Katholieke Universiteit Leuven

Departement Elektrotechniek

ESAT-SISTA/TR 2004-53

Computation of LTI system responses directly from input/output data [1]

Ivan Markovsky, Jan C. Willems, Paolo Rapisarda, and Bart L.M. De Moor[2]

22 February 2004
Submitted for publication to *CDC 2004*

[1]This report is available by anonymous ftp from *ftp.esat.kuleuven.ac.be* in the directory *pub/sista/markovsky/reports/04-53.ps.gz*

[2]K.U.Leuven, Dept. of Electrical Engineering (ESAT), Research group SCD (SISTA), Kasteelpark Arenberg 10, 3001 Leuven-Heverlee, Belgium,
Tel. 32/16/32 17 09, Fax 32/16/32 19 70,
WWW: *http://www.esat.kuleuven.ac.be/sista*.
E-mail: *ivan.markovsky@esat.kuleuven.ac.be*. Our research is supported by: Research Council KUL: GOA-Mefisto 666, several PhD/postdoc & fellow grants; Flemish Government: FWO: PhD/postdoc grants, projects, G.0240.99 (multilinear algebra), G.0407.02 (support vector machines), G.0197.02 (power islands), G.0141.03 (Identification and cryptography), G.0491.03 (control for intensive care glycemia), G.0120.03 (QIT), G.0800.01 (collective intelligence), research communities (ICCoS, ANMMM); AWI: Bil. Int. Collaboration Hungary/ Poland; IWT: PhD Grants, Soft4s (softsensors), Belgian Federal Government: DWTC (IUAP IV-02 (1996-2001) and IUAP V-22 (2002-2006), PODO-II (CP/40: TMS and Sustainability); EU: CAGE; ERNSI; Eureka 2063-IMPACT; Eureka 2419-FliTE; Contract Research/agreements: Data4s, Electrabel, Elia, LMS, IPCOS, VIB.

Abstract

An algorithm for computation of an arbitrary response of an LTI system from an exact finite input/output trajectory of the system is presented. In particular, the impulse response can be obtained directly from data. A special case of the algorithm corresponds to the oblique projection.

Computation of LTI System Responses Directly from Input/Output Data

Ivan Markovsky, Jan C. Willems, Paolo Rapisarda, and Bart L.M. De Moor

***Abstract*— An algorithm for computation of an arbitrary response of an LTI system from an exact finite input/output trajectory of the system is presented. In particular, the impulse response can be obtained directly from data. A special case of the algorithm corresponds to the oblique projection.**

***Index Terms*— Deterministic subspace identification, impulse response estimation, oblique projection.**

I. Introduction

The existing deterministic balanced subspace identification algorithms [1], [2], [3] compute the impulse response and sequential zero input responses of the identified system. This paper presents a method for computation of any response directly from data. Applied to the balanced subspace identification problem, this approach gives a the algorithm proposed in [3]. Alternatively combined with Kung's algorithm, it gives a different identification method.

Our approach is based on the result of [4], which shows when the image of a Hankel matrix $\mathscr{H}(\tilde{w})$ constructed from an exact trajectory $\tilde{w} = (\tilde{u}, \tilde{y})$ of a system contains any response of the system. Therefore, in this case, any response of length $\operatorname{row dim}\big(\mathscr{H}(w)\big)$ can be obtained as a linear combination $\mathscr{H}(\tilde{w})g$, for a suitably chosen $g \in \mathbb{R}^{\bullet}$. Important refinement of this basic idea is that a response can be constructed from sequential pieces, so that an arbitrary long response can be computed from a finite trajectory $\tilde{w}$.

II. Computation of a response from data

Let $\mathscr{H}_l(\bullet)$ be a block-Hankel matrix with l block-rows, *e.g.*, with $\tilde{u}(1), \ldots, \tilde{u}(T)$,

$$\mathscr{H}_l(u) = \begin{bmatrix} \tilde{u}(1) & \tilde{u}(2) & \cdots & \tilde{u}(T-l+1) \\ \tilde{u}(2) & \tilde{u}(3) & \cdots & \tilde{u}(T-l+2) \\ \vdots & \vdots & & \vdots \\ \tilde{u}(l) & \tilde{u}(l+1) & \cdots & \tilde{u}(T) \end{bmatrix},$$

$\mathtt{n}_{\max}$ be an upper bound on the system order $\mathtt{n}$, and define $U_{\mathrm{p}} \in \mathbb{R}^{\mathtt{n}_{\max}\mathtt{m} \times \bullet}$ $U_{\mathrm{f}} \in \mathbb{R}^{l\mathtt{m} \times \bullet}$, $Y_{\mathrm{p}} \in \mathbb{R}^{\mathtt{n}_{\max}\mathtt{p} \times \bullet}$, $Y_{\mathrm{f}} \in \mathbb{R}^{l\mathtt{p} \times \bullet}$ by

$$\mathscr{H}_{\mathtt{n}_{\max}+l}(\tilde{u}) =: \begin{bmatrix} U_{\mathrm{p}} \\ U_{\mathrm{f}} \end{bmatrix}, \qquad \mathscr{H}_{\mathtt{n}_{\max}+l}(\tilde{y}) =: \begin{bmatrix} Y_{\mathrm{p}} \\ Y_{\mathrm{f}} \end{bmatrix}.$$

Theorem 1. *Let $\tilde{w} = (\tilde{u}, \tilde{y})$ be a trajectory of a controllable LTI system $\mathscr{S}$ of order $\mathtt{n} \leq \mathtt{n}_{\max}$ and let $\tilde{u}$ be persistently exciting of order $l + 2\mathtt{n}_{\max}$. Then the system of equations*

$$\begin{bmatrix} U_p \\ U_f \\ Y_p \end{bmatrix} g = \begin{bmatrix} 0 \\ u_f \\ 0 \end{bmatrix}, \tag{1}$$

is solvable for any u_f and any particular solution $\bar{g}$ allows the computation of the response y_f of $\mathscr{S}$ due to the input u_f and zero initial conditions as $y_f = Y_f \bar{g}$.

Proof: Similar statement is proved in [3, Sec. 3]. The essential lemma used is presented in [4]. □

Theorem 1 gives a block algorithm for the computation of the response y_{f}. Let $\tilde{w} = \big(\tilde{w}(1), \ldots, \tilde{w}(T)\big)$. The persistency of excitation assumption implies that

$$l \leq \frac{T+1}{\mathtt{m}+1} - 2\mathtt{n}_{\max}, \tag{2}$$

so using the block algorithm, we are limited in the length of the response y_{f} that can be computed from a finite $\tilde{w}$. It is possible, however, to find an arbitrary many sample of the response. Algorithm 1 does this by computing iteratively blocks of l consecutive samples. (The notation σ, used in step 2.3. of the algorithm, is for the shift operator. Acting on a matrix it removes the first *block* row.)

Algorithm 1 *(Computation of response from data).*
Input: $\tilde{w} = (\tilde{u}, \tilde{y})$, u_{f}, $\mathtt{n}_{\max}$.

1) *Initialization:* choose l according to (2), set $k := 0$,

$$f_{\mathrm{u}}^{(0)} := \begin{bmatrix} 0_{\mathtt{m}\mathtt{n}_{\max} \times 1} \\ u_{\mathrm{f}}(1{:}l) \end{bmatrix}, \quad \text{and} \quad f_{\mathrm{y,p}}^{(0)} := 0_{\mathtt{p}\mathtt{n}_{\max} \times 1}.$$

2) *Repeat*
   2.1. Solve $\begin{bmatrix} U_{\mathrm{p}} \\ U_{\mathrm{f}} \\ Y_{\mathrm{p}} \end{bmatrix} g^{(k)} = \begin{bmatrix} f_{\mathrm{u}}^{(k)} \\ f_{\mathrm{y,p}}^{(k)} \end{bmatrix}$ and let $\bar{g}^{(k)}$ be the particular solution found.
   2.2. Compute the response $y_{\mathrm{f}}^{(k)} := Y_{\mathrm{f}} \bar{g}^{(k)}$.
   2.3. Shift: $f_{\mathrm{u}}^{(k+1)} := \begin{bmatrix} \sigma^l f_{\mathrm{u}}^{(k)} \\ u(kl+1{:}(k+1)l) \end{bmatrix}$, $f_{\mathrm{y,p}}^{(k+1)} := \sigma^l \begin{bmatrix} f_{\mathrm{y,p}}^{(k)} \\ y_{\mathrm{f}}^{(k)} \end{bmatrix}$.
   2.4. $k := k+1$
   Until $t < kl$ $\quad (t := \#$ of samples in $u_{\mathrm{f}})$

Output: $y_{\mathrm{f}} := \operatorname{col}\big(y_{\mathrm{f}}^{(0)}, \ldots, y_{\mathrm{f}}^{(k-1)}\big)$.

Algorithm 1 works by matching the initial conditions of the l-samples long responses $y_{\mathrm{f}}^{(0)}, y_{\mathrm{f}}^{(1)}, \ldots$, so that they become sequential pieces of the desired response y_{f}. In the special case $u_{\mathrm{f}} = \delta(t) e_i$, where δ is the Kronecker delta function and e_i is the i-th unit vector, Algorithm 1 computes the i-th column of the impulse response of $\mathscr{S}$.

Ivan Markovsky, Jan C. Willems, Bart L.M. De Moor are with the Electrical Engineering Department, K.U.Leuven, Kasteelpark Arenberg 10, B-3001 Leuven, Belgium.

Paolo Rapisarda is with Dept. Mathematics, University of Maastricht, 6200 MD Maastricht, The Netherlands

TABLE I

ERROR OF ESTIMATION $e = ||Y_0 - \hat{Y}_0||_F$ AND THE CORRESPONDING AMOUNT OF OPERATIONS f IN MEGA FLOPS.

Method	$\sigma = 0.0$		$\sigma = 0.1$		$\sigma = 0.2$		$\sigma = 0.4$	
	e	f	e	f	e	f	e	f
Iterative algorithm, using the QR decomposition	10^{-14}	130	1.2990	131	2.5257	132	4.7498	132
Oblique projection, computed from the definition	10^{-10}	182	1.6497	186	3.2063	187	6.0915	189
Oblique projection, using the QR decomposition	10^{-14}	251	1.6497	251	3.2063	251	6.0915	252

III. COMPUTATION OF SEQUENTIAL ZERO INPUT RESPONSES AND LINK WITH THE OBLIQUE PROJECTION

Consider the computation of a sequence of zero input responses with the property that the corresponding sequence of initial conditions form a valid state sequence of the system. Such zero input responses are called *sequential*.

Using the result of [4], a matrix Y_0 which columns are zero input responses can be computed from the data $\tilde{w}$ by solving the system

$$\begin{bmatrix} U_\mathrm{p} \\ Y_\mathrm{p} \\ U_\mathrm{f} \end{bmatrix} G = \begin{bmatrix} U_\mathrm{p} \\ Y_\mathrm{p} \\ 0 \end{bmatrix} \tag{3}$$

and setting $Y_0 = Y_\mathrm{f} G$. Moreover, the Hankel structure of U_p and Y_p implies that Y_0 is a matrix of sequential responses. System (3) and $Y_0 = Y_\mathrm{f} G$ give a *block algorithm* for the computation of sequential zero input responses.

An *iterative algorithm* similar to Algorithm 1 is presented in [3, Sec. 4]. (The only modification needed is to replace in the initialization step of Algorithm 1 the zero block in $f_\mathrm{u}^{(0)}$ with U_p and $f_{\mathrm{y,p}}^{(0)} = 0$ with $f_{\mathrm{y,p}}^{(0)} = Y_\mathrm{p}$.) It allows to compute arbitrary long sequential free responses. Moreover, the freedom to choose the parameter l allows to relax the persistency of excitation condition and to solve more overdetermined system of equations on each iteration step.

For any particular solution $\bar{G}$ of (3), $Y_0 = Y_\mathrm{f} \bar{G}$ is equal to the oblique projection $Y_\mathrm{f} /_{U_\mathrm{f}} \begin{bmatrix} U_\mathrm{p} \\ Y_\mathrm{p} \end{bmatrix}$ of the classical subspace algorithms. Consider the least squares least norm solution of (3)

$$\bar{G} = \begin{bmatrix} W_\mathrm{p}^\top & U_\mathrm{f}^\top \end{bmatrix} \begin{bmatrix} W_\mathrm{p} W_\mathrm{p}^\top & W_\mathrm{p} U_\mathrm{f}^\top \\ U_\mathrm{f} W_\mathrm{p}^\top & U_\mathrm{f} U_\mathrm{f}^\top \end{bmatrix}^+ \begin{bmatrix} W_\mathrm{p} \\ 0 \end{bmatrix}, \quad W_\mathrm{p} := \begin{bmatrix} U_\mathrm{p} \\ Y_\mathrm{p} \end{bmatrix}.$$

We have $Y_\mathrm{f} \bar{G} = Y_\mathrm{f} /_{U_\mathrm{f}} W_\mathrm{p}$, *i.e.*, the oblique projection of Y_f along U_f on W_p. (See [2, p. 21, eqn. (1.4)] for the definition of the oblique projection $Y_\mathrm{f} /_{U_\mathrm{f}} W_\mathrm{p}$.) Thus,

> the oblique projection is an implementation of the block algorithm for the computation of the matrix Y_0 of sequential zero input responses.

IV. SIMULATION EXAMPLE

The system $\mathscr{S}$ is of order $\mathtt{n} = 3$. The input $\tilde{u}$ is a unit variance white noise and the data available for identification is the corresponding trajectory $\tilde{w} = (\tilde{u}, \tilde{y})$ of $\mathscr{S}$, corrupted by white noise with standard deviation σ. The simulation time is $T = 100$ and the parameters $\mathtt{n}_{\max}$ and l of the iterative algorithm are chosen as follows: $\mathtt{n}_{\max} = l = 3$.

Table I shows the error of estimation $e := ||Y_0 - \hat{Y}_0||_F$ and the corresponding amount of operations, where Y_0 is a matrix of exact sequential zero input responses with length 10 and $\hat{Y}_0$ is its estimate computed from data. The estimate is computed in three ways: by the proposed iterative algorithm, implemented with the QR factorization; by the oblique projection, computed directly from the definition [2, eqn. (1.4)]; and by the oblique projection, computed via the QR factorization.

The iterative algorithm needs less computations and gives more accurate results than the alternatives. The reason for this is that selecting the parameter $l = 3$ instead of $l = 10$, as in a block computation, results in a more overdetermined system of equations in the iterative algorithm compared with system (3) used in the block algorithm. As a result the noise is averaged over more samples, which leads to statistically better estimate. Solving several more overdetermined systems of equations instead of one more rectangular system can be more efficient, as it is in the example. In fact, the freedom in the choice of the parameter l makes possible to optimize efficiency or another criterion.

ACKNOWLEDGMENTS

Our research is supported by: Research Council KUL: GOA-Mefisto 666, several PhD/postdoc & fellow grants; Flemish Government: FWO: PhD/postdoc grants, projects, G.0240.99 (multilinear algebra), G.0407.02 (support vector machines), G.0197.02 (power islands), G.0141.03 (Identification and cryptography), G.0491.03 (control for intensive care glycemia), G.0120.03 (QIT), G.0800.01 (collective intelligence), research communities (ICCoS, ANMMM); AWI: Bil. Int. Collaboration Hungary/ Poland; IWT: PhD Grants, Soft4s (softsensors), Belgian Federal Government: DWTC (IUAP IV-02 (1996-2001) and IUAP V-22 (2002-2006), PODO-II (CP/40: TMS and Sustainability); EU: CAGE; ERNSI; Eureka 2063-IMPACT; Eureka 2419-FliTE; Contract Research/agreements: Data4s, Electrabel, Elia, LMS, IPCOS, VIB.

REFERENCES

[1] M. Moonen and J. Ramos, "A subspace algorithm for balanced state space system identification," *IEEE Trans. on Aut. Control*, vol. 38, pp. 1727–1729, 1993.

[2] P. Van Overschee and B. De Moor, *Subspace Identification for Linear Systems: Theory, Implementation, Applications*. Kluwer Academic Publishers, 1996.

[3] I. Markovsky, J. C. Willems, P. Rapisarda, and B. D. Moor, "Algorithms for deterministic balanced subspace identification," Dept. EE, K.U. Leuven, Tech. Rep. 04–13, 2004.

[4] J. C. Willems, I. Markovsky, P. Rapisarda, and B. D. Moor, "A note on persistency of excitation," Dept. EE, K.U. Leuven, Tech. Rep. 04–54, 2004.