**SLICOT working note 1999-20**

# Symbolic and Computational pre-processing in physical parameter estimation of multi-body mechanical systems [1]

Michel Verhaegen[2]

November 1999

[1]This paper presents research results of the European Community BRITE-EURAM III Thematic Networks Programme NICONET (contract number BRRT-CT97-5040) and is distributed by the Working Group on Software WGS. *WGS secretariat*: Mrs. Ida Tassens, ESAT - Katholieke Universiteit Leuven, K. Mercierlaan 94, 3001-Leuven-Heverlee, BELGIUM. This report is also available by anonymous ftp from *wgs.esat.kuleuven.ac.be* in the directory *pub/WGS/REPORTS/SLWN1999-20.ps.Z*

[2]Univeristy of Twente, Department of Applied Physics, Systems and Control Engineering Division, P.O. Box 217, 7500 AE Enschede, The Netherlands

## Contents



# 1 Objective of the working note.

The objective of this note is to highlight the scope and computational (symbolic and/or arithmetic) tasks of turning a physical parameter estimation problem into a (constraint) optimization problem.
Concrete examples show the need for symbolic (object-oriented) modeling environments for defining the structure of the physical system to be used in the parameter optimization step. Without this (interactive) software environment for compiling a physical parameter estimation problem into an optimization problem, standarization of commercial optimization routines is of little or no interest.

# 2 Symbolic formulation of a parameter optimization problem

## 2.1 Continuous-time problem formulation

The problem of identifying unknown parameters in industrial robots is outlined e.g. in [6, 5]. In general terms, multi-body dynamical systems can be described by $n$ differential equations, where $n$ is the minimal number of rotational and translation robot joint coordinates, denoted as:

$$M(q(t))\ddot{q}(t) = u(t) + h(q(t), \dot{q}(t), t) \quad t \in [t_1, t_N] \tag{1}$$

For example, for $n = 3$, we have [5]:

$$\begin{pmatrix} M_{11}(q(t)) & M_{12}(q(t) & M_{13}(q(t) \\ M_{21}(q(t)) & M_{22}(q(t) & M_{23}(q(t) \\ M_{31}(q(t)) & M_{32}(q(t) & M_{33}(q(t) \end{pmatrix} \begin{pmatrix} \ddot{q}_1(t) \\ \ddot{q}_2(t) \\ \ddot{q}_3(t) \end{pmatrix} = \begin{pmatrix} u_1(t) \\ u_2(t) \\ u_3(t) \end{pmatrix} + \begin{pmatrix} h_1(q(t), \dot{q}(t), t) \\ h_2(q(t), \dot{q}(t), t) \\ h_3(q(t), \dot{q}(t), t) \end{pmatrix} \tag{2}$$

The robot joint coordinates $q = (q_1(t), q_2(t), q_3(t))^T$ are the state variables, and the normalised torques $u = (u_1(t), u_2(t), u_3(t))^T$ are the control vairables. The matrix $M(q)$ of Inertial moments is symmetric, positive definit. The term $h(q(t), \dot{q}(t), t)$ contains the torques due to Coriolis-, Centrifugal-, Gravitational- and friction forces. For the formulation of these differential equations use can be made of efficient Multi-body modeling languages, such as reported in [7].

The determination of the unknown parameters in the model (1) can be formulated as a parameter optimization problem. Let the following measurements of the joint coordinates $q_m(t) = [(q_m)_1, (q_m)_2, \cdots, (q_m)_n]^T$

and the normalized torques $u(t)$ for $t = t_1, \cdots, t_N$ be recorded, then the problem of determining the unknown parameter, stored in the parameter vector $\theta$ can be formulated as the following non-linear least squares problem:

$$V_N(\theta, q(t_1), \dot{q}(t_1)) = \sum_{j=1}^{N} \sum_{i=1}^{n} \left( (q_m)_i(t_j) - q_i(t_j) \right)^2 / \omega_{i,j} \tag{3}$$

subject to

$$\begin{aligned} M(q(t), \theta)\ddot{q}(t) &= u(t) + h(q(t), \dot{q}(t), t, \theta) \quad t \in [t_1, t_N] \\ 0 &= r(q(t_1), \dot{q}(t_1), \theta) \end{aligned}$$

The numerical solution of the resulting parameter optimization problem (3) can be done by the generalized Gauss-Newton algorithm or by the sequential programming method [1, 3, 4]. For that purpose, it is necessary to efficiently evaluate the gradients

$$(\partial q / \partial \theta)(t), (\partial q / \partial q(0))(t)$$

More information on the calculation, implementation on the use of gradients can be found in [2, 4].

**Remark 1** *Both the formulation of the constraints to the optimization problem stated in Eq. (3) and the evaluation of the above gradients need to be automized for realistic multi-body mechanical systems. Such an automation highly depends on the modeling language used to define the multi-body systems. Examples of such language is the object-oriented MODELICA [8] modeling language integrated in a number of commercially available modeling and simulation packages, such as e.g. Dymola.*
*Ready made commercial software package that perform the necessary pre-processing to formulate the determination of physical unknown parameters in a model defined based on first principles as a 'standard' parameter optimization problem is the EASY-FIT package [9].* □

The complexity of formulating the determination of physical unknown parameters in a model defined based on first principles as a 'standard' parameter optimization problem is illustrated by means of 2 simple examples. Before treating these problems, (3) is first formulated into the discrete-time case manipulating sampled data sequences.

## 2.2 Discrete-time problem formulation

To turn the symbolic problem formulation (3) into a numerical optimization problem, the continuous-time equations will be discretized. For that purpose, the time derivatives of $q(t)$ are approximated. This can be done for example making use of the explicit BDF (Backward Difference Formulae).

**Definition 1** *A* k-th *order backward differentiation formula to approximate the derivative of a sufficiently smooth vector funtion $x \in C(\mathbb{R}, \mathbb{R}^n)$, denoted by $D_\Delta x$, is given as:*

$$D_\Delta x(t) = \frac{1}{\Delta} \sum_{\ell=0}^{k} \alpha_\ell x(t - \ell\Delta) \tag{4}$$

*where for the constant step size case the coefficients $\alpha_\ell$ are given in Table 1*

**Remark 2** *From Table 1, we observe that,*

| $\alpha_\ell$ | $\ell=0$ | $\ell=1$ | $\ell=2$ | $\ell=3$ | $\ell=4$ | $\ell=5$ | $\ell=6$ |
|---|---|---|---|---|---|---|---|
| $k=1$ | $1$ | $-1$ | | | | | |
| $k=2$ | $\frac{3}{2}$ | $-2$ | $\frac{1}{2}$ | | | | |
| $k=3$ | $\frac{11}{6}$ | $-3$ | $\frac{3}{2}$ | $-\frac{1}{3}$ | | | |
| $k=4$ | $\frac{25}{12}$ | $-4$ | $3$ | $-\frac{4}{3}$ | $\frac{1}{4}$ | | |
| $k=5$ | $\frac{137}{60}$ | $-5$ | $5$ | $-\frac{10}{3}$ | $\frac{5}{4}$ | $-\frac{1}{5}$ | |
| $k=6$ | $\frac{147}{60}$ | $-6$ | $\frac{15}{2}$ | $-\frac{20}{3}$ | $\frac{15}{4}$ | $-\frac{6}{5}$ | $\frac{1}{6}$ |

Table 1: Coefficients $\alpha_\ell$ in the BDF integration schemes.

1. *for $k = 1$ we obtain the backward Euler method.*
2. *the coefficients $\alpha_i$ from the coefficients of an Hurwitz polynomial.*

□

Let the time inverval $[t_1, t_N]$ be sampled in an equidistant way, such that $t_j - t_{j-1} = \Delta$ (for $j > 1$) then, the backward Euler method leads to the following approximations of the derivatives $\dot{q}(t_1), \ddot{q}(t_j)$,

$$\dot{q}(t_1) = \frac{q(t_1) - q(t_0)}{\Delta} \tag{5}$$

$$\ddot{q}(t_i) = \frac{q(t_j) - 2q(t_{j-1}) + q(t_{j-2})}{\Delta^2} \quad \text{for} \quad j > 1 \tag{6}$$

In this way, the constraints of Eq. (3) are transformed into,

$$\begin{gathered} q(t_1) = \overline{x} \\ \frac{q(t_1) - q(t_0)}{\Delta} = \overline{y} \\ M(q(t_i), \theta)\frac{q(t_j) - 2q(t_{j-1}) + q(t_{j-2})}{\Delta^2} = u(t_j) - h(q(t_j), \frac{q(t_j) - q(t_{j-1})}{\Delta}, t_j, \theta) \quad \text{for} \quad j = 2 : N \end{gathered} \tag{7}$$

where $\overline{x}, \overline{y}$ are assumed to be given. If we denote the time sequence of the time sequence of generalized coordinates by the vector x as follows,

$$x = \begin{bmatrix} q(t_0) \\ q(t_1) \\ \vdots \\ q(t_N) \end{bmatrix}$$

and denote the Eqs. (7) compactly as,

$$A(x, \theta)x - b(x, \theta) = 0$$

then the discrete-time variant of (3) can be denoted as:

$$V_N^d(\theta, q(t_0)) = \sum_{j=1}^{N} \sum_{i=1}^{n} \left((q_m)_i(t_j) - q_i(t_j)\right)^2 / \omega_{i,j} \tag{8}$$

subject to

$$A(x, \theta)x - b(x, \theta) = 0$$

This problem can be addressed by standard numerical optimization methods, such as *Sequential Quadratic programming* or the *Penalty function method or the Barrieres function method.* From the NAG library use can be made of the `E04UPF` routine.

In the subsequent section, the need for a symbolic pre-processing tool just to define the functions $A(x,\theta)$ and $b(x,\theta)$ (and not their gradients with respect to the unknown parameter vector $\theta$) is illustrated by means of 2 simple multibody systems.

# 3 Illustration of the minimal required pre-processing

## 3.1 Example 1: A mass-spring system

Consider the mass-spring system with damping in Figure 1. The equations of motion, which are *symbolic*,

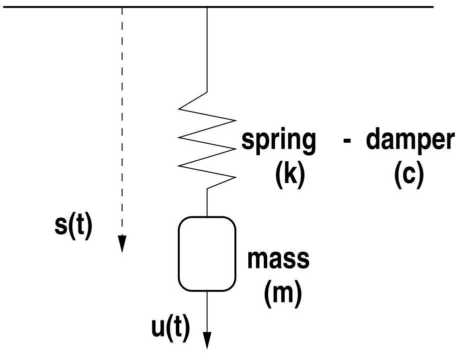


Figure 1: A schematic representation of the relevant quantities in the mass-spring system.

of the mass-spring system in Figure 1, under classical assumptions that all mechanical components are rigid and ideal, reads:

$$m\ddot{s}(t) = u(t) - c\dot{s}(t) - ks(t) \quad s(t_0) = \overline{x} \quad \dot{s}(t_0) = \overline{y} \tag{9}$$

The *unknown* system parameters are the characteristic quantities of the spring $c, k$ and the mass $m$. The external force $u(t)$ is caused by gravity and equals $mg$, where $g = 9.81\frac{m}{s^2}$. Recasting this equation of motion into the general form (1) leads to the correspondence listed in Table 2. With the analogy between

| **Quantity in the General model (3)** | **mass-spring specification** |
|---|---|
| $q(t)$ | $s(t)$ |
| $M(q(t))$ | $m$ |
| $u(t)$ | $0$ |
| $h(q(t),\dot{q}(t))$ | $mg - c\dot{x}(t) - kx(t)$ |
| $\theta$ | $[s(t_1), \dot{s}(t_1), m, c, k]^T$ |

Table 2: Definition of the relevant functions in the general multibody representation (3) for the mass-spring system.

the general representation (1) and the specific mass-spring system one can *symbollically* construct the constraints

$$A(x,\theta)x - b(x,\theta) = 0$$

to the parameter optimization problem (8). Generating from this the appropriate input to numerical solvers for solving this optimization, such as `E04UPF` of the NAG library, still requires the construction of a parser to translate the symbolic equation into the right numerical format.

### 3.2 Example 2: An inverted pendulum on a car

Consider the inverted pendulum on a cart in Figure 2. Again following Newton's principles, we can derive

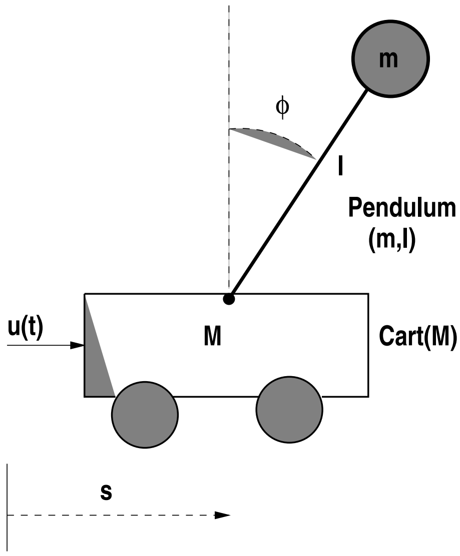


Figure 2: A schematic representation of the relevant quantities in the inverted pendulum on a cart.

from this figure, the following equations of motion of this multibody dynamical systems:

$$\begin{aligned} (M+m)\ddot{s}(t) - mlsin(\phi(t))\dot{\phi}^2(t) + mlcos(\phi(t))\ddot{\phi}(t) &= u(t) \\ mlcos(\phi(t))\ddot{s}(t) + ml^2\ddot{\phi}(t) - mglsin(\phi(t)) &= 0 \end{aligned} \tag{10}$$

With initial conditions:

$$\begin{pmatrix} \phi(t_1) \\ s(t_1) \end{pmatrix} = \bar{x} \quad \text{en} \quad \begin{pmatrix} \dot{\phi}(t_1) \\ \dot{s}(t_1) \end{pmatrix} = \bar{y} \tag{11}$$

The function $u(t)$ now represents the external force applied to the cart, see Figure 2. The equations of motion of the cart-pendulum example can be recasted into the general multibody dynamics form (1):

$$\begin{bmatrix} (M+m) & mlcos(\phi(t)) \\ mlcos(\phi(t)) & ml^2 \end{bmatrix} \begin{bmatrix} \ddot{s}(t) \\ \ddot{\phi}(t) \end{bmatrix} = \begin{bmatrix} u(t) \\ 0 \end{bmatrix} + \begin{bmatrix} mlsin(\phi(t))\dot{\phi}^2(t) \\ mglsin(\phi(t)) \end{bmatrix} \tag{12}$$

This leads to the correspondence listed in Table 3.

The table again shows how to proceed further *symbolically* to derive the constraints of the optimization problem (8).

## 4 Conclusions

This working note has analysed the preparation necessary to formulate the problem of estimating the physical parameters of multibody dynamical systems as parameter estimation problem. Via concrete

| Quantity in the General model (3) | cart-pendulum specification |
|---|---|
| $q(t)$ | $\begin{bmatrix} s(t) \\ \phi(t) \end{bmatrix}$ |
| $M(q(t))$ | $\begin{bmatrix} (M+m) & mlcos(\phi(t)) \\ mlcos(\phi(t)) & ml^2 \end{bmatrix}$ |
| $u(t)$ | $\begin{bmatrix} u(t) \\ 0 \end{bmatrix}$ |
| $h(q(t), \dot{q}(t))$ | $\begin{bmatrix} mlsin(\phi(t))\dot{\phi}^2(t) \\ mglsin(\phi(t)) \end{bmatrix}$ |
| $\theta$ | $\begin{bmatrix} s(t_1), \dot{s}(t_1), \phi(t_1), \dot{\phi}(t_1), M, m, l \end{bmatrix}^T$ |

Table 3: Definition of the relevant functions in the general multibody representation (3) for the cart-pendulum system.

examples it was shown that *symbolic preprocessing* is necessary for *defining* the parameter estimation as a standard least squares optimization problem. The symbolic preprocessing, which is also helpful in deriving the gradients to be used in the numerical optimization, are naturally used in modern *object oriented* modeling languages, such as MODELICA [8], or special purpose physical parameter identification packages, such as EASY-FIT [9]. Since symbolic manipulation is not within the scope of the NICONET project and since creating these features in SLICOT will require at least one manyear, it is proposed not to fill this part of the planned SLICOT library.

# References


[1] Bock, H.G., Eich, E., Schlöder, J.P.: Numerical solution of constrained least squares boundary value problems in differential-algebraic equations. In: K. Strehmel (Hrsg.): Numerical treatment of Differential Equations, Proc. of the NUMDIFF-4 Conference, Halle-Wittenberg, 1987, Teubner Texte zur Mathematik 104 (1988), 269-280.

[2] Buchauer, O., Hiltmann, P., Kiehl, M.: Sensitivity analysis of initial-value problems with application to shooting techniques. Numerische Mathematik 67 (1994), 151-159.

[3] Gill, P.E., Murray, W., Saunders M.A., Wright, M.H.: User's guide for NPSOL (Version 4.0), Report SOL 86-2, Dept. of Operations Research, Stanford University, California, USA, 1986.

[4] Heim A., Parameteridentifizierung in differential-algebraischen Gleichungssystemen. Diplomarbeit, Mathematisches Institut, Technische Universität München, 1992.

[5] Heim, A., von Stryk, O., Pesch, H.J., Schäffler, H., Scheuer, K., Parameteridentifikation, Bahnoptimierung und Echtzeitsteuerung von Robotern in der industriellen Anwendung.

[6] Kozlowski, K., Modeling and Identification in Robotics, Springer Verlag, 1998.

[7] Schiehlen, W.O. (Hrsg): Multibody system handbook. (Berlin, Heidelberg, New York, Tokyo: Springer-Verlag, 1990).

[8] H. Elmqvist and S.E. Mattsson, "MODELICA — The Next Generation Modeling Language: An international Design Effort," in Proc. of the 1st World Congress on System Simulation, Sept. 1-3, Singapore, 1997.

[9] Accessible through WWW (URL:http://www.ask.uni-karlsruhe.de/doc/easyfit)