Author's response to the reviewers' comments

Manuscript number 09-0233
Title: Closed-loop data-driven simulation
Author: Ivan Markovsky

October 5, 2009

We thank the reviewers and the editors for their relevant and useful comments. In this document, we quote in **bold face** statements from the reports. Our replies follow in ordinary print.

Answer to the editors

the notations of the paper may have caused problems for people to understand the contribution ... new notations keep popping up

In the revised version of the paper, we took special care to avoid unnecessary notation and define in advance the notation that is used. Subsection "Notation" of the new section "Introduction" collects all definitions used in the paper.

... undefined notation, including r_k in (2) ...

r_k for $k = 0, 1, \dots, n$ denote the coefficients of a polynomial $r \in \mathbb{R}[z]$ (of degree n). In the revised version of the paper, we have added the missing definition: $r(z) = r_0 + r_1 z + \cdots + r_n z^n$.

... and R_w and R_r in step 1 of Algorithm 1

Indeed, R_w and R_r are not defined in Algorithm 1. (They are only defined in the proof of the theorem). In fact, the steps of the algorithm can not be understand without reading their derivation first. (See also the next comment.) In the revised version of the paper, we give the derivation of the algorithm first (in the original version of the paper it was in the proof of the theorem) and then the summary of the algorithm. We believe this reversal in the exposition caused the original confusion, which is now avoided.

I personally have had hard time to understand the equation in 1 of Algorithm 1 that does not follow easily from the kernel representation and image representation in page 2. Some derivations are needed.

Yes, we agree. In fact, one page of derivations are needed—they are given in the proof of the theorem. We revised the paper by first giving the solution of the closed-loop data-driven simulation problem and then the summary of the resulting algorithm. (See the previous comment.)

Answer to Reviewer 1

Whether this [data-driven simulation] is conducted in open or closed loop is inessential.

1. In data-driven simulation, a desired response is computed from a response of the unknown system. In closed-loop data-driven simulation, the closed-loop behavior is computed from a response of the unknown system *and*

a representation of a controller. It is not clear a priori how a trajectory of the plant and a representation of a controller can be combined in a computation of the behavior of the closed-loop system plant–controller. The way to do this involves ideas from the behavioral system theory [WRMM05] and is a new result in this paper.

2. Another important difference between data-driven simulation and closed-loop data-driven simulation is that in the former a single response is sought, while in the latter, in general, a set of responses (the closed-loop behavior) is computed.

Since the underlying system is LTI, the essential problem is to solve a set of linear equations.

We agree. However, this fact does not imply that the problem is trivial or unimportant. Certainly, solution of a set of linear equations appears in many nontrivial problems, *e.g.*, realization theory, which is without doubt a major result in linear system theory from the 70's and also boils down to solution of a system of linear equations. It is not necessary to argue that solution of a set of linear equations also is the computational tool for many (all?) practically important problems. Therefore, the criticism of the reviewer to the presented results in the paper does not imply that closed-loop data-driven simulation is trivial or unimportant. In the answer to the next comment and the introduction of the paper, we argue that the problem is nontrivial and important indeed.

I fail to see whether the paper has brought up anything new and significant, either conceptually or numerically.

We hope the reviewer agrees that the paper proposes and solves a *new* problem. As pointed out in the answer to the first comment of the reviewer, the problem is not trivial and as explained in the paper, it is motivated by an application in unfalsified control, so we believe that it is practically useful as well as theoretically interesting. We bring new ideas of combing a trajectory of an unknown system with a representation of a control for that system. The results in the paper lead to concrete computational algorithms that everyone can implement (or download the Matlab code from our web page) and use.

Answer to Reviewer 2

It is assumed in this paper that the plant is linear time invariant and measured data are noise free. . . . It seems to me that the assumptions in this paper are too strong.

We agree that the exact data assumption is strong. Indeed, it is almost never satisfied in practice. This, however, does not invalidate the results for the following reason: a trivial modification of the algorithm—replace solution of a linear system of equation by an approximate solution in, *e.g.*, the least squares sense, rank test by a numerical rank test, and computation of a basis of subspace by computation of an approximate basis, using the singular value decomposition—leads to a practically useful algorithm that can cope with a moderate amount of noise on the data.

Note that a high level library for numerical linear algebra computations will probably do these substitutions automatically, so code designed for exact data may actually work without modification with noisy data. This is the case for Matlab code: Figure 1 shows the degradation of the algorithm's performance in the presence of noise. It is noteworthy that the algorithm does not break in the presence of noise; its performance gradually degrades as the SNR decreases.

We would like to point out that in the class of the highly successful subspace identification methods, historically the first algorithms [Gop69, Bud71] are designed for exact data. (In the literature [VD96, Chapter 2] they are called deterministic subspace identification methods.) Subsequently, subspace identification was extended to the arguably more useful stochastic and combined stochastic-deterministic problems. The core computational steps (orthogonal and oblique projections) in these newer variations, however, are exactly the same as the ones in the original deterministic algorithms. For special identification problems, where additional prior knowledge is used, more efficient algorithms were proposed, however the value of the breakthrough came from the deterministic subspace identification methods. This fact is often overlooked.

We are aware of data-driven simulation algorithms specifically designed to deal with noisy data: our result from

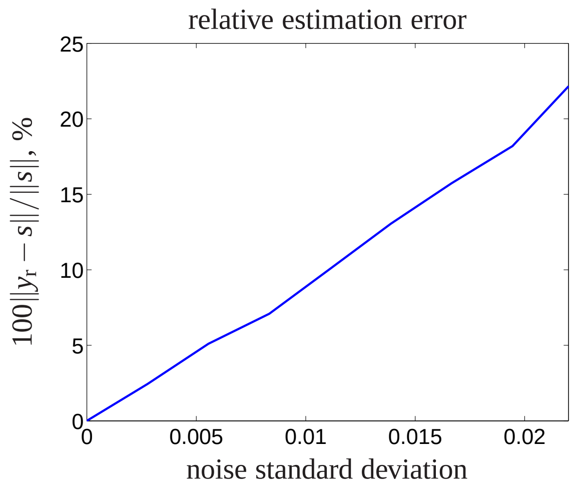

Figure 1: Empirical results for the performance of the Algorithm 1 in case of noisy data ($y_d = y_{true} + \text{noise}$). The simulation setup is the same as the one described in the paper. Please note that the given data is only 10 samples long noisy trajectory, which is extremely short for obtaining good estimates. This is compensated by the relatively small amount of noise. The presented results are averaged over 100 noise repetitions. Main point: performance degrades gracefully rather and we do not see a breakdown of the algorithm due to the noisy data.

[MWRM05], which is derived under the same assumptions as the ones in the present paper, was extended to the noisy case in [RPR08]. We envisage that the same will happen with the closed-loop data-driven simulation problem.

A paragraph explaining these important points was added in the introduction:

> A standing assumption throughout the paper is that the data is generated by an LTI system of bounded complexity. Admittedly, this assumption is practically unrealistic, however, it is convenient for the theoretical study (*cf.*, deterministic subspace identification) and trivial modifications of the algorithm—replace solution of a linear system of equation by an approximate solution, rank test by a numerical rank test, *etc.*—leads to practically useful algorithms that can cope with noise on the data (*cf.*, stochastic subspace identification). We envisage that stochastic version of the results presented in this paper will appear in near future.

Under these assumptions, there should not be a significant difference between normal LTI model and measured input/output data.

We disagree—obviously the data may not be informative to specify the data generating system, so there is difference between data and model. If the analysis is done properly, this difference is significant. This nontrivial aspect of the exact identification and data-driven simulation problems was resolved in [WRMM05].

The main contribution of this paper is limited.

Please refer to the answer of the third comment of Reviewer 1.

technical comments

1. The author should present certain advantages to study the problem with "the behavioral language". It seems to me that this tool makes the problem to be more complicated.

We have added the following explanation in the introduction:

> The behavioural setting [Wil87] is especially suitable for solution of data-driven simulation and control problems because it treats a dynamical system as a set of trajectories (rather than equations) thus making explicit the relation between a trajectory and the system that generates the trajectory.

Also the current work steps on previous work done in the behavioral setting [WRMM05, MWRM05, MWVD06], which makes it natural to use the same philosophy and notation.

2. The notations regarding to the description of the system are quite confused. For example $\mathbb{R}^w$ and $\mathbb{R}^m$ in page 1 have different meaning but the author did not clarify it.

We took extra care to avoid unnecessary notation and to define all notation that we use in advance. The meaning of $\mathbb{R}^{\mathtt{w}}$ and $\mathbb{R}^{\mathtt{m}}$ on page 1 is not different — these are real vector spaces, of dimension $\mathtt{w}$ and $\mathtt{m}$, respectively, as now defined in subsection "Notation".

3. The author stated in second line of Section 1 that a system is a set of function from $\mathbb{N}$ to $\mathbb{R}^w$. And then in the second equation of page 1, there is $(w_p, w_f) \in \mathscr{B}$. It should be noticed that w_p is a function from $1, \dots, T$ to $\mathbb{R}^{\mathtt{w}}$. Author should carefully present all notations in the paper.

The meaning of (w_p, w_f) is concatenation of the trajectories w_p and w_f. We have not explain this in the original version of the paper, which is probably the reason for the confusion. The revised version of the paper introduces all relevant notation and, in particular, the meaning of (w_p, w_f).

4. The paper does not include sections for introduction and conclusion. It is very unusual.

"Introduction" and "Conclusion" sections are added.

References

[Bud71] M. A. Budin. Minimal realization of discrete linear systems from input-output observations. *IEEE Trans. Automat. Control*, 16(5):395–401, 1971.

[Gop69] B. Gopinath. On the identification of linear time-invariant systems from input-output data. *The Bell System Technical J.*, 48(5):1101–1113, 1969.

[MWRM05] I. Markovsky, J. C. Willems, P. Rapisarda, and B. De Moor. Algorithms for deterministic balanced subspace identification. *Automatica*, 41(5):755–766, 2005.

[MWVD06] I. Markovsky, J. C. Willems, S. Van Huffel, and B. De Moor. *Exact and Approximate Modeling of Linear Systems: A Behavioral Approach*. Number 11 in Monographs on Mathematical Modeling and Computation. SIAM, March 2006.

[RPR08] E. Reynders, R. Pintelon, and G. De Roeck. Consistent impulse-response estimation and system realization from noisy data. *IEEE Transactions on Signal Processing*, 56:2696–2705, 2008.

[VD96] P. Van Overschee and B. De Moor. *Subspace Identification for Linear Systems: Theory, Implementation, Applications*. Kluwer, Boston, 1996.

[Wil87] J. C. Willems. From time series to linear system—Part I. Finite dimensional linear time invariant systems, Part II. Exact modelling, Part III. Approximate modelling. *Automatica*, 22, 23:561–580, 675–694, 87–115, 1986, 1987.

[WRMM05] J. C. Willems, P. Rapisarda, I. Markovsky, and B. De Moor. A note on persistency of excitation. *Control Lett.*, 54(4):325–329, 2005.