Robotica (1999) volume 17, pp. 427–435. Printed in the United Kingdom © 1999 Cambridge University Press

A systematic procedure for the identification of dynamic parameters of robot manipulators

G. Antonelli, F. Caccavale and P. Chiacchio

Dipartimento di Informatica e Sistemistica, Università degli Studi di Napoli Federico II, Via Claudio 21, 80125 Napoli (Italy)

(Received in Final Form: November 9, 1998)

SUMMARY

In this paper a complete and systematic procedure for the identification of the dynamic parameters of rigid robot manipulators is presented. Starting from the basic results on the subject present in the literature and on a new technique to find exciting trajectories for the estimation, the procedure is developed. A set of algorithms is provided for the implementation of the various steps of the procedure for a generic open-chain structure. The algorithms have been coded in the popular Matlab/Maple environment and the procedure has been tested in a practical case study to identify the dynamic parameters of a six-degree-of-freedom conventional industrial robot.

KEYWORDS: Dynamic parameters; Rigid robots; Open-chain structure; Identification algorithms

1. INTRODUCTION

Advanced control techniques for robot manipulators have become more and more affordable thanks to the increasing power of computing resources and their dramatic cost reduction. This is the case of model-based control strategies for robot manipulators whose potential and reliability have been confirmed even for industrial setups.[1,2]

However, the design and the implementation of a model-based controller requires an accurate knowledge of manipulator's dynamic model. The structure of the equations of motion for an open-chain rigid manipulator can be obtained by using classical mechanics concepts (see, e.g. reference 3). On the other hand, the parameters appearing in the equations are often unknown or only approximately known via CAD estimates. In order to achieve a more accurate knowledge of the dynamic parameters, experimental identification techniques can be used,[4,5] where the estimates are obtained from direct measurements collected during robot's motion.[2]

Although several theoretical results regarding the identification of robot manipulators are available in the literature, the research still lacks in proposing an effective and reliable procedure providing the user with guidelines for designing and performing identification experiments for a generic industrial manipulator. This paper is aimed at setting up a complete and systematic procedure for the identification of dynamic parameters of open-chain rigid manipulators. The goal has been achieved by using some of the main research results already present in the literature applied in an industrial-oriented context.

Browsing the literature on the subject, it can be recognized that the identification of dynamic parameters is articulated in a few main points:

a. First, the dynamic model of the manipulator is derived in symbolic form.
b. The minimal set of dynamic parameters to be identified (i.e. the base parameters) is computed.
c. Eventually, the dynamic model could be simplified either by neglecting the contribution of certain parameters to the joint torques, or by exploiting some topological properties of the manipulator's structure.
d. Trajectories to be executed by the manipulator have to be found in order to collect the best – in a sense to be further specified – set of data for the estimation algorithm.
e. The identified values of the dynamic parameters are obtained by applying a suitable estimation algorithm.
f. The results of the estimation are experimentally verified.

In the paper the different steps in which the procedure is articulated are detailed. Also, a new technique to find suitably exciting trajectories (see point d above) is proposed to reduce the computational burden of the related optimization problem, while preserving a satisfactory degree of excitation of the resulting set of data. Moreover, two techniques are proposed for reducing the computational load of the model.

A set of software modules is designed and developed in the popular Matlab/Maple environment aimed at implementing the different steps of the identification procedure.

Finally, the procedure is tested in an practical case study for the identification of the dynamic parameters for a six-degree-of-freedom conventional industrial robot. Remarkably, only one experiment has been necessary to achieve accurate estimates of the dynamic parameters; hence, the procedure is effective and only a limited effort is required to the user. A second experiment has been performed to validate the estimation; the obtained results demonstrate the reliability of the proposed procedure as well as its applicability in an industrial-oriented context.

2. THE IDENTIFICATION PROCEDURE

In this section the fundamental theory of identification for open-chain manipulators is reviewed. Also, a new approach

https://doi.org/10.1017/S026357479900140X Published online by Cambridge University Press

for finding exciting trajectories for the estimation is proposed.

2.1. Derivation of the dynamic model

Consider an open-chain manipulator composed of n rigid links connected through n joints (rotational and/or prismatic), eventually via rigid gears. The equations of motion of such a system can be obtained via the recursive Newton-Euler formulation (see, e.g. reference 3). Assume that the Denavit-Hartenberg notation is used to select the frame i attached to the corresponding link, and denote by $\boldsymbol{R}_k^j$ the rotation matrix representing the orientation of frame k with respect to frame j. In the case of rotational joints, the equations of motion for the frame i attached to the ith link, whose origin is on the axis of joint $i+1$, are

$$\boldsymbol{\omega}_i^i = (\boldsymbol{R}_i^{i-1})^T(\boldsymbol{\omega}_{i-1}^{i-1} + \dot{q}_i \mathbf{z}_0) \tag{1}$$

$$\boldsymbol{\omega}_i^i = (\boldsymbol{R}_i^{i-1})^T(\dot{\boldsymbol{\omega}}_{i-1}^{i-1} + \dot{q}_i \boldsymbol{\omega}_{i-1}^{i-1} \times \mathbf{z}_0 + \ddot{q}_i \mathbf{z}_0) \tag{2}$$

$$\boldsymbol{v}_i^i = (\boldsymbol{R}_i^{i-1})^T(\dot{\boldsymbol{v}}_{i-1}^{i-1} + \dot{\boldsymbol{\omega}}_i^i \times \mathbf{r}_{i-1,i}^i + \boldsymbol{\omega}_i^i \times (\boldsymbol{\omega}_i^i \times \mathbf{r}_{i-1,i}^i)) \tag{3}$$

$$\boldsymbol{f}_i^i = \boldsymbol{R}_{i+1}^i \boldsymbol{f}_{i+1}^{i+1} + m_i \dot{\boldsymbol{v}}_i^i + \dot{\boldsymbol{\omega}}_i^i \times m_i \mathbf{r}_{i,ci}^i + \boldsymbol{\omega}_i^i \times (\boldsymbol{\omega}_i^i \times m_i \mathbf{r}_{i,ci}^i) \tag{4}$$

$$\boldsymbol{\mu}_i^i = \boldsymbol{R}_{i+1}^i \boldsymbol{\mu}_{i+1}^{i+1} + \mathbf{r}_{i-1,i}^i \times \boldsymbol{f}_i^i + m_i \mathbf{r}_{i,ci}^i \times \dot{\boldsymbol{v}}_i^i + \boldsymbol{I}_i \dot{\boldsymbol{\omega}}_i^i + \boldsymbol{\omega}_i^i \times \boldsymbol{I}_i \boldsymbol{\omega}_i^i \tag{5}$$

$$\tau_i = (\boldsymbol{R}_{i-1}^i \boldsymbol{\mu}_i^i)^T \mathbf{z}_0 + I_{mot,i} \ddot{q}_i + \tau_{f,i} \tag{6}$$

where the superscript denotes the frame in which a vector is expressed. For brevity the equations in the case of prismatic joints are not reported (see reference 3 for their expressions).

In the above equations $\boldsymbol{v}_i$ is the linear velocity of the origin of frame i, $\boldsymbol{\omega}_i$ is the angular velocity of frame i, q_i is the joint variable, $\mathbf{z}_0 = [0\ 0\ 1]^T$, $\boldsymbol{f}_i$ is the force applied at the origin of frame i, $\boldsymbol{\mu}_i$ is the moment applied to the link and τ_i is the force (torque) applied to the joint axis; m_i is the link mass, $\mathbf{r}_{i,ci}$ is the vector position from origin of frame i to the center of mass of link i, $\mathbf{r}_{i-1,i}$ is the vector position from origin of frame $i-1$ to the origin of frame i, $\boldsymbol{I}_i$ is the matrix expressing the inertia tensor with respect to frame i and $I_{mot,i}$ is the inertia of the ith actuator as seen at the joint axis. As for the friction torque $\tau_{f,i}$, several models have been proposed in the literature (see reference 6 for a thorough review); for identification purposes the following model has been adopted

$$\tau_{f,i} = b_{s,i}\ \mathrm{sign}(\dot{q}_i) + b_{v,i}\ \dot{q}_i, \quad |\dot{q}_i| \geq \dot{q}_{s,i} \tag{7}$$

where $\dot{q}_{s,i}$ is a suitable velocity threshold, $b_{s,i}$ is the static friction coefficient, and $b_{v,i}$ is the viscous friction coefficient. When $|\dot{q}_i| < \dot{q}_{s,i}$ the friction torque has been considered null since it is difficult to derive a simple model for the stiction effects. It is worth noticing that the above friction model is linear in the parameters $b_{s,i}$ and $b_{v,i}$.

From (4), (5), (6) and (7), it can be seen that the equations of motion are linear in the vector of the dynamic parameters, i.e. they can be rewritten as

$$\boldsymbol{\tau} = \boldsymbol{P}(\boldsymbol{q}, \dot{\boldsymbol{q}}, \ddot{\boldsymbol{q}})\boldsymbol{\gamma} \tag{8}$$

where the $(n \times 13n)$ upper triangular matrix $\boldsymbol{P}(\boldsymbol{q}, \dot{\boldsymbol{q}}, \ddot{\boldsymbol{q}})$ depends only on kinematic quantities, and $\boldsymbol{\gamma} \in \mathbb{R}^{13n}$ is the vector of dynamic parameters, which, for each link, includes:

- the mass m_i (1 unknown);
- the first-order moment $m_i \mathbf{r}_{i,ci}^i$ (3 unknowns);
- the symmetric inertia matrix $\boldsymbol{I}_i$ expressed in frame i (6 unknowns);
- the actuator inertia $I_{mot,i}$ (1 unknown);
- the friction coefficients $b_{s,i}$ and $b_{v,i}$ (2 unknowns).

2.2. Base parameters

For an open-chain manipulator only a restricted degree of relative motion between the links is allowed, i.e. the actuators can apply torques only around (along) the joint axes. This leads to recognize that not all the parameters in $\boldsymbol{\gamma}$ give contribution to the joint torques. Indeed, the dynamic parameters in $\boldsymbol{\gamma}$ can be regrouped into three different categories:

- absolutely identifiable parameters
- identifiable in linear combination with other parameters;
- unidentifiable parameters.

Absolutely identifiable parameters correspond to linear independent columns of $\boldsymbol{P}$. The second class of parameters gives contribution to the joint torques only in linear combination with other parameters; this means that the corresponding columns of $\boldsymbol{P}$ can be replaced by a linear combination of themselves. Unidentifiable parameters do not give contribution to the torque applied at the joints (e.g. the mass of the base link of an anthropomorphic manipulator), and thus they can be discarded from the model.

Therefore, the vector $\boldsymbol{\gamma}$ can be replaced by a vector of base parameters $\boldsymbol{\theta} \in \mathbb{R}^p$ with $p < 13n$, and the matrix $\boldsymbol{P}$ can be replaced by the corresponding $(n \times p)$ reduced matrix $\boldsymbol{W}$

$$\boldsymbol{\tau} = \boldsymbol{W}(\boldsymbol{q}, \dot{\boldsymbol{q}}, \ddot{\boldsymbol{q}})\boldsymbol{\theta} \tag{9}$$

Since the objective is not the estimation of the single parameters, but the minimization of the error between the computational model of the manipulator and its real behaviour, the reduction of $\boldsymbol{\gamma}$ to a minimal set of parameters is not restrictive.

In order to obtain the set of base parameters, a procedure based on the symbolic manipulation of eq. (8) is proposed in reference 7. However, such a method requires a case-by-case analysis to handle some particular situations. In order to build an automatic procedure for the derivation of the set of base parameters, a numerical method based on the Singular Value Decomposition (SVD) is adopted;[8] this method is reviewed and reformulated in the Appendix.

2.3. Parameters estimation

By measuring joint positions, velocities, accelerations and commanded torques during a trajectory executed by the manipulator at N time instants $t_2, t_2, \ldots, t_N$ (with $nN > p$), it is possible to write:

$$\boldsymbol{\tau}_N = \begin{bmatrix} \boldsymbol{\tau}(t_1) \\ \boldsymbol{\tau}(t_2) \\ \vdots \\ \boldsymbol{\tau}(t_N) \end{bmatrix} = \begin{bmatrix} \boldsymbol{W}(\boldsymbol{\zeta}(t_1)) \\ \boldsymbol{W}(\boldsymbol{\zeta}(t_2)) \\ \vdots \\ \boldsymbol{W}(\boldsymbol{\zeta}(t_N)) \end{bmatrix} \boldsymbol{\theta} = \boldsymbol{W}_N \boldsymbol{\theta} \tag{10}$$

where the vector $\boldsymbol{\zeta}(t_i) \in \mathbb{R}^{3n}$ defines the "measurement

https://doi.org/10.1017/S026357479900140X Published online by Cambridge University Press

point" at time t_i (i.e., a vector stacking the joint variables, velocities and accelerations at the corresponding time instant), $\boldsymbol{W}_N$ denotes the $(Nn \times p)$ full-rank matrix obtained from $\boldsymbol{W}$ computed in the N measurement points and $\boldsymbol{\tau}_N \in \mathbb{R}^{Nn}$ is the vector obtained from the N measured torque vectors $\boldsymbol{\tau}(t_i)$.

Equation (10) represents an overdetermined system of equations which has no exact solution. Nevertheless, it can be solved using a least-squares technique:

$$\hat{\boldsymbol{\theta}} = (\boldsymbol{W}_N^T \boldsymbol{W}_N)^{-1} \boldsymbol{W}_N^T \boldsymbol{\tau}_N. \tag{11}$$

If an *a priori* knowledge about the values of the dynamic parameters is available, a weighted pseudo-inverse of $\boldsymbol{W}_N$ can be used in (11) to improve the efficiency of the estimation on each parameter.[9]

It is worth pointing out that a recursive least-squares algorithm can be used instead of the batch algorithm (11), although its adoption would be necessary only for on-line parameters estimation.

2.4. Exciting trajectories

In order to improve the numerical efficiency of the least-squares algorithm the excitation matrix $\boldsymbol{\Pi}_N = \boldsymbol{W}_N^T \boldsymbol{W}_N$,[9] or simply the regression matrix $\boldsymbol{W}_N$,[10], should have a small condition number and a large minimum singular value. Therefore, a small sensitivity of the estimation algorithm to measurement noise and unmodeled dynamics can be obtained if a suitable cost function is minimized to find the N measurement points $\boldsymbol{\zeta}(t_i)$ $(i = 1, N)$.

An exciting trajectory is defined as a set of measurement points $\boldsymbol{\zeta}(t_i)$ $(i = 1 \ldots N)$ satisfying the requirements stated above, i.e. the set which can be obtained by minimizing the cost function

$$f(\boldsymbol{\zeta}(t_i), \ldots, \boldsymbol{\zeta}(t_N))) = \lambda_1 \operatorname{cond}(\boldsymbol{W}_N) + \lambda_2 \frac{1}{\sigma_p(\boldsymbol{W}_N)}. \tag{12}$$

In (12) $\sigma_p(\boldsymbol{W}_N)$ is the minimum singular value of $\boldsymbol{W}_N$, $\operatorname{cond}(\boldsymbol{W}_N)$ is the condition number of $\boldsymbol{W}_N$. The scalars λ_1 and λ_2 set the relative weight of the two terms in the cost function; moreover, they have the meaning of normalization factors. It is worth noticing that a simpler expression of f could be devised in terms of $\sigma_1(\boldsymbol{W}_N)$ and $\sigma_p(\boldsymbol{W}_N)$ with only one weighting scalar. However, it is preferable to refer to the expression (12) of the cost function with two distinct terms to remark the different meaning and role of the two terms. In fact, it can be shown that the condition number affects the relative norm error of the least-square solution, while the estimation error covariance does not depend on $\operatorname{cond}(\boldsymbol{W}_N)$ but only on the minimum singular value $\sigma_p(\boldsymbol{W}_N)$.

Once the optimization problem has been solved, the N optimal measurement points must be interpolated to obtain smooth joint trajectories to be executed by the manipulator, e.g. by using 5th order polynomials or splines.

Although an euristic (i.e. trial and error) procedure could be adopted to find an exciting trajectory,[2] a major drawback is that several trials could be necessary to obtain a (sub)optimal set of measurement points.

Hence, it is preferable to solve the optimization problem by resorting to numerical techniques. In order to minimize the numerical problems related to the optimization algorithm and its computational load, it is advisable to resort to a minimum number of decision variables for the optimization problem. To the purpose the minimum number of measurements points N required to build the matrix $\boldsymbol{W}_N$ in (11) is taken, i.e. N is chosen as the minimum integer such that the following inequality is satisfied

$$nN \geq p \tag{13}$$

In fact, a number of measurements such that $nN < p$ would result in an underdetermined system of equations (9), and thus it would be unsuitable for the application of the least-squares algorithm (11). This implies that the minimum number of decision variables of the optimization problem is $3nN$, being each measurement point $\boldsymbol{\zeta}(t_i)$ defined by $3n$ variables.

It is worth noticing that the square of each singular value of $\boldsymbol{W}_N$ is monotonically increasing with the number of rows (see reference 11, corollary 4.9). Apparently, this would suggest to enhance the set of measurements by taking all the samples collected along the interpolated trajectory in addition to the N original samples in $t_1 \ldots t_N$. As a result, a new regressor matrix $\boldsymbol{W}_*$ can be built to evaluate the least-squares solution

$$\hat{\boldsymbol{\theta}} = (\boldsymbol{W}_*^T \boldsymbol{W}_*)^{-1} \boldsymbol{W}_*^T \boldsymbol{\tau}_*, \tag{14}$$

where $\boldsymbol{W}_*$ is matrix built in the same way as $\boldsymbol{W}_N$, but using all the measurements collected along the interpolated trajectory. However, this guarantees that only the second term in (12) decreases, while nothing can be concluded on the behaviour of the first term when the number of samples is increased. Therefore, inspired by the ideas in reference 12, the additional samples to be taken along the interpolated trajectory can be selected via the following criterion:

$$\begin{aligned} &\text{if } f(\boldsymbol{\zeta}(t_1), \ldots, \boldsymbol{\zeta}(t_K)) - f(\boldsymbol{\zeta}(t_1), \ldots, \boldsymbol{\zeta}(t_{K-1})) > \eta \\ &\text{then discard the sample } \boldsymbol{\zeta}(t_K) \end{aligned} \tag{15}$$

where η is a suitably selected threshold. Then, all the components $\boldsymbol{W}(\boldsymbol{\zeta}(t_i))$ in the matrix $\boldsymbol{W}_*$ corresponding to the discarded data are eliminated. Remarkably, the selection of samples according to (15) may cause aliasing and biasing phenomena due to undersampling of data; this is avoided by performing a preliminary low-pass filtering of the sequence of the sequence of data $\boldsymbol{\zeta}(t_i)$

2.5. Derivation of simplified models

The dynamic model in the form (9) expresses the mapping between joint torques and motion variables. In view of real time use of (9), e.g. in computed torque control, it is worth considering simplified models characterized by a satisfactory trade-off between computational load associated to the model and its accuracy.

A simplified model can be obtained by neglecting the contributions to the joint torques due to some links of the open chain. This can be done, e.g. for anthropomorphic manipulators, in which the dynamic effect of the wrist on the elbow is usually neglectable. This simplification is rather straightforward to obtain if the Newton-Euler formulation is used to derive (8), by omitting the contribution

https://doi.org/10.1017/S026357479900140X Published online by Cambridge University Press

of the last wrist's link in the backward force/moment propagation towards the elbow. It is worth remarking that such approach does not lead to neglect the wrist's parameters, but only the effect of the wrist's motion on the dynamic behaviour of the elbow structure, i.e. the corresponding coupling terms; hence, the computational burden of the model is reduced but the number of parameters remains the same. It is clear that the above mentioned reduction procedure is strictly related to the mechanical structure of the manipulator, and the accuracy of the reduced model can be evaluated only *a posteriori*.

A second type of model reduction can be obtained considering that some of the estimated parameters may give a negligible contribution to the joint torques.[13] Therefore, it is possible to eliminate those parameters from the model, and thus reduce the dimensions of the model by eliminating the corresponding columns in $\boldsymbol{W}$. To the purpose, equation (9) can be computed along a selected trajectory to obtain

$$\boldsymbol{\tau}_r = \boldsymbol{W}_r \boldsymbol{\theta} = \sum_{i=1}^{p} \boldsymbol{w}_{r,i} \theta_i, \tag{16}$$

where $\boldsymbol{W}_r$ is a matrix built in the same way as $\boldsymbol{W}_N$, using the samples taken along the chosen trajectory, and $\boldsymbol{w}_{r,i}$ is the ith column of $\boldsymbol{W}_r$. Therefore, the parameters giving negligible contributions can be selected on the basis of the following test:

$$\| \boldsymbol{w}_{r,i} \theta_i \| \leq w_{max}, \tag{17}$$

where w_{max} is a user-defined threshold. The main advantage of this approach is in possibility of setting *a priori* the desired accuracy of the simplified model via w_{max}.

3. SOFTWARE IMPLEMENTATION

Based on the concepts described in the previous section, the procedure sketched in Figure 1 is implemented as a set of software tools implementing the different stages.

3.1. Model derivation

First, the symbolic expression of $\boldsymbol{W}$ (see eq. (9)) is to be derived starting from the kinematic description of the manipulator. To the purpose, a procedure has been developed by exploiting the symbolic computation capabilities of the environment Maple V (Waterloo Maple, Inc.).

The user provides as input a formatted ASCII file containing the Denavit-Hartenberg parameters describing the kinematics of the manipulator; the output of the procedure is the symbolic expression of the matrix $\boldsymbol{W}$ and the set of base parameters. Also, the symbolic expression of the direct kinematics can be optionally required in a compact and optimized form.

The dynamic model is derived via the Newton-Euler formulation described in Section II-A. The set of base parameters $\boldsymbol{\theta}$ and the corresponding matrix $\boldsymbol{W}$ in (9) is obtained via the numeric SVD algorithm described in the Appendix.

Moreover, the first kind of simplification described in Section II-E can be implemented by simply setting a flag in the input file; the flag sets the boundary joint marking the links belonging to the "wrist" (i.e., the links whose dynamic influence on the other joints is to be neglected).

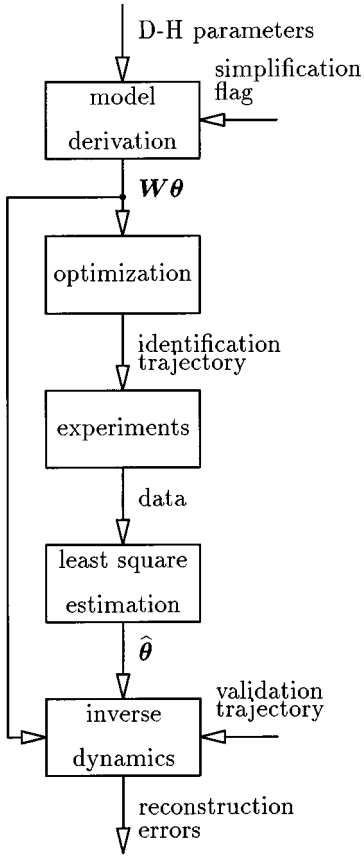


Fig. 1. Sketch of the identification procedure.

3.2. Planning of the exciting trajectories

The software procedure to find the optimally exciting trajectory for the identification is implemented by exploiting the Matlab Optimization Toolbox in the environment Matlab (Mathworks Inc.). The input of this software module is the symbolic expression of the matrix $\boldsymbol{W}$ (that is already provided in Matlab format by the Maple V routine described above) and the mechanical limits for the joint positions, velocities and accelerations. The output of the routine is the optimal set of measurement points $\boldsymbol{\zeta}(t_1) \ldots \boldsymbol{\zeta}(t_N)$.

Then, an interpolation algorithm is implemented to provide a smooth trajectory to be executed by the manipulator. To the purpose, for each positional component of the N measurement points 5th order polynomials are used. In each of the $N-1$ time intervals (t_j, t_{j+1}) $(j=1, \ldots, N-1)$ the polynomial coefficients for each joint are derived by imposing continuity of $\boldsymbol{q}$, $\dot{\boldsymbol{q}}$ and $\ddot{\boldsymbol{q}}$ at the extremities of each time interval. Thus, the trajectory for the ith joint in the jth time interval is expressed as

$$q_i(t) = a_{ij,o} t^5 + a_{ij,1} t^4 + a_{ij,2} t^3 + a_{ij,3} t^2 + a_{ij,4} t + a_{ij,5} \quad \text{with } t \in (t_j, t_{j+1}). \tag{18}$$

However, the peak values of $q_i(t)$, $\dot{q}_i(t)$, $\ddot{q}_i(t)$ in each interval could be greater than the joint mechanical limits. In such a case the interval amplitudes $\Delta T_j = t_{j+1} - t_j$ $(j=1, \ldots, N-1)$ can be modified to iteratively find a set of polynomials

https://doi.org/10.1017/S026357479900140X Published online by Cambridge University Press

fulfilling the joint limits for positions, velocities and accelerations. The output of the interpolation routine consists of an ASCII file containing the coefficients of the polynomials for all the joint trajectories and the length of each time interval.

3.3. Least-squares algorithm

After the execution of the identification trajectory and the collection of the data, the last stage of the identification procedure is performed to compute the estimates of the parameters via the least-squares algorithm (11). To the purpose, a Matlab routine has been developed to compute the least-squares solution.

4. EXPERIMENTAL CASE STUDY

An experimental case study is developed to test the proposed identification procedure. The setup is composed by a conventional industrial manipulator: the SMART-3S manufactured by Comau (Figure 2). The manipulator has six revolute joints actuated by brushless motors via gear trains. Motor positions are measured by shaft absolute resolvers, joint velocity sensors are not available. The original control unit is a VME-based system: the C3G 9000. The robot is equipped with an *open* version of the controller; a bus-to-bus communication link is established with a standard PC, on which the user can develop control algorithms as C modules. Feedback motion control is performed by standard PID control laws, implemented at a sampling rate of 1 kHz. During each sampling interval the PC receives the motor position measurements and provides to the servos the reference torques based on the control algorithm.

The kinematic structure of the manipulator is characterized by a nonspherical wrist (last three joints), while the first three joints are an anthropomorphic structure with a shoulder offset. The Denavit-Hartenberg parameters of the structure are reported in Table I.

The full dynamic model (8) is derived, where the matrix $\boldsymbol{P}$ has dimensions (6×78). Further, a set of 52 base

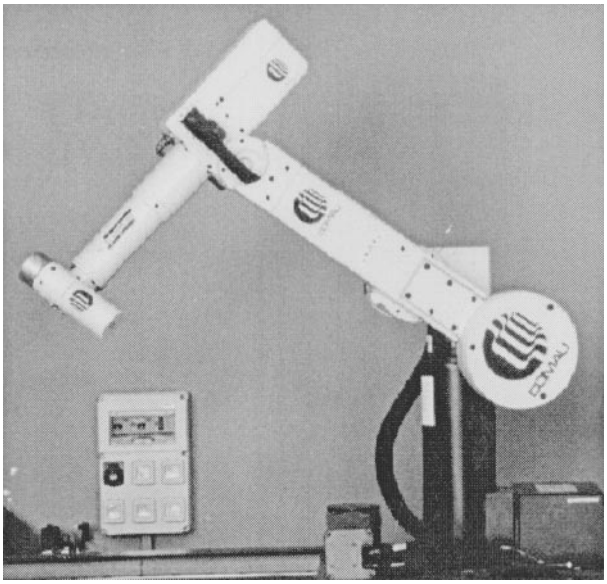


Fig. 2. The industrial robot SMART-3 S.

Table I. D-H parameters [m,rad].

	a	d	θ	α
1	0.150	0	θ_1	$-\pi/2$
2	0.610	0	θ_2	0
3	0.110	0	θ_3	$-\pi/2$
4	0	0.610	θ_4	$\pi/2$
5	0	-0.113	θ_5	$-\pi/2$
6	0	0.103	θ_6	0

parameters has been obtained by applying the SVD-based procedure described in the Appendix; hence, the model (9) written in terms of base parameters is characterized by the (6×52) matrix $\boldsymbol{W}$; this will be denoted as the model A.

A first reduced model is derived by neglecting the dynamic effects of the last three links (i.e. the manipulator's wrist) on the remaining links; by applying the SVD-based procedure described in the Appendix the number of parameters for this model is decreased from 78 to the 51 base parameters. The resulting model will be denoted as model B.

The symbolic expressions of the base parameters for the models A and B are not reported for brevity and they can be found in reference 14.

The minimum number of measurements points required to build $\boldsymbol{W}_N$ is 9 for both models A and B. Hence, the related optimization problem is characterized by 162 decision variables. The obtained values of the cost function f in (12) after the numerical optimization are reported in Table II.

For each of the two models A and B the 9 (sub)optimal measurement points are then interpolated to obtain smooth joint trajectories to be executed in the identification experiments. The two identification trajectories have been executed, and motor positions and currents have been recorded at a sampling rate of 100 Hz. Velocities and accelerations are numerically reconstructed off-line from the joint positions via the digital filter

$$\dot{x}_k = \frac{x_{k+1} - x_{k-1}}{2\Delta T}. \tag{19}$$

In order to mitigate the effect of measurement noise on the estimation a preliminary low-pass filtering of the measured joint positions is performed as well as of the numerically reconstructed joint velocities. Moreover, it should be noted the digital filter (19) performs a band-pass filtering of the input signals, i.e. positions and velocities.

The parameters estimate is then obtained via (11), where $\boldsymbol{W}_*$ is computed by using all the recorded measurements points except for those discarded on the basis of (15). The resulting values of the cost function for both models is reported in Table III; it can be noticed that, as expected, the values of cost function are improved when the set of

Table II. Cost function values.

	Model A	Model B
f	700	508
cond($\boldsymbol{W}_N$)	235.6	134.6
$\sigma_p(\boldsymbol{W}_N)$	0.214	0.267

https://doi.org/10.1017/S026357479900140X Published online by Cambridge University Press

Table III. Cost function values.

	Model *A*	Model *B*
f	118	135
cond($\boldsymbol{W}_*$)	105.8	118.7
$\sigma_p(\boldsymbol{W}_*)$	7.880	5.758

measures is enhanced by taking all the samples along the identification trajectory, except for those satisfying the test (15). The identification trajectory is reported, only for the model *A*, in Figure 3.

In order to test the accuracy of the identified models a validation trajectory has been executed by the manipulator. During the motion, currents and joint positions have been recorded at a sampling rate of 100 Hz, while joint velocities and accelerations have been computed off-line via (19). By using the collected data, the joint torques are reconstructed via manipulator's inverse dynamics (9) and compared to those measured during the motion. Figures 4 and 5 show the results in terms of measured and reconstructed torques; it can be easily recognized that a good accuracy is achieved even for the simplified model.

A second simplified model (model *C*) is derived by using the test (17) on a set of experimental data collected along a suitable trajectory; the resulting matrix $\boldsymbol{W}$ is (6×25). The accuracy of this model has also been verified on the same validation trajectory. The results of the validation for this model reported in Figure 6 show that the accuracy of the estimation is nearly the same as for the previous models.

In order to carry out a comparison between the three identified models, a global measure of the model accuracy has been computed as the mean of the absolute value of the reconstruction errors between measured and reconstructed

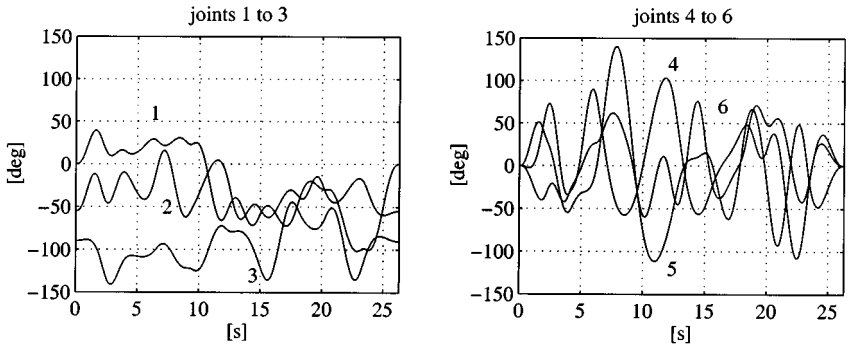


Fig. 3. Identification trajectory for model *A*.

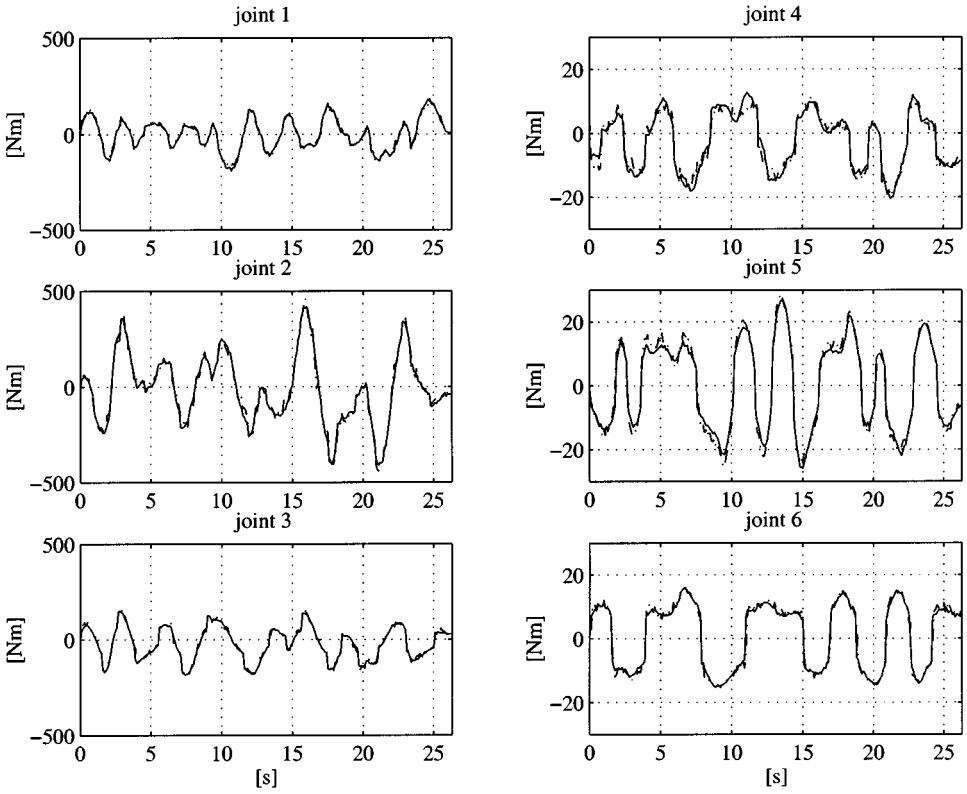


Fig. 4. Model *A*. Measured (dashdot) and reconstructed (solid) torques.

https://doi.org/10.1017/S026357479900140X Published online by Cambridge University Press

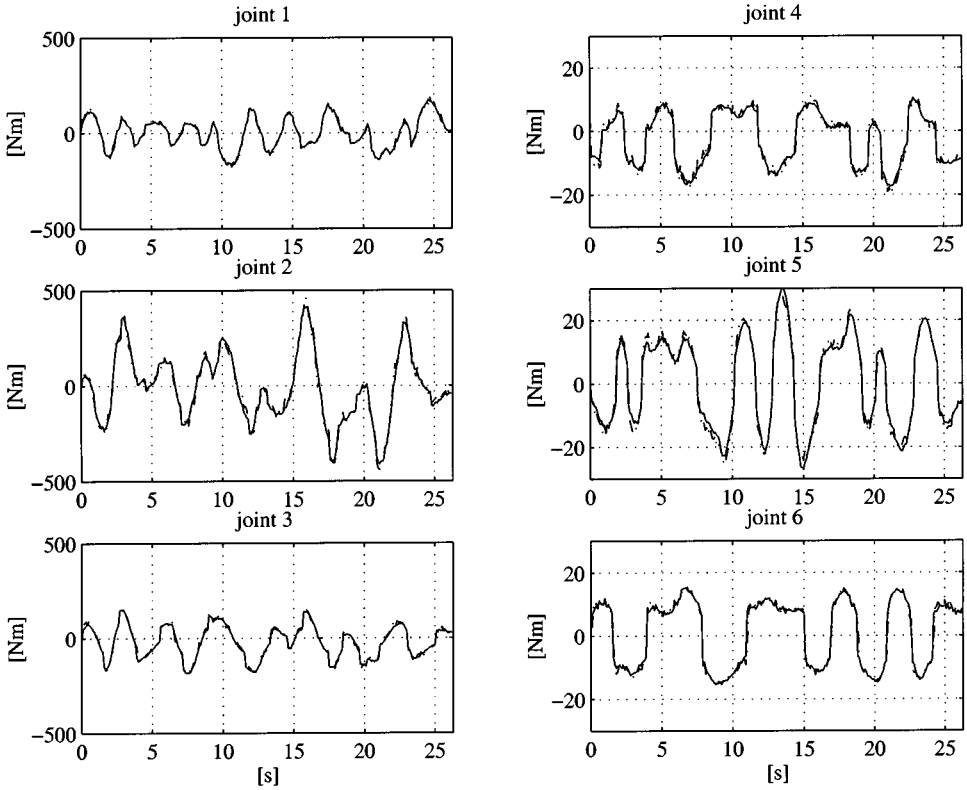


Fig. 5. Model *B*. Measured (dashdot) and reconstructed (solid) torques.

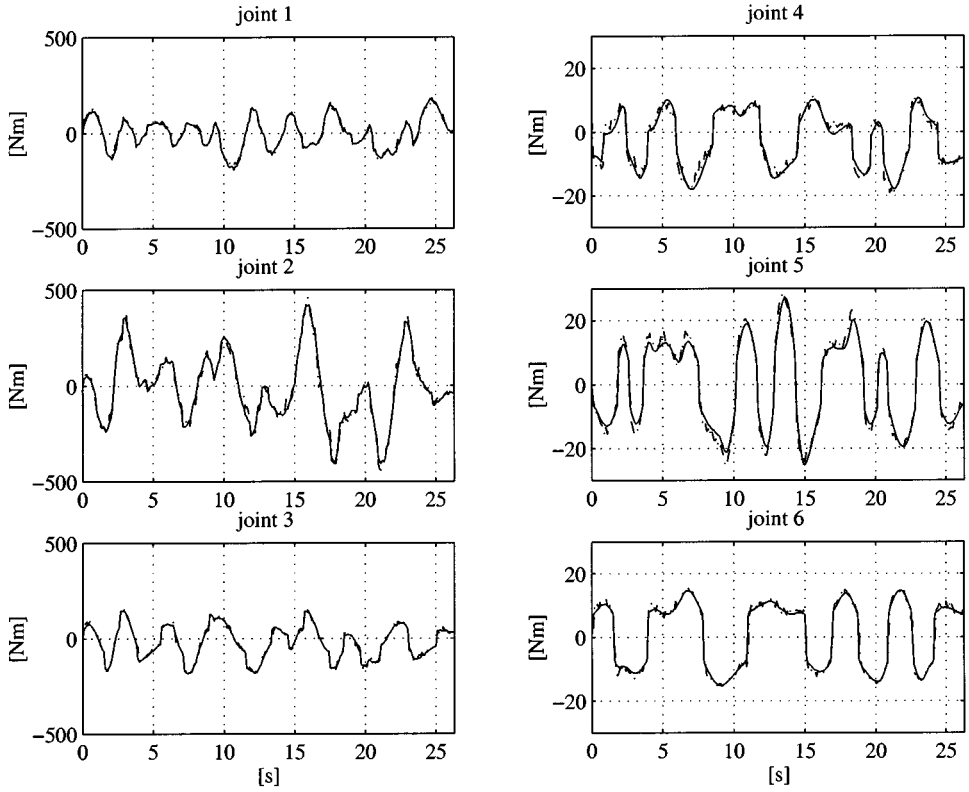


Fig. 6. Model *C*. Measured (dashdot) and reconstructed (solid) torques.

https://doi.org/10.1017/S026357479900140X Published online by Cambridge University Press

Table IV. Reconstruction errors ([Nm]).

	Joint 1	Joint 2	Joint 3	Joint 4	Joint 5	Joint 6
model *A*	8.08	16.8	8.47	1.55	1.67	0.89
model *B*	7.87	16.1	8.28	1.10	1.48	0.81
model *C*	8.44	16.8	8.02	1.43	1.75	0.95

Table V. Relative errors ($\times 100$).

	Joint 1	Joint 2	Joint 3	Joint 4	Joint 5	Joint 6
model *A*	3.76	3.50	4.16	7.52	5.77	5.58
model *B*	3.66	3.37	4.06	5.35	5.12	5.03
model *C*	3.92	3.50	3.94	6.92	6.06	5.90

Table VI. Computational load.

	Model *A*	Model *B*	Model *C*
flops	4763	827	658

joint torques along the validation trajectory; the results of the comparison are reported in Table IV. In Table V the mean errors relative to the maximum absolute values of the corresponding joint torques are compared.

Finally, in Table VI the computational load related to each model is compared in terms of the number of floating point operations required to compute the joint torques from $\boldsymbol{q}$, $\dot{\boldsymbol{q}}$ and $\ddot{\boldsymbol{q}}$ (i.e., the inverse dynamics).

In conclusion, the two simplified models (i.e. models *B* and *C*) are characterized by the same degree of accuracy as for the complete model (model *A*) and, at the same time, provide a dramatic reduction of the computational effort needed to evaluate the inverse dynamics.

5. CONCLUSIONS

In this paper a systematic procedure for the identification of dynamic parameters for rigid robot manipulators has been proposed. The procedure is articulated in the main points needed for the identification for a generic open-chain manipulator: model derivation, design of the identification trajectories, estimation algorithm and model simplification. Each step of the procedure has been deeply analyzed and implemented as a software routine developed by using the popular software environment Matlab/Maple. The result is a tool aimed at assisting the user in setting up the identification procedure from the modeling stage to the design of experiments. Finally, the procedure has been experimentally tested on a six-degree-of-freedom conventional industrial robot. The obtained results show that the proposed procedure leads to accurate and reliable estimates of the dynamic parameters also in an industrial-oriented context, while a limited effort is required to the user.

References

1. M.B. Leahy and G.N. Saridis, "Compensation of industrial manipulator dynamics" *Int. J. Robotics Research* **8**, 73–84, (1989).
2. F. Caccavale and P. Chiacchio, "Identification of dynamic parameters and feedforward control for a conventional industrial manipulator" *Control Engineering Practice* **2**, 1039–1050 (1994).
3. L. Sciavicco and B. Siciliano, *Modeling and Control of Robot Manipulators* (McGraw-Hill, New York, NY, 1996).
4. C.G. Atkenson, C.H. An and J.M. Hollerbach, "Estimation of inertial parameters of manipulator loads and links" *Int. J. Robotics Research*, **5**, 101–119 (1989).
5. M. Gautier and W. Khalil, "On the identification of inertial parameters of robots" *Proceedings of the 27th IEEE Conference on Decision and Control* Austin, TX (1988) pp. 2264–2268.
6. B. Armstrong-Hélouvry, P. Dupont and C. Canudas De Wit, "A survey of models, analysis tools and compensation methods for the control of machines with friction" *Automatica* **30**, 1083–1138 (1994).
7. M. Gautier and W. Khalil, "Direct calculation of minimum set of inertial parameters of serial robots" *IEEE Transactions on Robotics and Automation* **6**, 368–373 (1990).
8. M. Gautier, "Numerical calculation of the base inertial parameters of robots" *Proceedings of the 1990 IEEE Conference on Robotics and Automation*, Cincinnati, OH (1990) pp. 1020–1025.
9. C. Presse and M. Gautier, "New criteria of exciting trajectories for robot identification" *Proceedings of the 1993 IEEE International Conference on Robotics and Automation*, Atlanta, GA (1993) pp. 1131–1138.
10. B. Armstrong, "On finding 'exciting' trajectories for identification experiments involving systems with non-linear dynamics" *Proceedings of the 1987 IEEE International Conference on Robotics and Automation*, Raleigh, NC (1987) pp. 1131–1138.
11. G.W. Stewart and J.G. Sun, *Matrix Perturbation Theory, Computer Science and Scientific computing*, (Academic Press, San Diego, 1990).
12. P. Carrette, Y. Genin, G. Bastin and M. Gevers, "Discarding data may improve the parameter estimation accuracy in system identification" *Proceedings of the 3rd European Control Conference*, Roma, Italy (1995) pp. 827–832.
13. G. Antonelli, F. Caccavale and P. Chiacchio, "Experimental estimation of dynamic parameters for an industrial manipulator" *Proceedings of the 2nd IMACS Symposium on Mathematical Modelling*, Wien, A (1997) pp. 667–672.
14. G. Antonelli, F. Caccavale and P. Chiacchio, "Experimental Identification of the Comau SMART-3 S Robot" *PRISMA Tech. Rep.* Napoli, Italy 98–102 (1998).

APPENDIX: SVD-BASED DETERMINATION OF THE BASE PARAMETERS

In the following the procedure proposed in reference 8 to find the set of base parameters is reviewed and suitably reformulated for software implementation purposes.

Let $r=Mn$, $c=13n$ $(r \geq c)$, and let $\boldsymbol{P}_M$ be the $(r \times c)$ regression matrix obtained by computing (8) in correspondence of M (random) samples

$$\boldsymbol{\tau}_M = \begin{bmatrix} \boldsymbol{\tau}(t_1) \\ \boldsymbol{\tau}(t_2) \\ \vdots \\ \boldsymbol{\tau}(t_M) \end{bmatrix} = \begin{bmatrix} \boldsymbol{P}(\boldsymbol{\zeta}(t_1)) \\ \boldsymbol{P}(\boldsymbol{\zeta}(t_2)) \\ \vdots \\ \boldsymbol{P}(\boldsymbol{\zeta}(t_M)) \end{bmatrix} \boldsymbol{\gamma} = \boldsymbol{P}_M \boldsymbol{\gamma} \qquad (20)$$

with $\text{rank}(\boldsymbol{P}_M)=p \leq c$ (where p is the dimension of the vector $\boldsymbol{\theta}$ of base parameters). An SVD decomposition of $\boldsymbol{P}_M$ yields

$$\boldsymbol{P}_M = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^T = \boldsymbol{U} \begin{bmatrix} \Sigma & \boldsymbol{O} \\ \boldsymbol{O} & \boldsymbol{O} \end{bmatrix} \begin{bmatrix} \boldsymbol{V}_1^T \\ \boldsymbol{V}_2^T \end{bmatrix} \qquad (21)$$

https://doi.org/10.1017/S026357479900140X Published online by Cambridge University Press

where $\boldsymbol{U} \in \mathbb{R}^{r\times r}$, $\boldsymbol{V}_1 \in \mathbb{R}^{c\times p}$, $\boldsymbol{V}_2 \in \mathbb{R}^{c\times c-p}$, $\Sigma \in \mathbb{R}^{p\times p}$ is the diagonal matrix of the singular values $\sigma_1 \ldots \sigma_p$, and $\boldsymbol{U}$, $\boldsymbol{V}$ are orthogonal matrices.

Equations (20) and (21) yield

$$\boldsymbol{\tau}_M = \boldsymbol{U}\begin{bmatrix} \Sigma \boldsymbol{V}_1^T \boldsymbol{\gamma} \\ \boldsymbol{O} \end{bmatrix} \tag{22}$$

from which it can be recognized that the unidentifiable parameters (i.e., the parameters giving identically null contribution to the joint torques) are those corresponding to the null columns of $\boldsymbol{V}_1^T$. Moreover, the rows of $\boldsymbol{V}_2^T$ span a subspace in $\mathbb{R}^c$ orthogonal to that spanned from the rows of $\boldsymbol{V}_1^T$, and thus the identifiable parameters corresponds to the null columns of $\boldsymbol{V}_2^T$.

The remaining parameters are those identifiable only in linear combinations. In the following a procedure to find a set of linear combinations of such parameters is given. To the purpose, notice that the columns of $\boldsymbol{P}_M$ corresponding to the absolutely identifiable and unidentifiable parameters can be eliminated, and a new regression matrix $\boldsymbol{P}'_M$ can be derived. By means of an SVD of $\boldsymbol{P}'_M$

$$\boldsymbol{P}'_M = \boldsymbol{U}'\boldsymbol{S}'\boldsymbol{V}'^T = \boldsymbol{U}'\begin{bmatrix} \Sigma' & \boldsymbol{O} \\ \boldsymbol{O} & \boldsymbol{O} \end{bmatrix}\begin{bmatrix} \boldsymbol{V}_1'^T \\ \boldsymbol{V}_2'^T \end{bmatrix} \tag{23}$$

the following relation can be established

$$\boldsymbol{P}'_M \boldsymbol{V}'_2 = \boldsymbol{O} \tag{24}$$

which reveals that the columns of $\boldsymbol{V}'_2$ define the sought linear combinations between the columns of $\boldsymbol{P}'_M$. From the above equation it follows

$$\boldsymbol{P}'_M \boldsymbol{\gamma}' = \boldsymbol{P}'_M(\boldsymbol{\gamma}' + \boldsymbol{V}'_2 \boldsymbol{\gamma}'_a) \tag{25}$$

where $\boldsymbol{\gamma}'$ is the reduced vector of parameters and $\boldsymbol{\gamma}'_a$ is an arbitrary vector in $\mathbb{R}^p$. Hence, there exist an infinite number of vectors in the form

$$\boldsymbol{\gamma}'_r = \boldsymbol{\gamma}' + \boldsymbol{V}'_2 \boldsymbol{\gamma}'_{2a} \tag{26}$$

satisfying (25). Notice that the columns of $\boldsymbol{V}'_2$ (and the corresponding elements in $\boldsymbol{\gamma}'$) can be rearranged in such a way that the following holds

$$\boldsymbol{E}^T \boldsymbol{V}'_2 = \begin{bmatrix} \boldsymbol{V}'_{21} \\ \boldsymbol{V}'_{22} \end{bmatrix} \tag{27}$$

$$\boldsymbol{E}^T \boldsymbol{\gamma}' = \begin{bmatrix} \boldsymbol{\gamma}'_1 \\ \boldsymbol{\gamma}'_2 \end{bmatrix} \tag{28}$$

where $\boldsymbol{E}$ is a permutation matrix and $\boldsymbol{V}'_{22}$ is a regular matrix. Notice that there exist infinite permutation matrices $\boldsymbol{E}$ such that $\boldsymbol{V}'_{22}$ is regular. Therefore, a solution for equation (26) with $\boldsymbol{\gamma}'_{r,2} = \boldsymbol{0}$ is

$$\boldsymbol{\gamma}'_{r,1} = \boldsymbol{\gamma}'_1 - \boldsymbol{V}'_{21}\boldsymbol{V}'^{-1}_{22}\boldsymbol{\gamma}'_2 \tag{29}$$

Hence, $\boldsymbol{\gamma}'_{r,1}$ represents the vector containing the sought linear combinations of the parameters in $\boldsymbol{\gamma}'$.

Therefore, the vector $\boldsymbol{\theta}$ of the base parameters is given by the elements of $\boldsymbol{\gamma}'_{r,1}$ together with the parameters previously recognized as absolutely identifiable. The corresponding matrix $\boldsymbol{W}$ in (9) can be built by taking the columns of $\boldsymbol{P}$ corresponding to the absolutely identifiable parameters and the columns corresponding to the parameters in $\boldsymbol{\gamma}'_1$.

Finally it is worth noticing that in reference 8 an algorithm based on the QR decomposition of the matrix $\boldsymbol{P}_M$ is proposed in alternative to that reviewed in this section.

https://doi.org/10.1017/S026357479900140X Published online by Cambridge University Press