expressed in its own frame (i.e., $\mathbf{z}_i = [0 \ 0 \ 1]^T$ for all i), $\mathbf{p}_i$ represents the position of the origin of frame $\mathcal{R}_i$ with respect

to $\mathcal{R}_{i-1}$, m_i is the mass of link i , $\mathbf{s}_i$ is the position vector of the center of mass of link i , and Γ_i is the inertia tensor with respect to frame $\mathcal{R}_i$. For notational convenience, vectors with subscript i in Eqs. (1) and (2) are represented in the reference frame $\mathcal{R}_i$.

The friction torques τ_{fi} can be expressed according to one of several models proposed in the literature, taking into account the different friction components, e.g., stiction, Coulomb, and viscous friction.¹⁶ In general, the friction torque vector $\boldsymbol{\tau}_f$ can be written as

$$\boldsymbol{\tau}_f = \mathbf{F}(\mathbf{q}, \dot{\mathbf{q}}) \boldsymbol{\theta}_f, \quad (3)$$

where the expression of the function $\mathbf{F}(\mathbf{q}, \dot{\mathbf{q}})$ and the choice of the parameter vector $\boldsymbol{\theta}_f \in \mathbb{R}^{n_f}$ depend on the considered friction model. Note that if only Coulomb and viscous friction are included in the model, then $\mathbf{F}(\mathbf{q}, \dot{\mathbf{q}})$ is a function of velocity $\dot{\mathbf{q}}$ only.

By introducing vector $\boldsymbol{\tau}$, containing the n torques applied to the joints, Eqs. (1) and (2) can be rearranged in the form

$$\boldsymbol{\tau} = \bar{\mathbf{D}}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \boldsymbol{\theta}_m, \quad (4)$$

which is linear with respect to the vector $\boldsymbol{\theta}_m \in \mathbb{R}^m$ which contains all the inertial parameters of the manipulator, i.e., the masses m_i , the inertia moments (six for each link), and the first-order moments (three for each link) of the robot links^{3,17} and the n_f friction parameters. $\bar{\mathbf{D}}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$, which is a nonlinear function of joint position, velocity, and acceleration vectors, is then an $n \times m$ matrix with $m = 10n + n_f$.

It is well known that only some of the inertial parameters of the robot really affect its dynamics and are then identifiable by calibration procedures. It is possible to determine the set of minimum number of inertial parameters, i.e., the base parameter set containing only the p identifiable dynamic parameters collected in $\boldsymbol{\theta}_p \in \mathbb{R}^p$, by means of the reduction procedure proposed by Gautier and Khalil,⁴ based on the energy model of the robot. Equation (4) can be then rewritten as

$$\boldsymbol{\tau} = \mathbf{D}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \boldsymbol{\theta}, \quad (5)$$

in which $\boldsymbol{\theta} := [\boldsymbol{\theta}_p^T \boldsymbol{\theta}_f^T]^T \in \mathbb{R}^{n_p}$, with $n_p = p + n_f$, and matrix $\mathbf{D}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$ is properly defined. Relation (5), which is linear with respect to $\boldsymbol{\theta}$, can be used for the estimation of $\boldsymbol{\theta}$ by collecting the values of $\boldsymbol{\tau}$, $\mathbf{q}$, $\dot{\mathbf{q}}$, and $\ddot{\mathbf{q}}$ (by measurements and/or reconstructed by

observers) at n_m time instants (with $n_m n > n_p$) during the execution of a task, thus obtaining the equation

$$\mathbf{y} = \mathbf{H}\boldsymbol{\theta} + \mathbf{v} \quad (6a)$$

with $\mathbf{y} := [\tau(t_1) \cdots \tau(t_{n_m})]^T$ and

$$\mathbf{H} := \begin{bmatrix} \mathbf{D}(\mathbf{q}(t_1), \dot{\mathbf{q}}(t_1), \ddot{\mathbf{q}}(t_1)) \\ \vdots \\ \mathbf{D}(\mathbf{q}(t_{n_m}), \dot{\mathbf{q}}(t_{n_m}), \ddot{\mathbf{q}}(t_{n_m})) \end{bmatrix}, \quad (6b)$$

where $\mathbf{v}$ is assumed to be the zero mean measurement noise vector, uncorrelated from $\boldsymbol{\theta}$, and $\mathbf{H}$ is the data matrix, assumed to be deterministic (the effects of uncertainty on the data matrix $\mathbf{H}$ are discussed in ref. 7). The base parameter vector $\boldsymbol{\theta}$ can be estimated from the regression equation (6a) using a least-squares algorithm as discussed in the following subsection.

We note that the energy-based calibration approach^{3,17} also leads to a standard least-squares identification problem, which may be cast in the form (6a). Therefore, all the subsequent developments can be applied equally to the energy-based calibration method.

2.1. Least-Squares Parameter Estimation

We recall the regression equation (6a),

$$\mathbf{y} = \mathbf{H}\boldsymbol{\theta} + \mathbf{v},$$

where $\mathbf{y} \in \mathbb{R}^{n_m}$ is the vector of measurements, $\boldsymbol{\theta} \in \mathbb{R}^{n_p}$ is the parameter vector to be identified, $\mathbf{H} \in \mathbb{R}^{n_m, n_p}$ is the regression matrix (assumed to be deterministic), and $\mathbf{v} \in \mathbb{R}^{n_m}$ is a zero mean random vector representing measurement errors. Vector $\mathbf{v}$ is assumed to be uncorrelated from $\boldsymbol{\theta}$, and the autocorrelation matrix $\mathbf{R}_v > 0$ of $\mathbf{v}$ is assumed to be known. *A priori* information about the unknown parameter vector is also included in the problem via the mean value $\bar{\boldsymbol{\theta}}$ of $\boldsymbol{\theta}$ (for instance, $\bar{\boldsymbol{\theta}}$ may be taken as the vector of nominal parameters) and the “confidence” on the nominal parameters, expressed by a given covariance matrix $\mathbf{R}_\theta > 0$. It is then well known¹⁸ that the linear least-mean-squares estimator of $\boldsymbol{\theta}$, given $\mathbf{y}$, is

$$\hat{\boldsymbol{\theta}} = \bar{\boldsymbol{\theta}} + (\mathbf{R}_\theta^{-1} + \mathbf{H}^T \mathbf{R}_v^{-1} \mathbf{H})^{-1} \mathbf{H}^T \mathbf{R}_v^{-1} (\mathbf{y} - \mathbf{H}\bar{\boldsymbol{\theta}}), \quad (7)$$

and the corresponding error covariance matrix is given by

$$\mathbf{P}_\theta = (\mathbf{R}_\theta^{-1} + \mathbf{H}^T \mathbf{R}_v^{-1} \mathbf{H})^{-1}. \quad (8)$$

The above formulas refer to the batch solution of the least-squares (LS) problem for the accumulated measurements $\mathbf{y}$. In the experimental tests, the actual estimate will be computed using the following recursive formulation of the LS algorithm (see ref. 18). Let $\mathbf{H}_i \in \mathbb{R}^{1, n_p}$ and $y_i \in \mathbb{R}$ represent the i th row of the regression matrix and the i th measurement, respectively. Let $\hat{\boldsymbol{\theta}}_i$ and $\mathbf{P}_{\theta, i}$ be defined as the estimate of the parameter vector $\boldsymbol{\theta}$, given the measurements up to i , and the relative covariance, respectively. The estimate can then be recursively updated using the recursions

$$\hat{\boldsymbol{\theta}}_{i+1} = \hat{\boldsymbol{\theta}}_i + \mathbf{K}_{i+1} (y_{i+1} - \mathbf{H}_{i+1} \hat{\boldsymbol{\theta}}_i), \quad \hat{\boldsymbol{\theta}}_0 = \bar{\boldsymbol{\theta}},$$

$$\mathbf{K}_{i+1} = \mathbf{P}_{\theta, i} \mathbf{H}_{i+1}^T (\mathbf{R}_v + \mathbf{H}_{i+1} \mathbf{P}_{\theta, i} \mathbf{H}_{i+1}^T)^{-1},$$

$$\mathbf{P}_{\theta, i+1} = \mathbf{P}_{\theta, i} - \mathbf{K}_{i+1} \mathbf{H}_{i+1} \mathbf{P}_{\theta, i}, \quad \mathbf{P}_{\theta, 0} = \mathbf{R}_\theta.$$

We recall that the solution obtained from the recursive algorithm for $i = n_m$ coincides with the batch solution, i.e., $\hat{\boldsymbol{\theta}}_{n_m} = \hat{\boldsymbol{\theta}}$; $\mathbf{P}_{\theta, n_m} = \mathbf{P}_\theta$. The recursive version is, therefore, a computationally efficient way to solve (7), especially when a large number of measurements is to be processed.

If *a priori* knowledge is neglected, then $\mathbf{R}_\theta = \infty$, and it follows from (8) that the “size” of the estimation error depends on the *normal equations matrix* $\boldsymbol{\Pi} = \mathbf{H}^T \mathbf{R}_v^{-1} \mathbf{H}$. A frequently used scalar measurer on $\boldsymbol{\Pi}$ is given by $J_d = \log \det(\boldsymbol{\Pi})$. Maximization of J_d with respect to the input excitation is referred to as *D optimality*,¹⁹ and it is desirable in order to get estimates with minimal uncertainty bounds. Another significant index is derived from the condition number of $\boldsymbol{\Pi}$: $J_k = \text{cond}(\boldsymbol{\Pi})$. A small J_k is desirable to minimize the bias of the estimate due to unmodeled dynamics errors (which are correlated with the regressors), to enhance the convergence rate of the algorithm, and to minimize the sensitivity of the solution to variations of the measurements vector $\mathbf{y}$ (see refs. 19 and 20).

3. TRAJECTORY OPTIMIZATION

The previously discussed optimality indices depend on the kinematic structure of the manipulator, which is fixed for a given type of manipulator, and on the

time history of the joint reference positions, velocities, and accelerations. Denoting with $\mathcal{Q}$ a class of continuous functions of the time variable $t \in \mathbb{R}$, defined over the interval $\mathcal{T} = [t_{\min}, t_{\max}]$, the problem of determining the most “exciting” reference trajectory $\mathbf{q}^*(t)$ with respect to the generic index J can be formulated as

$$\mathbf{q}^*(t) = \arg \min J(\mathbf{q}(t)) \quad (9a)$$

subject to

$$q_i(t) \in \mathcal{Q} \quad \forall i, \quad (9b)$$

$$|q_i(t)| \leq q_{i,\max} \quad \forall i, t, \quad (9c)$$

$$|\dot{q}_i(t)| \leq v_{i,\max} \quad \forall i, t, \quad (9d)$$

$$|\ddot{q}_i(t)| \leq \alpha_{i,\max} \quad \forall i, t, \quad (9e)$$

where $q_{i,\max}$, $v_{i,\max}$, and $\alpha_{i,\max}$ represent physical constraints on the i th joint position, velocity, and acceleration, respectively. The problem (9) is a non-linear functional optimization problem, for which different solutions have been presented in literature, as discussed in section 1.

The method that we propose herein is based on a parameterization of the class $\mathcal{Q}$ of joint position time functions. Formally, we consider that the class $\mathcal{Q}$ is of the form

$$\mathcal{Q} = \mathcal{Q}(\mathbf{p}), \quad \mathbf{p} \in \mathcal{P},$$

where $\mathbf{p}$ is a vector of real-valued parameters and $\mathcal{P}$ represents the set of physical constraints on the values of the parameters. The number of decision parameters is, therefore, independent of the number of time samples along the trajectory.

The two most popular optimality criteria, among the various ones proposed in the literature (see, e.g., refs. 7, 9, and 11) are considered: the condition number J_k of the regression matrix, as a measure of the disturbance influence on the parameter estimates and the scalar measure ($\log \det(\cdot)$) of the Fisher information matrix (i.e., the J_d index) to guarantee minimal uncertainty of the parameter estimates.

The resulting finite-dimensional optimization problem is then solved using the paradigm of genetic evolution. The genetic algorithm used in the solution is objective (fitness) function based and does not require the computation of gradients and Hessians. Also, the GA is oriented toward the identification of the global optimum, as is discussed in some detail in the following subsections.

3.1. Parameterization of the Joints Trajectories

Though the method outlined in this article is conceptually valid for any parametric family $\mathcal{Q}(\mathbf{p})$, we treat in detail the class of harmonic functions with a predefined and fixed number W of harmonic terms. This class of trajectories was also suggested in ref. 21 and, independently, in ref. 11.

Given W (typically a small integer, $W = 3, 4, \dots$), the reference trajectory for the j th joint is assumed to be

$$q_j(t) = \sum_{k=1}^W a_{j,k} \sin(\omega_{j,k} t), \quad (10)$$

where $a_{j,k}$ and $\omega_{j,k}$, $j = 1, \dots, n$, $k = 1, \dots, W$, are now the decision variables of the optimization problem (9). The number of decision variables $n_v = 2Wn$ is independent of the number of time samples along the trajectory. As

$$|q_j(t)| = \left| \sum_{k=1}^W a_{j,k} \sin(\omega_{j,k} t) \right| \leq \sum_{k=1}^W |a_{j,k}|, \quad (11)$$

the constraint (9c) may be conservatively substituted by

$$|a_{j,k}| \leq \frac{q_{j,\max}}{W} \quad \forall j, k.$$

Similarly, we obtain that

$$|\dot{q}_j(t)| = \left| \sum_{k=1}^W a_{j,k} \omega_{j,k} \cos(\omega_{j,k} t) \right| \leq \frac{q_{j,\max}}{W} \sum_{k=1}^W |\omega_{j,k}|$$

and the constraint (9d) is conservatively substituted by

$$|\omega_{j,k}| \leq \frac{v_{j,\max}}{q_{j,\max}} \quad \forall j, k.$$

Also

$$|\ddot{q}_j(t)| = \left| \sum_{k=1}^W a_{j,k} \omega_{j,k}^2 \sin(\omega_{j,k} t) \right| \leq \frac{q_{j,\max}}{W} \sum_{k=1}^W |\omega_{j,k}^2|$$

and the constraint (9e) is conservatively substituted by

$$|\omega_{j,k}| \leq \sqrt{\frac{\alpha_{j,\max}}{q_{j,\max}}} \quad \forall j, k.$$

The optimization problem (9) specializes now to

$$\underset{a_{j,k}, \omega_{j,k}; \forall j,k}{\text{minimize}} J(\mathbf{q}(t)) \quad (12a)$$

subject, $\forall j, k$, to the constraints

$$q_j(t) \in \mathcal{Q}, \quad (12b)$$

$$|a_{j,k}| \leq a_{j,\max} = \frac{q_{j,\max}}{W}, \quad (12c)$$

$$\omega_{j,k} \leq \omega_{j,\max} = \min \left\{ \frac{v_{j,\max}}{q_{j,\max}}, \sqrt{\frac{\alpha_{j,\max}}{q_{j,\max}}} \right\}. \quad (12d)$$

3.2. GA Coding

The optimization problem (12) is solved using a GA with binary coding.¹² In the language of genetic programming, each candidate solution is represented by a binary string of n_b digits which is referred to as an *individual*. The GA works with a *population* of N_p individuals and emulates the paradigm of natural selection and evolution. At each *generation*, some members of the population are selected for reproduction into the next generation. The choice of the individuals to be reproduced is guided by the *fitness* value of the individuals and by a random component. The reproduction step takes place by emulating the genetic operators of *crossover* and *mutation* on the *genotypes* (the bit strings) of the individuals. The evolution from one generation to the next is very likely to enhance the average and best fitness of the individuals in the population. After a certain number of generations, the GA converges to an individual which is the best suited for the given fitness function. The convergence of the GA to the global optimum is not, in general, guaranteed, but in practice the GA has proved to be extremely efficient in solving nonlinear and multimodal optimization problems.^{12,13} The GA turns out to be particularly efficient and robust in the global search of solutions, while usual gradient-based methods are efficient in the local solution search.

In the problem at hand, each decision variable $a_{j,k}$ and $\omega_{j,k}$ is coded as a binary integer $A_{j,k}$ and $\Omega_{j,k}$ of $n_{j,k}^{(a)}$ and $n_{j,k}^{(\omega)}$ digits, respectively, and an *individual* is represented by a binary string X obtained by connecting all the binary strings head to tail relative to the decision variables: $X = [A_{1,1}, \dots, A_{1,W}, A_{2,1}, \dots, A_{2,W}, \dots, A_{n,1}, \dots, A_{n,W}, \Omega_{1,1}, \dots, \Omega_{1,W}, \Omega_{2,1}, \dots, \Omega_{2,W}, \dots, \Omega_{n,1}, \dots, \Omega_{n,W}]$.

The constraints (12c) and (12d) are directly taken into account in the decoding phase of the genotype X . Prior to evaluation of the fitness function, each element of X is decoded to its real value in the manner

$$a_{j,k} = \left(\frac{[A_{j,k}]_{10}}{2^{n_{j,k}^{(a)}} - 1} - \frac{1}{2} \right) 2a_{j,\max} \quad \forall j,k, \quad (13)$$

$$\omega_{j,k} = \frac{[\Omega_{j,k}]_{10}}{2^{n_{j,k}^{(\omega)}} - 1} \omega_{j,\max} \quad \forall j,k, \quad (14)$$

where $[A_{j,k}]_{10}$ and $[\Omega_{j,k}]_{10}$ are the decimal values of the binary integers $A_{j,k}$ and $\Omega_{j,k}$. The bounds on the decision variables are, therefore, implicitly included in the decoding step, so that the resulting problem is unconstrained in the binary string X . The GA has been implemented in a MATLAB package developed by the authors and is available on request.

4. APPLICATION TO A PLANAR MANIPULATOR

The effectiveness of the method is validated by experimental tests for the dynamic calibration of the SCARA two-link planar manipulator, which is produced by IMI and is shown in Figure 1. The considered manipulator has two revolute joints equipped with brushless (direct-drive) motors produced by NSK (Nippon Seiko K. K., Tokyo, Japan), incorporating resolvers for position measurements. The control unit of the robot is constituted by a Pentium-90 PC with a TMS320C30 Digital Signal Processing (DSP) board, and the control algorithms are implemented in C. The joint coordinates are defined according to Denavit-Hartenberg notation and collected in vector $\mathbf{q} := [q_1 \ q_2]^T$, where q_i represents the angular position of joint i . The command torque vector is defined as $\boldsymbol{\tau} := [\tau_1 \ \tau_2]^T$, where τ_i is the torque applied to joint i , and the lengths of the links are, respectively, given by $l_1 = 0.359$ m and $l_2 = 0.24$ m.

4.1. Dynamic Model of the Planar Manipulator

The application of the reduction procedure developed by Gautier and Khalil⁴ leads to a base parameter set constituted by four inertial parameters given by the inertia moments of the links with respect to the z_0 axis (i.e., the axis perpendicular to the motion plane) and the first-order moments $m_2 s_{2x}$, $m_2 s_{2y}$ of

Figure 1. Photograph and schematic model of the IMI two-link manipulator.

the second link. The vector θ_p of the identifiable inertial parameters is then given by

$$\theta_p = [\Gamma_{1z} + m_2 l_1^2 \quad m_2 s_{2x} \quad m_2 s_{2y} \quad \Gamma_{2z}]^T.$$

Due to the small number of identifiable parameters for our 2 dof robot, no further reduction of the parameter vector is needed, while an approximate, simplified model (which neglects, if possible, the effects of some of the identifiable parameters) could be usefully employed for greater manipulators, as suggested in ref. 22.

In the considered case, only Coulomb and viscous friction are modeled, under the assumption that other components (e.g., hysteresis effects and/or stiction) can be considered as negligible. The friction torque acting on each joint is then described as

$$\tau_{f,i}(\dot{q}_i) = \sigma_{s,i} \operatorname{sgn}(\dot{q}_i) + \sigma_{v,i} \dot{q}_i, \quad (15)$$

where the parameters $\sigma_{s,i}$ and $\sigma_{v,i}$ represent Coulomb dynamic friction and the viscous friction coefficient, respectively. Note that this model can correctly describe friction only at “sufficiently high” velocities; otherwise, a specified model for stiction must be introduced. According to (15), the friction parameters of the robot can be collected in a vector $\theta_f \in \mathbb{R}^4$, which is defined as $\theta_f := [\sigma_{s,1}, \sigma_{v,1}, \sigma_{s,2}, \sigma_{v,2}]^T$. The complete parameter vector θ is then given by $\theta := [\theta_p^T \quad \theta_f^T]^T \in \mathbb{R}^8$. Following the Newton–Euler approach, the planar robot dynamic

model is expressed as in (5), where

$$\begin{aligned} D_{1,1} &= \ddot{q}_1, \\ D_{1,2} &= l_1(c_2(2\ddot{q}_1 + \ddot{q}_2) - s_2\dot{q}_2(2\dot{q}_1 + \dot{q}_2)), \\ D_{1,3} &= l_1(-s_2(2\ddot{q}_1 + \ddot{q}_2) - c_2\dot{q}_2(2\dot{q}_1 + \dot{q}_2)), \\ D_{1,4} &= \ddot{q}_1 + \ddot{q}_2, \\ D_{1,5} &= \operatorname{sgn}(\dot{q}_1), \\ D_{1,6} &= \dot{q}_1, \\ D_{1,7} &= D_{1,8} = 0, \\ D_{2,1} &= 0, \\ D_{2,2} &= l_1(c_2\ddot{q}_1 + s_2\dot{q}_1^2), \\ D_{2,3} &= l_1(-s_2\ddot{q}_1 + c_2\dot{q}_1^2), \\ D_{2,4} &= \ddot{q}_1 + \ddot{q}_2, \\ D_{2,5} &= D_{2,6} = 0, \\ D_{2,7} &= \operatorname{sgn}(\dot{q}_2), \\ D_{2,8} &= \dot{q}_2 \end{aligned} \quad (16)$$

with $s_i := \sin q_i$ and $c_i := \cos q_i$.

4.2. Robot Control and Physical Constraints

The control of the two-link manipulator is performed by an independent joint feedback control law, which is implemented by inserting an observer for the estimation of the joint velocities, because only position measurements are available. Unsatisfactory results have been obtained by simply deriv-

ing the joint velocities at discrete time from the position measurements. The high-gain observer developed in ref. 23 were chosen after a comparison among different solutions proposed in literature (e.g., refs. 24–27) because it seems to offer, in our case, a good compromise in terms of tracking error and quality of the reconstructed signals (with reference in particular to velocities, which must be used both for the control and for the parameter identification procedure). The implemented controller–observer scheme is then described by

$$\hat{\mathbf{x}}_1 = \hat{\mathbf{x}}_2 + \frac{1}{\varepsilon} \mathbf{H}_p (\mathbf{q} - \hat{\mathbf{x}}_1), \quad (17a)$$

$$\dot{\hat{\mathbf{x}}}_2 = \frac{1}{\varepsilon^2} \mathbf{H}_v (\mathbf{q} - \hat{\mathbf{x}}_1), \quad (17b)$$

$$\boldsymbol{\tau} = \mathbf{K}_p (\mathbf{q}_d - \mathbf{q}) + \mathbf{K}_d (\dot{\mathbf{q}}_d - \hat{\mathbf{x}}_2), \quad (17c)$$

where the observer state variables $\hat{\mathbf{x}}_1$ and $\hat{\mathbf{x}}_2$, respectively, denote the dynamic estimates of the joint positions and velocities, $\mathbf{q}$ is the vector of the joint positions measured by the resolvers incorporated in the motors, $\mathbf{q}_d(t)$ and $\dot{\mathbf{q}}_d(t)$ are the assigned reference position and velocity vectors, obtained by the optimization procedure, and $\boldsymbol{\tau}$ is the vector of the applied torques in (5). The observer gain matrices $\mathbf{H}_p = \text{diag}(h_{p,i})$ and $\mathbf{H}_v = \text{diag}(h_{v,i})$ are chosen so that the spectra of the characteristic polynomials $p_i(\lambda) = \lambda^2 + h_{p,i}\lambda + h_{v,i}$, $i = 1, 2$, are in the open left half-plane, whereas ε is a small positive parameter. The controller gain matrices $\mathbf{K}_p$ and $\mathbf{K}_d$ are diagonal, positive definite matrices that are selected to satisfy the desired closed-loop specifications. In the tests carried out, the controller–observer scheme is designed at continuous time to obtain a closed-loop system with a bandwidth of about 60 rad/s and a maximum peak ≤ 4 dB, and then implemented at discrete time with sampling time $T = 1$ ms (sample rate $\omega_c = 2\pi/T \approx 6280$ rad/s). The selected observer gains are $\mathbf{H}_p = \text{diag}(7, 6.2)$, $\mathbf{H}_v = \text{diag}(2, 4)$, and $\varepsilon = 0.002$.

The reference joint trajectories are assumed to belong to the class $\mathcal{C}$ of harmonic functions defined in (10), with the physical bounds (9c)–(9e) on the angular positions, velocities, and accelerations given by $q_{\max} = 1.5$ rad, $v_{\max} = 8$ rad/s, and $\alpha_{\max} = 5$ rad/s² for both joints. From (12d) it follows that the frequency content of the reference signals is always below $\omega_{\max} \approx 1.825$ rad/s for all joints.

4.3. Data Acquisition and Processing

The main problems to be solved for the acquisition of the data that are required for the calibration procedures can be summarized as follows:

1. The need to guarantee a good quality of the measured or observed signals to be used for the parameter estimation.
2. Computation of the signals not directly available: joint accelerations and command torques (because they are not directly measurable in our case).
3. Characterization of the torque measurement errors to be used in the estimation algorithm.

The joint velocities reconstructed by the observer [(17a) and (17b)] and directly used for the robot control need to be filtered before the computation of the acceleration signal and the parameter estimation. This filtering should be performed off-line by means of a noncausal zero phase shift filter to avoid distortion in the dynamic regressor, as discussed in ref. 6. However, due to our hardware limitations and conflicting interrupt problems during the motion of the robot, it was not possible for us to acquire samples at the control rate ω_c and then filter and decimate the data off-line. Therefore, we preferred to filter the velocity signal *on-line* using the following discrete time filter ($T = 2\pi/\omega_c = 1$ ms) that has cutoff frequency $\omega_f \approx 14$ rad/s,

$$\begin{aligned} \dot{\mathbf{q}}_{\text{fil}}(kT) = & \alpha_1 \hat{\mathbf{x}}_2(kT) + \alpha_2 \dot{\mathbf{q}}_{\text{fil}}((k-1)T) \\ & + \alpha_3 \dot{\mathbf{q}}_{\text{fil}}((k-2)T), \end{aligned}$$

with $\alpha_2 = 1.95550858$, $\alpha_3 = -0.95599748$, and $\alpha_1 = 1 - \alpha_2 - \alpha_3$. Note that this filter introduces negligible phase shift for the reference signals used, which have frequency content below $\omega_{\max}$. The joint accelerations are then computed on-line at the control rate ω_c from the filtered velocity by means of a central difference algorithm, as discussed in refs. 6 and 28.

The torques are reconstructed by their expression (17c) and then filtered on-line at the control rate, using the same discrete time filter employed for the velocities. The filtered torque vector is then given by

$$\begin{aligned} \boldsymbol{\tau}_{\text{fil}}(kT) = & \alpha_1 \boldsymbol{\tau}(kT) + \alpha_2 \boldsymbol{\tau}_{\text{fil}}((k-1)T) \\ & + \alpha_3 \boldsymbol{\tau}_{\text{fil}}((k-2)T). \end{aligned}$$

For each reference trajectory, the following procedure is used to estimate the mean-square error (sample variance) of the torque measurements. The given trajectory is repeated by the manipulator for $M=20$ times and n_m data samples are collected over each repetition of the trajectory. The sample variance of the torque measurements for the i th joint is computed as

$$\sigma_i^2 = \frac{1}{n_m(M-1)} \sum_{k=1}^{n_m} \sum_{j=1}^M (\tau_{ij}(k) - \bar{\tau}_i(k))^2,$$

where $\tau_{ij}(k)$ represents the k th measurement of the torque at joint i during the j th trajectory repetition and $\bar{\tau}_i(k)$ represents the average over the M repetitions of the value of the i th joint torque at the k th sample.

5. EXPERIMENTAL PARAMETER ESTIMATION

Optimal excitation trajectories are computed according to the two criteria previously discussed. The reference trajectories so found are then executed by the manipulator and a recursive LS algorithm is applied to the acquired data to estimate the inertial and friction parameters.

The LS normal equation matrix is, in this case, given by $\Pi = \mathbf{H}^T \mathbf{R}_v^{-1} \mathbf{H}$, where $\mathbf{H}$ is given by (6) and $\mathbf{R}_v = \text{diag}\{\sigma_1^2, \sigma_2^2\}$ is the diagonal covariance matrix relative to the torque measurements. Whereas the estimated sample variances on the torque measurements depend on the reference trajectory, it was decided to compute the optimal reference trajectory assuming $\mathbf{R}_v = \mathbf{I}$ and then to estimate the measurement variances on the computed trajectories for the application of the recursive parameter estimation algorithm.

It was determined from simulations that parameter resolution (i.e., the number of binary digits used to encode each of the decision variable) is not an issue in the problem at hand; the choice of $n_{j,k}^{(a)} = n_{j,k}^{(\omega)} = 4 \forall j, k$ is satisfactory for our purposes. Numerical simulations also showed that a number of harmonics as low as $W=3$ is sufficient to determine a good trajectory, because larger values of W gave no significant improvement on the optimal objective. The number of variables for the optimization problem is, in this case, $n_v = 2Wn = 12$, where $n = 2$ is the number of joints of the manipulator and each solution string (genotype) is represented by the

GA as a $n_b = 4n_v = 48$ digits binary string X . Notice that the number n_v of decision variables is independent of the number of time samples n_m along the trajectory, while it is proportional to the considered number W of harmonic terms. The length of the genotypic string X is proportional to n_v and to the selected resolution (number of binary digits) to which the variables are encoded. The fitness function is, in this case, given by $-J_k$ or by J_d , because the GA is traditionally a maximization algorithm.

5.1. Estimation Using J_k Optimal Trajectories

We first consider the optimality criterion based on the minimization of the condition number $J_k(\Pi)$. The resulting three-harmonic trajectory is shown in Figure 2. The minimum value of J_k ($J_k = 13.34$) is achieved by the GA after less than 20 generations, as shown in Figure 3. The order of the computational time is 5 min on an “old” Intel 486/66 PC.

The computed optimal trajectory is generated at 1 ms and passed to the manipulator that repeats it 20 times. The torque measurement variances are estimated over the 20 repetitions as $\sigma_1^2 = 3.069 \text{ N}^2 \cdot \text{m}^2$ and $\sigma_2^2 = 0.157 \text{ N}^2 \cdot \text{m}^2$. The LS algorithm described in Section 2.1 is used for the off-line parameter estimation, collecting $n_m = 80$ equally spaced samples for each of the first six trajectory repetitions. The evolution of the estimates is shown in

Figure 2. Optimal joint reference trajectories using index J_k . Solid line, first joint; dashed line, second joint.

Figure 3. GA evolution. Crosses represent fitness values ($= -J_k$) of the best individual in the generation. The solid line represents the average fitness over the population members at each generation.

Figure 4, and the final obtained estimates, which are consistent with the physical meaning of the parameters, are given by

$$\hat{\theta} = [3.923; 0.808; 0.328; 0.141; 2.61; 7.343; 1.460; 0.787]^T,$$

Figure 4. Evolution of the recursive LS parameter estimates.

Figure 5. Torque estimation residuals for the first (top) and second joints (bottom); six data batches.

whereas the diagonal elements of the obtained estimation covariance matrix are

$$\sigma_{\theta} = [0.0054; 0.0028; 0.0032; 0.0004; 0.0225; 0.0473; 0.0011; 0.0025]^T.$$

The values obtained from the manufacturer data for the first four parameters are

$$\theta_{\text{nom},1} = 3.689 \text{ kg} \cdot \text{m}^2,$$

$$\theta_{\text{nom},2} = 0.97 \text{ kg} \cdot \text{m},$$

$$\theta_{\text{nom},3} = 0 \text{ kg} \cdot \text{m},$$

$$\theta_{\text{nom},4} = 0.275 \text{ kg} \cdot \text{m}^2,$$

but we remark that the goal of the applied LS method is to determine parameters that minimize the torque residual on the basis of the assumed dynamic and friction model, without regard to the physical meaning of each single parameter.

The estimated parameters are used to compare the reconstructed torques with the measured ones. Figure 5 shows the torque estimation residuals, while the measured and reconstructed torques are compared in Figure 6. The root-mean-squared (rms) estimation residuals (difference between the measured and reconstructed torques) are computed over the estimation set as $\varepsilon_1 = 1.34 \text{ N} \cdot \text{m}$ and $\varepsilon_2 = 0.24 \text{ N} \cdot \text{m}$; the relative estimation errors, defined as the ratio between the rms estimation error and the rms value of the measured torque, are $\varepsilon_{r,1} = 15\%$ and $\varepsilon_{r,2} = 11.7\%$.

Figure 6. Comparison of measured (dashed) and reconstructed (solid) torques: first (top) and second joints (bottom); two data batches.

Estimate Validation

The obtained estimate is tested against two validation data sets that differ from the data used for estimation. As a first validation set, we consider another batch of six repetitions of the optimal reference trajectory. The rms values of the residuals are in this case $\varepsilon_1 = 1.38 \text{ N} \cdot \text{m}$ and $\varepsilon_2 = 0.23 \text{ N} \cdot \text{m}$, while the relative rms errors are $\varepsilon_{r,1} = 16.5\%$ and $\varepsilon_{r,2} = 12.2\%$. The second validation set is constructed upon measurements on six repetitions of the test trajectory shown in Figure 7. The resulting residual errors are $\varepsilon_1 = 1.3 \text{ N} \cdot \text{m}$, $\varepsilon_2 = 0.31 \text{ N} \cdot \text{m}$, $\varepsilon_{r,1} = 17.2\%$, and $\varepsilon_{r,2} = 14.6\%$, and the torque reconstruction residuals are shown in Figure 8.

5.2. Estimation Using J_d Optimal Trajectories

Computation of the optimal three-harmonic reference trajectory also was performed with respect to the second optimality criterion, based on the index $J_d = \log \det(\mathbf{\Pi})$. The maximum obtained value is, in this case, $J_d = 17.9$, corresponding to a $\det(\mathbf{\Pi})$ value of about 8×10^{17} . A plot of the reference trajectory is reported in Figure 9. The torque measurement variances were estimated as previously explained, yielding the values $\sigma_1^2 = 113.77 \text{ N}^2 \cdot \text{m}^2$ and $\sigma_2^2 = 0.39 \text{ N}^2 \cdot \text{m}^2$. By applying the recursive LS algorithm, the parameter estimates

$$\hat{\theta} = [3.445; 0.698; 0.012; 0.107; 0.500; 7.884; 1.023; 0.883]^T$$

Figure 7. Validation trajectory. Solid line, first joint; dashed line, second joint.

were obtained with the following values for the diagonal elements of the covariance matrix:

$$\sigma_{\theta} = [0.0339; 0.0010; 0.0011; 0.0000; 1.0446; 0.5603; 0.0035; 0.0019]^T.$$

The rms torque reconstruction residuals on the estimation data set are $\varepsilon_1 = 2.98 \text{ N} \cdot \text{m}$, $\varepsilon_2 = 0.24 \text{ N} \cdot \text{m}$, $\varepsilon_{r,1} = 20.3\%$, and $\varepsilon_{r,2} = 9.3\%$. Figure 10 shows the

Figure 8. Torque reconstruction residuals for the validation trajectory: first (top) and second joints (bottom); six data batches.

Figure 9. J_d -optimal reference trajectories. Solid line, first joint; dashed line, second joint.

torque reconstruction residuals, while the measured and reconstructed torques are compared in Figure 11.

Estimate Validation

Using a first validation data set given by another batch of six repetitions of the optimal reference trajectory, the following rms residuals were obtained: $\varepsilon_1 = 3.65 \text{ N}\cdot\text{m}$, $\varepsilon_2 = 0.31 \text{ N}\cdot\text{m}$, $\varepsilon_{r,1} = 26.6\%$, and $\varepsilon_{r,2} = 13\%$. The larger error on the reconstruction of the torque at the first joint is caused by the large mean-square error in the measurements of this

Figure 10. Torque reconstruction residuals: first (top) and second joints (bottom); six data batches.

Figure 11. Comparison of measured (dashed) and reconstructed (solid) torques: first (top) and second joints (bottom); two data batches.

torque. Validation on the test trajectory of Figure 7 yields $\varepsilon_1 = 1.78 \text{ N}\cdot\text{m}$, $\varepsilon_2 = 0.4 \text{ N}\cdot\text{m}$, $\varepsilon_{r,1} = 23.8\%$, and $\varepsilon_{r,2} = 19\%$.

5.3. Remarks

Trajectories optimized with respect to the J_k index provided parameter estimates with the lowest relative rms prediction errors. The obtained parameter estimates allowed reconstruction of the input torques (that may be used for feedforward compensation) with about 20% rms error on both joints. J_d -optimal trajectories tend to move the manipulator faster and result in higher measurement variance; therefore, the relative parameter estimates yield higher rms errors in the reconstructed torques.

Some of the parameter estimates obtained in the two cases (i.e., when tracking a J_k - or a J_d -optimal trajectory) are quite different. In particular, the estimates of the static friction $\sigma_{s,1}$ of the first joint are, respectively, given by 2.61 and 0.5 $\text{N}\cdot\text{m}$. This fact may be partially due to the differences between the imposed trajectories (see Figures 2 and 9): Because the first one is slower than the second one, the effect of the static friction is stronger in the first case, so that greater values of $\sigma_{s,i}$ are estimated (compare also the estimates of $\sigma_{s,2}$, 1.460 and 1.023 $\text{N}\cdot\text{m}$). Therefore, a more accurate model of the friction phenomena may be useful if direct friction compensation is required for control purposes. Notice also that the value of the static friction parameter for the first joint for the J_d -based experiment has little sig-

nificance, since the relative variance is high compared to the nominal value of the parameter.

Finally, it must be stressed that the parameter estimates obtained using standard LS identification may sometimes lead to physically unfeasible numerical values of the parameters (i.e., negative moments of inertia, etc.) because no specific constraints are imposed in the optimization problem. Future work will be devoted to study the effect of the inclusion of physical constraints on the torque reconstruction residuals.

6. CONCLUSIONS

In this article, an efficient method for the determination of “exciting” trajectories for robot inertial parameter estimation has been presented. The method is based on harmonic parameterization of the joint time functions and minimization of different suitable measures on the LS normal equation matrix performed by means of a binary coded genetic algorithm. The proposed framework may be applied to both Newton–Euler and energy-based parameter estimation models.

Numerical simulations were carried out using the Newton–Euler model and showed that good trajectories can be found using low parameter resolution and a small number of harmonic terms. The computed trajectories respect the frequency bandwidth constraints and the physical limits on joint position, velocity, and acceleration.

The results of the experimental identification of the inertial and friction parameters of a SCARA two-link manipulator, using the optimal input excitations, also were presented and discussed.

REFERENCES

1. C.G. Atkeson, C.H. An, and J.M. Hollerbach, Estimation of inertial parameters of manipulator loads and links, *Int J Robotics Res* 5 (1986), 101–119.
2. C.P. Neuman and P.K. Khosla, Identification of robot dynamics: An application of recursive estimation, *Proc 4th Yale Workshop on Applications of Adaptive System Theory*, Yale University, 1985, pp. 42–49.
3. S.-Y. Sheu and M.W. Walker, Estimating the essential parameter space of the robot manipulator dynamics, *Proc 28th Conf on Decision and Control*, 1989, pp. 2135–2140.
4. M. Gautier and W. Khalil, Direct calculation of minimum set of inertial parameters of serial robots, *IEEE Trans Robotics Automation* 6 (1990), 368–373.
5. B. Bona and M. Indri, Experiments on the dynamic calibration of a planar manipulator, *Proc 27th Int Symp on Industrial Robots*, Milan, Italy, 1996, pp. 117–122.
6. M. Gautier, A comparison of filtered models for dynamic identification of robots, *Proc 35th CDC*, 1996, pp. 875–880.
7. B. Armstrong, On finding exciting trajectories for identification experiments involving systems with non-linear dynamics, *Proc IEEE Int Conf on Robotics and Automation*, 1987, pp. 1131–1139.
8. F. Caccavale and P. Chiacchio, Identification of dynamic parameters and feedforward control for a conventional industrial manipulator, *Control Eng Practice* 2 (1994), 1039–1050.
9. M. Gautier and W. Khalil, Exciting trajectories for the identification of base inertial parameters of robots, *Int J Robotics Res* 11 (1992), 362–375.
10. G.W. Van der Linden and A.J.J. Van der Weiden, Practical rigid body parameter estimation, *Proc 4th IFAC Symp on Robot Control*, 1994, pp. 631–636.
11. J. Swevers, C. Ganseman, D.B. Tukul, J. de Schutter, H. Van Brussel, Optimal robot excitation and identification, *IEEE Trans Robotics Automation* 13 (1997), 730–740.
12. D.E. Goldberg, *Genetic algorithms in search, optimization, and machine learning*, Addison-Wesley, Reading, MA, 1989.
13. Z. Michalewicz, *Genetic algorithms + data structures = evolution programs*, Springer-Verlag, New York, 1992.
14. M.W. Chen and A.M.S. Zalzal, Dynamic modelling and genetic-based trajectory generation for non-holonomic mobile manipulators, *Control Eng Practice* 5 (1997), 39–48.
15. H. Zhuang, J. Wu, and W. Huang, Optimal planning of robot calibration experiments by genetic algorithms, *J Robotic Syst* 14 (1997), 741–752.
16. B. Armstrong-Hélouvy, P.E. Dupont, and C. Canudas de Wit, A survey of models, analysis tools and compensation methods for the control of machines with friction, *Automatica* 30 (1994), 1083–1138.
17. S.-Y. Sheu and M.W. Walker, Basis sets for manipulator inertial parameters, *Proc IEEE Int Conf on Robotics and Automation*, 1989, pp. 1517–1522.
18. T. Kailath, *Lectures on Wiener and Kalman filtering*, Springer-Verlag, New York, 1981.
19. L. Ljung, *System identification, theory for the user*, Prentice-Hall, Englewood Cliffs, NJ, 1987.
20. G.H. Golub and C.F. Van Loan, *Matrix computations*, The Johns Hopkins University Press, Baltimore, 1989.
21. G. Calafiore and M. Indri, Experiment design for robot dynamic calibration, *Proc 1998 IEEE Int Conf on Robotics and Automation*, 1998, pp. 3303–3309.
22. A. Visioli and G. Legnani, Experiments on model identification and control of an industrial robot manipulator, *Proc 6th IFAC Symp on Robot Control (SYROCO'00)*, 2000, Vol. 1, pp. 283–288.
23. S. Nicosia, A. Tornambè, and P. Valigi, Experimental results in state estimation of industrial robots, *Proc 29th IEEE Conf on Decision and Control* 1990, pp. 360–365.

24. H. Berghuis and H. Nijmeijer, Robust control of robots via linear estimated state feedback, *IEEE Trans Automat Control* 39 (1994), 2159–2162.
25. H. Berghuis and H. Nijmeijer, Global regulation of robots using only position measurements, *Syst Control Lett* 21 (1993), 289–293.
26. R. Kelly, A simple set-point robot controller by using only position measurements, *Proc 12th IFAC World Congress*, 1993, Vol. 6, pp. 173–176.
27. M.F. Khelfi, M. Zasadzinski, H. Rafaralahy, E. Richard, and M. Darouach, Reduced-order observer-based point-to-point and trajectory controllers for robot manipulators, *Control Eng Practice* 4 (1996), 991–1000.
28. P.R. Belanger, X. Meng, and P. Dobrovolny, Estimation of angular velocity and acceleration from equally-spaced shaft encoder measurements, *IEEE Int Conf on Robotics and Automation*, Nice, 1992.
29. B. Bona and M. Indri, Identification of the dynamic parameters of a planar manipulator: Simulations and experiments, *Proc 2nd World Automation Congress (WAC 96)*, Montpellier, France, 1996.